Analyzing and Visualizing the EV Vehicles Population

Venkateswara Rao[1], Vinay Marella[2], Harsha Sai Teja Nukala[3], Khaja Nayab Rasool Shaik[4]
Northwest Missouri State University, Maryville MO 64468, USA
{S566615, s565585, S566999, S566579}@nwmissouri.edu

Abstract

The global automotive industry is increasingly adopting electric vehicles (EVs) to address environmental and energy challenges. This research examines EV population data from various sources, offering a detailed perspective on the market. Utilizing advanced visualization tools such as interactive maps and charts, it uncovers key insights into adoption trends, regional distributions, and contributing factors. By delivering actionable insights for policymakers and industry leaders, this study supports efforts to expedite the shift towards sustainable transportation. Ultimately, it serves as a critical resource for decision-makers navigating the intricacies of the EV market, fostering progress toward a greener future.

1 Introduction

The global automotive industry is undergoing a significant transformation toward electric vehicles (EVs), driven by increasing concerns about environmental sustainability and energy efficiency. This research undertakes an in-depth analysis and visualization of the EV population, leveraging advanced tools and technologies such as PySpark and Tableau.

By utilizing data from diverse sources, including government records, industry reports, and academic publications, the study examines key metrics such as regional distribution, adoption trends, vehicle categories, and technological innovations. The integration of PySpark and Tableau ensures streamlined data handling, visualization, and analysis, providing actionable insights for decision-makers to accelerate the transition toward sustainable transportation.

2 Steps Involved

1. **Data Acquisition:** Download the EV population dataset in CSV format from Kaggle.
2. **Data Preprocessing:** Use Python libraries like Pandas to handle missing values, clean data, and apply formatting.

3. **Visualization with Tableau:** Create visualizations such as:
    - Line plots: To illustrate EV adoption trends.
    - Bar charts: To compare EV populations by region or manufacturer.
    - Histograms: To show the distribution of EVs by model year.

3 Architecture

The project follows a structured framework encompassing data collection, preprocessing, analysis, and visualization. Figure 1 illustrates the high-level architecture of data flow.

1. **Data Collection:** Download the EV population dataset from Kaggle.
2. **Data Preprocessing:** Use PySpark for cleaning and filtering data.
3. **Data Analysis:** Perform analyses such as adoption rates, vehicle types, trends by model year, and price sensitivity using PySpark's DataFrame API.
4. **Data Visualization:** Use Tableau to create interactive visualizations for stakeholders.
5. **Presentation:** Combine insights and visualizations into a comprehensive report.

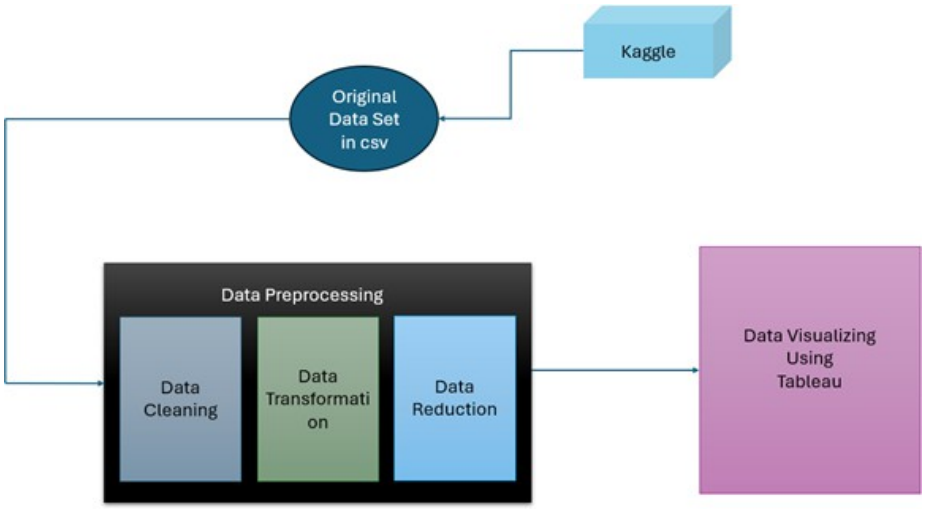


Figure 1: High-Level Architecture of Data Flow

4 Project Description

The project addresses key metrics such as data quality, the 5Vs, latency, processing time, resource utilization, security, and cost:

- **Data Quality:** PySpark ensures data integrity by addressing missing values and inconsistencies.
- **5Vs of Big Data:**
  - Volume: PySpark handles large datasets efficiently.
  - Variety: Supports diverse data types (e.g., numerical, categorical, temporal).
  - Velocity: Processes data in real-time or near-real-time.
  - Veracity: Ensures data reliability.
  - Value: Provides actionable insights for stakeholders.
- **Latency:** PySpark's in-memory computation minimizes latency.
- **Processing Time:** Distributed computing reduces overall processing time.
- **Resource Utilization:** Dynamically adjusts resources for optimal performance.
- **Security:** Implements encryption, access controls, and authentication mechanisms.
- **Cost:** Optimizes infrastructure and licensing costs.

5 Research Goals

1. **Incorporating the 5Vs:** Analyze and visualize the EV population data by addressing the dimensions of volume, variety, velocity, and value to gain comprehensive insights.
2. **Volume (EV Adoption Rates by Region):**Compare EV adoption rates across various regions to understand geographic preferences and identify regional trends in EV adoption.
3. **Volume (Total EV Sales):** Assess the total number of EVs sold to date, providing a quantitative measure of market growth and adoption over time.
4. **Variety (Types of Electric Vehicles):** Investigate the range of EV types in the market, including passenger vehicles, commercial EVs, and alternative options such as electric bicycles and scooters, to capture the diversity of offerings.

5. **Velocity (Electric Range Averages)** Evaluate the average electric range of vehicles over specific timeframes to gauge advancements in EV battery technology and overall efficiency.

6. **Velocity (Trends by Model Year):** Analyze adoption trends by vehicle model year to track the pace of market growth and the introduction of new EV models over time.

7. **Value (Price Sensitivity Analysis):** Examine consumer price sensitivity to evaluate the value proposition of EVs compared to traditional vehicles. This includes analyzing factors like upfront costs, long-term savings, and maintenance expenses.

article graphicx url float

6 Results Summary

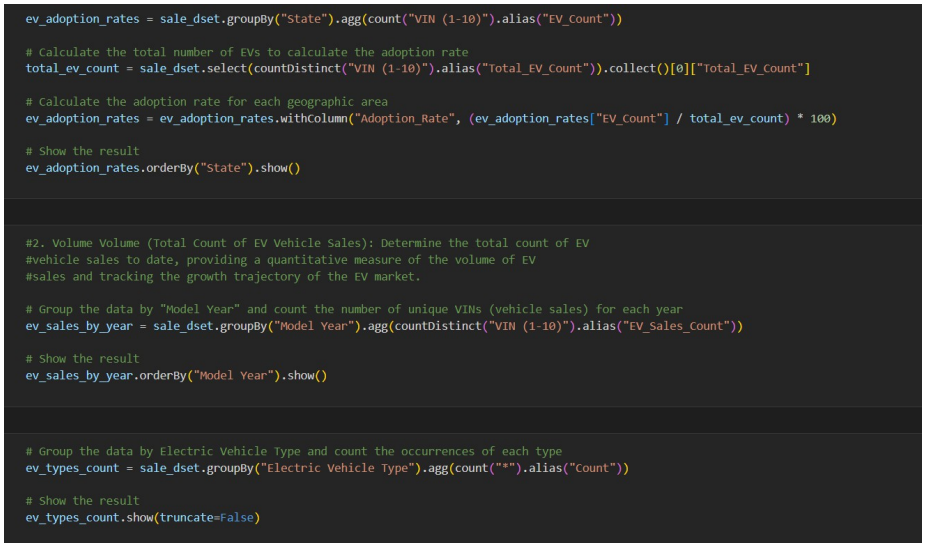
This project demonstrates the use of PySpark for analyzing large-scale EV population datasets and Tableau for visualization. By addressing adoption rates, sales trends, and vehicle characteristics, it provides actionable insights for stakeholders to make informed decisions about EV adoption and market strategy.

```python
# Import necessary libraries
from pyspark.sql import SparkSession
from pyspark.sql import functions as F
from pyspark.sql.functions import col, max, sum, min, rank , countDistinct,count,avg
from pyspark.sql.window import Window

+ Code  + Markdown

# Create SparkSession
spark = SparkSession.builder.appName("EV Population Analysis").getOrCreate()


# Read the dataset
sale_dset = spark.read.csv("Electric_Vehicle_Population_Data.csv", header=True, inferSchema=True)
# sale_dset.show(truncate=False)
sale_dset.select("Electric Vehicle Type", "Model").show()
```

Figure 2: Sample Source Code: Data Preprocessing with PySpark

```python
# Group the data by Model Year and calculate the average electric range for each year
average_range_by_year = sale_dset.groupBy("Model Year").agg(avg("Electric Range").alias("Average_Electric_Range"))

# Show the result
average_range_by_year.orderBy("Model Year").show()


# Group the data by Model Year and count the number of EVs for each year
ev_adoption_by_year = sale_dset.groupBy("Model Year").agg(count("*").alias("EV_Count"))

# Show the result
ev_adoption_by_year.orderBy("Model Year").show()


# Group the data by Model Year and Make and count the number of sales for each combination
sales_by_year_make = sale_dset.groupBy("Model Year", "Make").agg(count("*").alias("Sales_Count"))

# Show the result
sales_by_year_make.orderBy("Model Year", "Make").show()
```

Figure 3: Sample Source Code: Feature Engineering with PySpark



```
ev_adoption_rates = sale_dset.groupBy("State").agg(count("VIN (1-10)").alias("EV_Count"))

# Calculate the total number of EVs to calculate the adoption rate
total_ev_count = sale_dset.select(countDistinct("VIN (1-10)").alias("Total_EV_Count")).collect()[0]["Total_EV_Count"]

# Calculate the adoption rate for each geographic area
ev_adoption_rates = ev_adoption_rates.withColumn("Adoption_Rate", (ev_adoption_rates["EV_Count"] / total_ev_count) * 100)

# Show the result
ev_adoption_rates.orderBy("State").show()
```

```
#2. Volume Volume (Total Count of EV Vehicle Sales): Determine the total count of EV
#vehicle sales to date, providing a quantitative measure of the volume of EV
#sales and tracking the growth trajectory of the EV market.

# Group the data by "Model Year" and count the number of unique VINs (vehicle sales) for each year
ev_sales_by_year = sale_dset.groupBy("Model Year").agg(countDistinct("VIN (1-10)").alias("EV_Sales_Count"))

# Show the result
ev_sales_by_year.orderBy("Model Year").show()
```

```
# Group the data by Electric Vehicle Type and count the occurrences of each type
ev_types_count = sale_dset.groupBy("Electric Vehicle Type").agg(count("*").alias("Count"))

# Show the result
ev_types_count.show(truncate=False)
```

Figure 4: Sample Source Code: Visualization Integration with Tableau

7 Conclusion

This project demonstrates the use of PySpark for analyzing large-scale EV population datasets and Tableau for visualization. By addressing adoption rates, sales trends, and vehicle characteristics, it provides actionable insights for stakeholders to make informed decisions about EV adoption and market strategy.

References

1. Zaharia, M., et al. (2010). Spark: Cluster computing with working sets.
2. Electric Vehicle (EV) Data: `https://ww2.arb.ca.gov/our-work/programs/electric-vehicle-population-data`
3. Union of Concerned Scientists: `https://www.ucsusa.org/resources/electric-vehicles-evs`
4. Congressional Research Service: `https://fas.org/sgp/crs/misc/R42502.pdf`
5. International Energy Agency (IEA): `https://www.iea.org/reports/electric-vehicle-market-repo`
6. Battery University: `https://batteryuniversity.com/learn/article/electric_vehicle_ev`
7. GitHub Repository: `https://github.com/HarshaNWMS/Spirit_EV`