Assignment 2 Report

Team 51

Muskan Raina (2021101066)

Arghya Roy (2021115008)

Attribute Oriented Induction (AOI)

This task involved extracting characteristic rules using attribute-oriented induction. We did the following steps on the cleaned data.

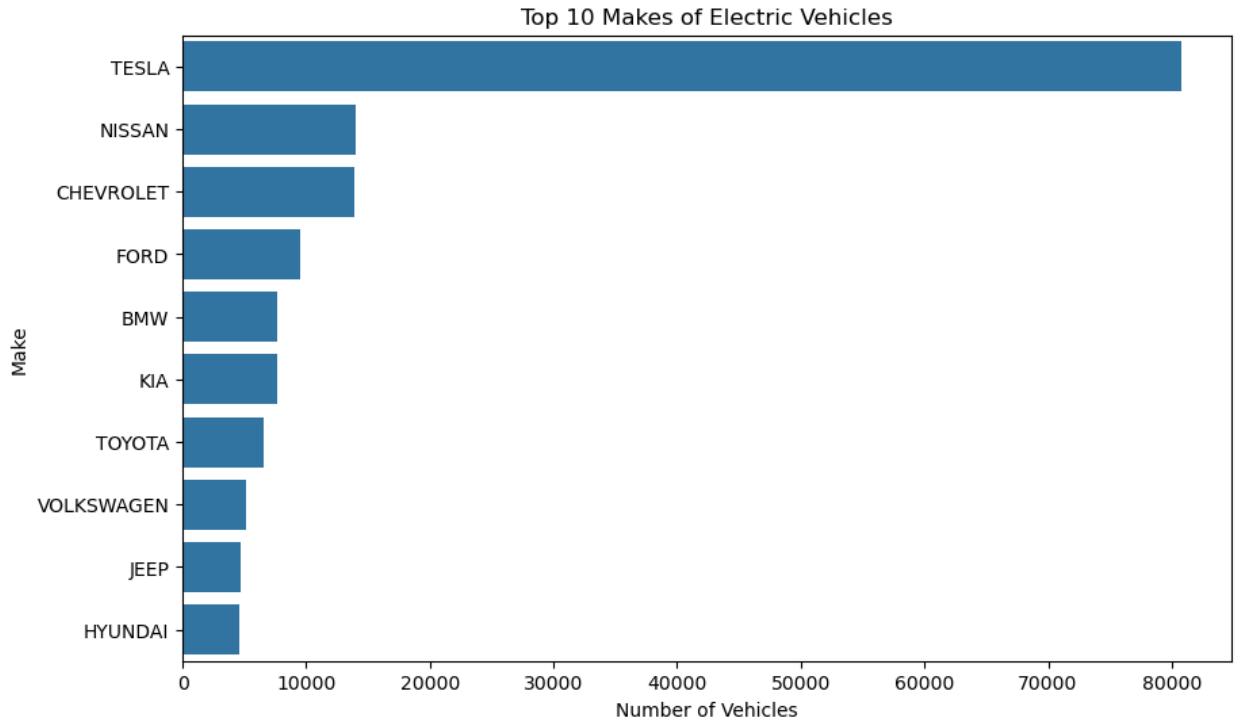
Steps

1. Select the task-relevant data relation
2. Perform attribute-oriented induction i.e generalization is performed on each attribute of P
3. Simplify the generalized relation - if only one attribute of several tuples contains distinct values, the several tuples can be reduced into one by taking the distinct values of that attribute as a set

The following strategies were used for generalization:

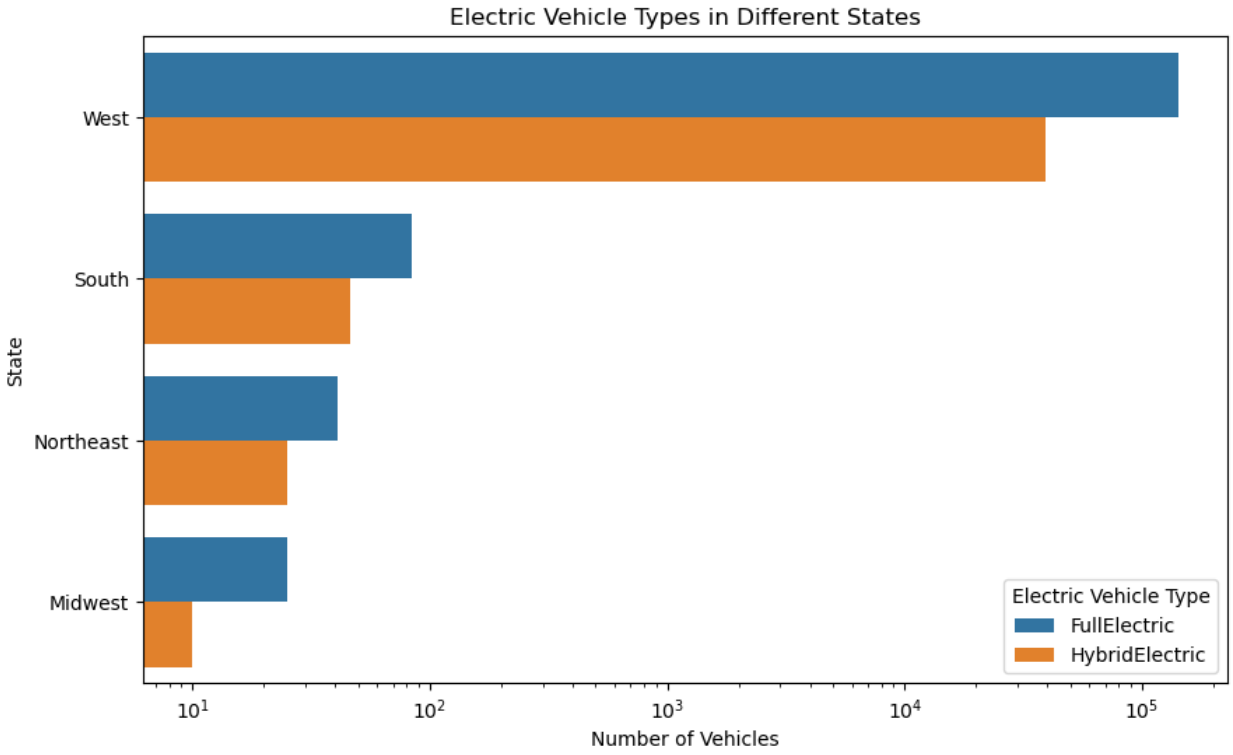
1. If there is a large set of distinct values for an attribute but there is no higher level concept provided for the attribute, the attribute should be removed during generalization.
2. Generalization should be performed on the smallest decomposable components of a data relation.
3. If there are many distinct values for an attribute and there exists a higher level concept in the concept tree for the attribute, each value in the attribute of the relation should be substituted by a higher level concept in the learning process.
4. If the number of distinct values in a resulting relation is larger than the specified threshold value, further generalization on this attribute should be performed.

Analysis



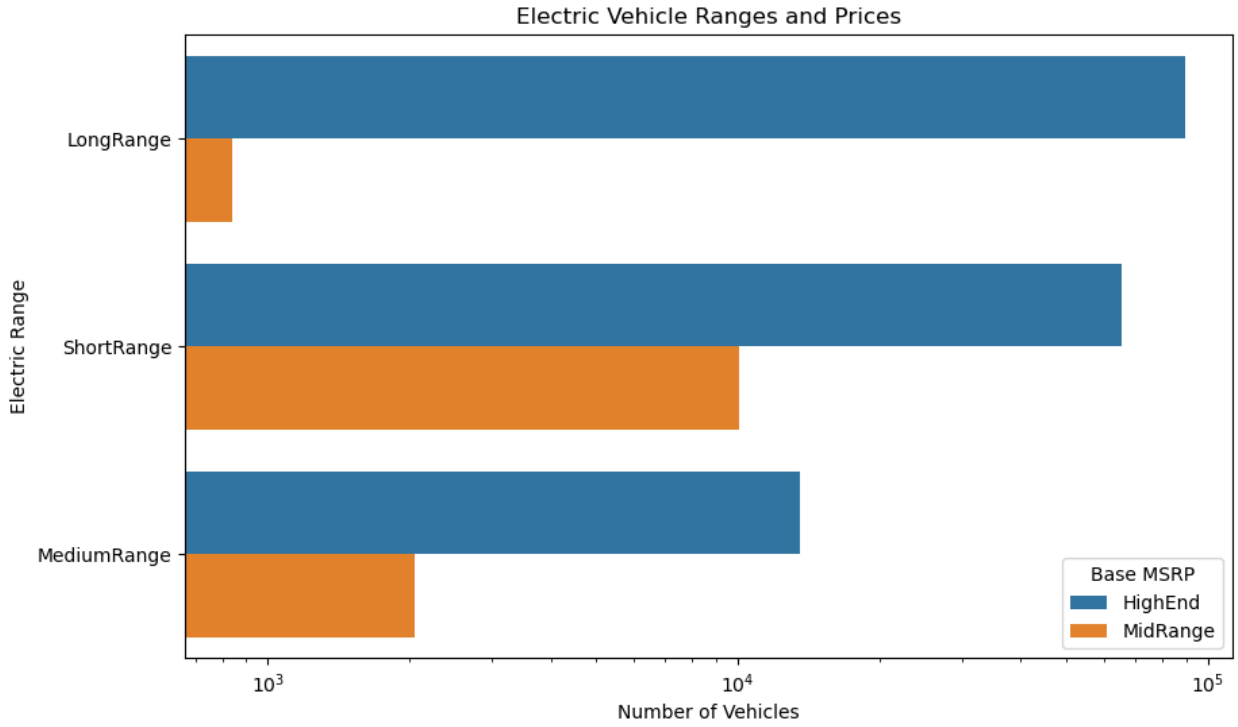
The image shows a horizontal bar chart of the "Top 10 Makes of Electric Vehicles" by number of vehicles.

Tesla is clearly dominating the electric vehicle market among these top 10 manufacturers, with significantly more vehicles than any other brand - about 80,000 vehicles. Nissan and Chevrolet are the next most popular makes, but they have far fewer vehicles compared to Tesla - roughly 20,000 each. The remaining brands (Ford, BMW, Kia, Toyota, Volkswagen, Jeep, and Hyundai) have progressively smaller market shares.



This graph shows the distribution of electric vehicle types (Full Electric and Hybrid Electric) across different regions of the United States. It provides valuable insights into the geographic distribution of electric vehicle adoption in the US, highlighting regional preferences and potential areas for growth in the EV market.

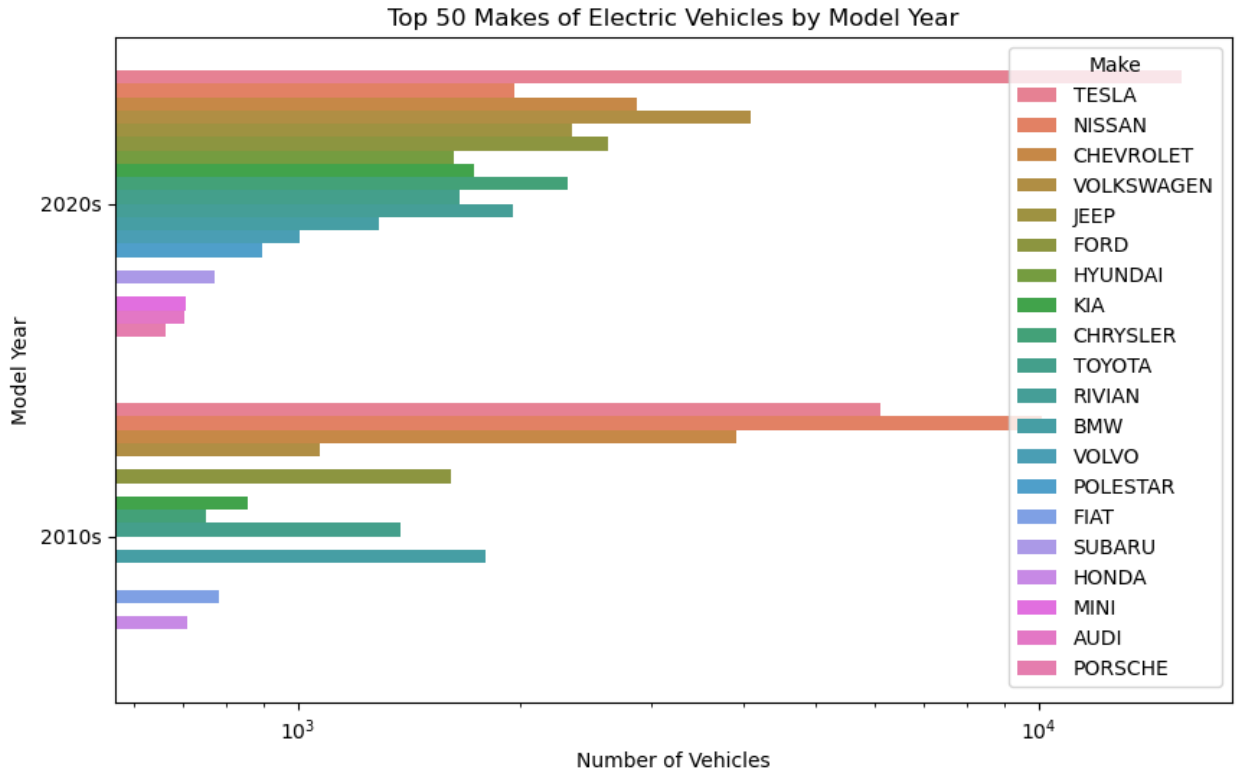
- West has the highest number of both full electric and hybrid electric vehicles by a significant margin while south ranks second in both categories.
- In all regions, there are more full electric vehicles than hybrid electric vehicles.
- The West dominates the electric vehicle market, with numbers in the tens of thousands for both types. Other regions have adoption in the hundreds to low thousands range.
- The significant regional differences suggest that local policies, incentives, and infrastructure may play a large role in EV adoption.



This graph presents information about electric vehicle ranges and their corresponding price categories (Base MSRP).

- For long range vehicles, the vast majority are HighEnd pricey and very few MidRange priced options. ShortRange vehicles are more evenly split between HighEnd and MidRange prices
- Across all range categories, HighEnd priced vehicles outnumber MidRange priced ones
- LongRange, HighEnd vehicles dominate the market, suggesting consumers prioritize range and are willing to pay for it.

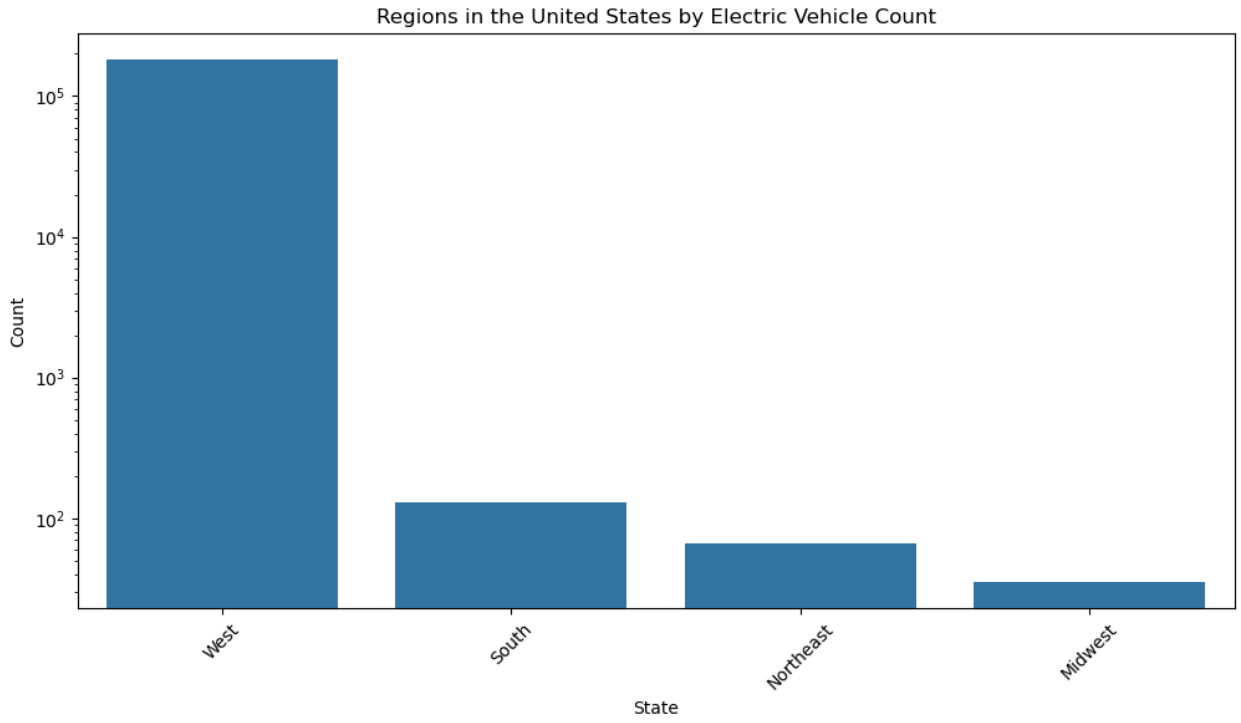
This graph provides valuable insights into the current state of the electric vehicle market, highlighting the relationship between range capabilities and pricing, and revealing potential areas for market development in more affordable options.



This graph shows the top 50 makes of electric vehicles by model year, divided into two decades: 2010s and 2020s.

- Tesla is the clear market leader in the 2020s decades, with significantly more vehicles than any other brands, whereas Nissan leads in the 2010s.
- The 2020s show a more diverse and competitive EV market with more brands entering. Overall numbers of EVs have increased significantly from the 2010s to 2020s across most brands.
- The increased number and diversity of brands in the 2020s suggest a maturing EV market with more options for consumers.

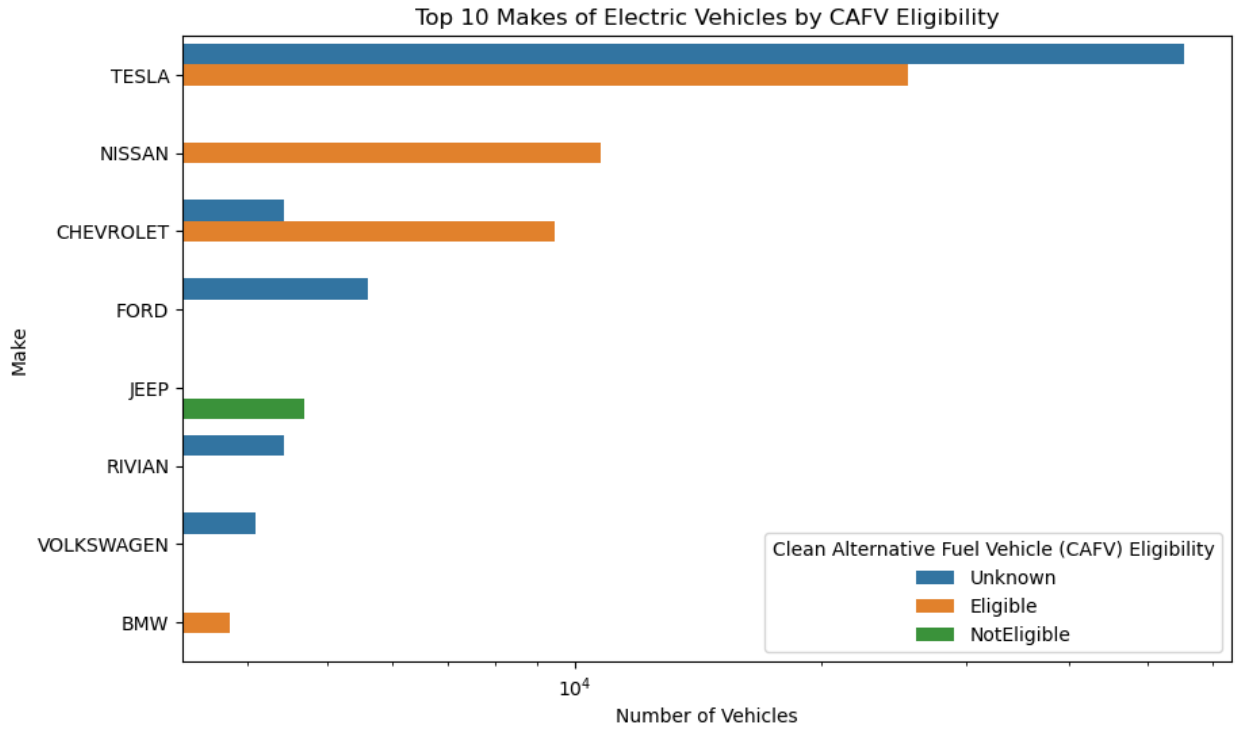
This graph provides valuable insights into the evolution of the electric vehicle market over the past decade, highlighting the emergence of new players and the changing strategies of established automakers in the EV space.



This graph shows the distribution of electric vehicle counts across different regions in the United States, grouped by region.

- The West's electric vehicle count is approximately two orders of magnitude higher than the next closest region. This suggests a significant concentration of EV adoption in Western states.
- This dominance could be due to factors such as stricter environmental regulations, presence of EV manufacturers (e.g., Tesla in California) and cultural factors favoring EV adoption
- Policymakers and manufacturers might focus on increasing adoption in the Midwest and other regions to balance out the distribution.

This visualization effectively highlights the regional disparities in EV adoption across the United States, with the West standing out as the clear leader in the electric vehicle market.



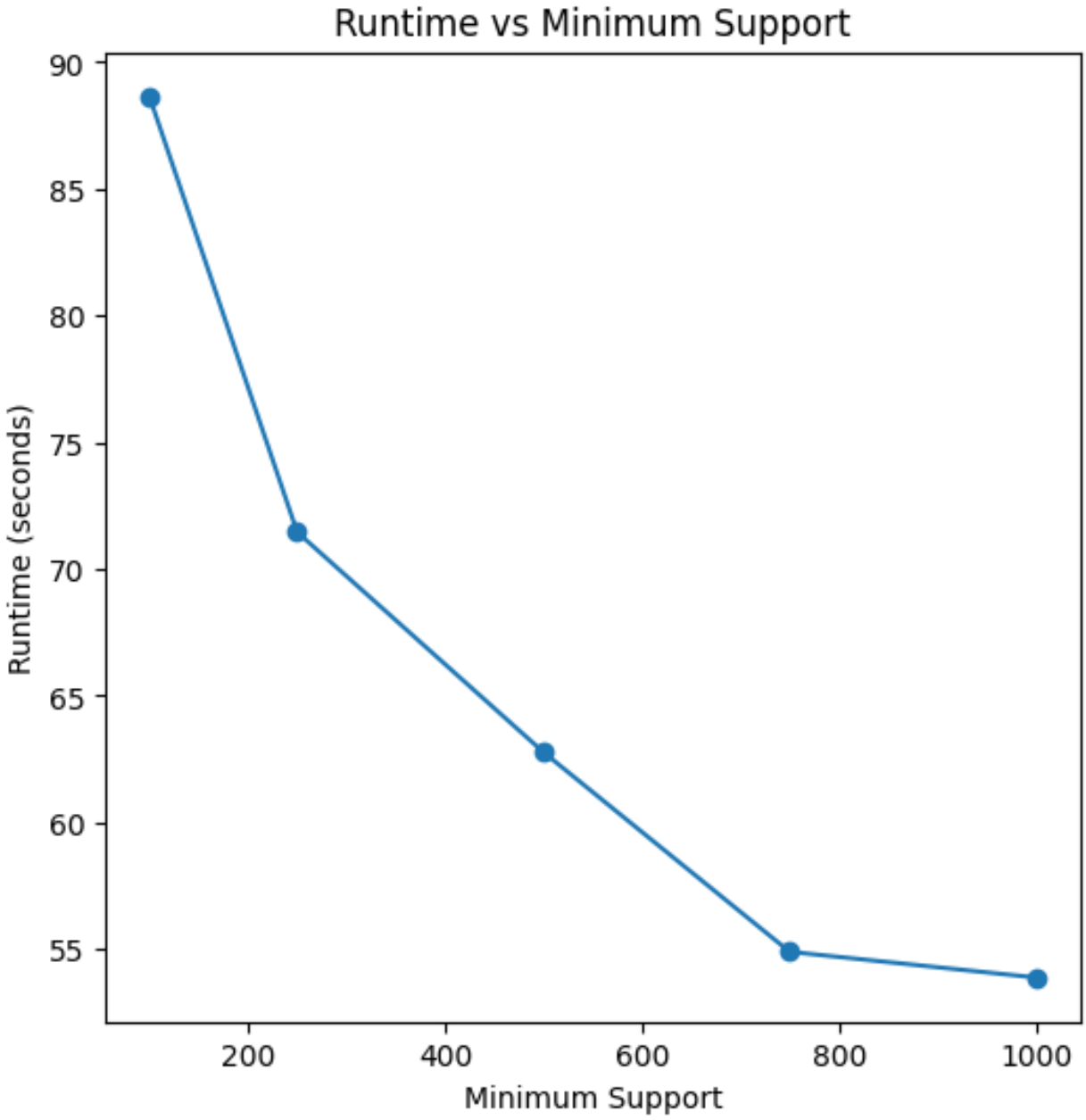
This graph shows the top 10 makes of electric vehicles categorized by their Clean Alternative Fuel Vehicle (CAFV) Eligibility.

- Tesla is the clear market leader, with significantly more vehicles than any other brand. Nissan and Chevrolet follow as distant second and third.
- Majority of Tesla vehicles have known CAFV eligibility status. All Nissan vehicles shown are "Eligible" for CAFV. Majority of Chevrolet vehicles are "Eligible" and a small portion has "Unknown" eligibility.
- Most manufacturers have either "Eligible" or "Unknown" status for their vehicles. "NotEligible" status is rare, with Jeep being the notable exception.

This visualization provides insights into the CAFV eligibility of various electric vehicle brands, which could be valuable for consumers considering eco-friendly options and for policymakers assessing the impact of CAFV incentives on the EV market.

Bottom Up Cube (BUC) Algorithm

In-Memory Implementation



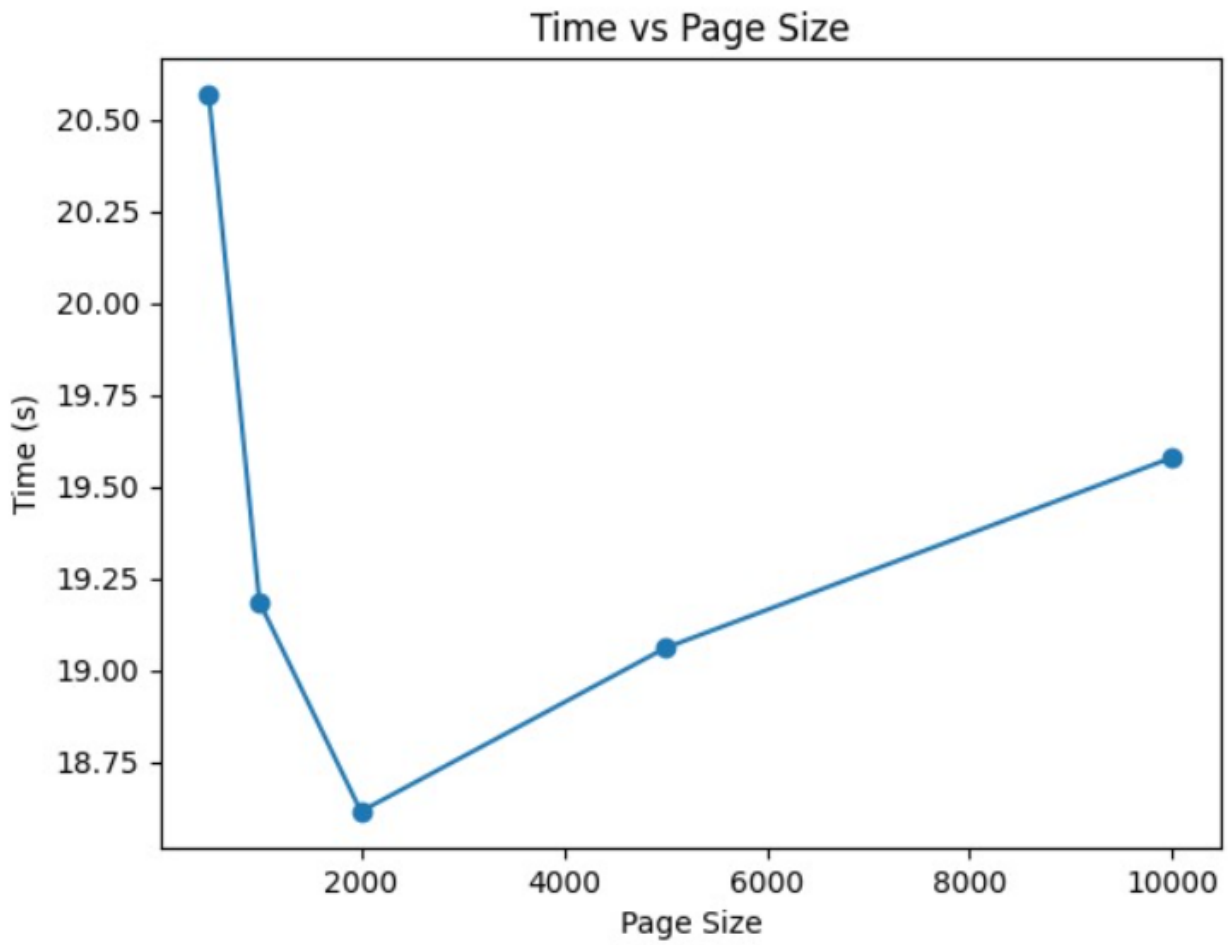
The graph shows the relationship between Minimum Support on the x-axis and Runtime (in seconds) on the y-axis for the BUC in-memory implementation.

1. Inverse relationship: There's a clear inverse relationship between Minimum Support and Runtime. As Minimum Support increases, the Runtime decreases.

2. Non-linear trend: The decrease in Runtime is not linear. The curve shows a steeper decline at lower Minimum Support values and becomes less steep as Minimum Support increases.

This trend is seen because increasing the Minimum Support threshold reduces the number of candidate itemsets to be evaluated, thus decreasing the overall runtime. However, this comes at the cost of potentially missing less frequent but possibly interesting patterns in the data.

BUC Out-of-Memory Implementation



This graph shows the relationship between Page Size and Time (in seconds) for BUC out-of-memory implementation.

Small page sizes: At very small page sizes, the execution time is high because the algorithm must frequently read from and write to disk. This results in excessive I/O operations, which are time-consuming.

Decreasing time as page size initially increases: As the page size grows, fewer I/O operations are needed. More data can be processed in memory before writing back to disk, improving efficiency.

Increasing time with larger page sizes: As pages become very large, memory management becomes more complex, cache utilization may decrease and thus the system might need to swap data more frequently if pages exceed available memory, all which increase the processing time.

Optimisations

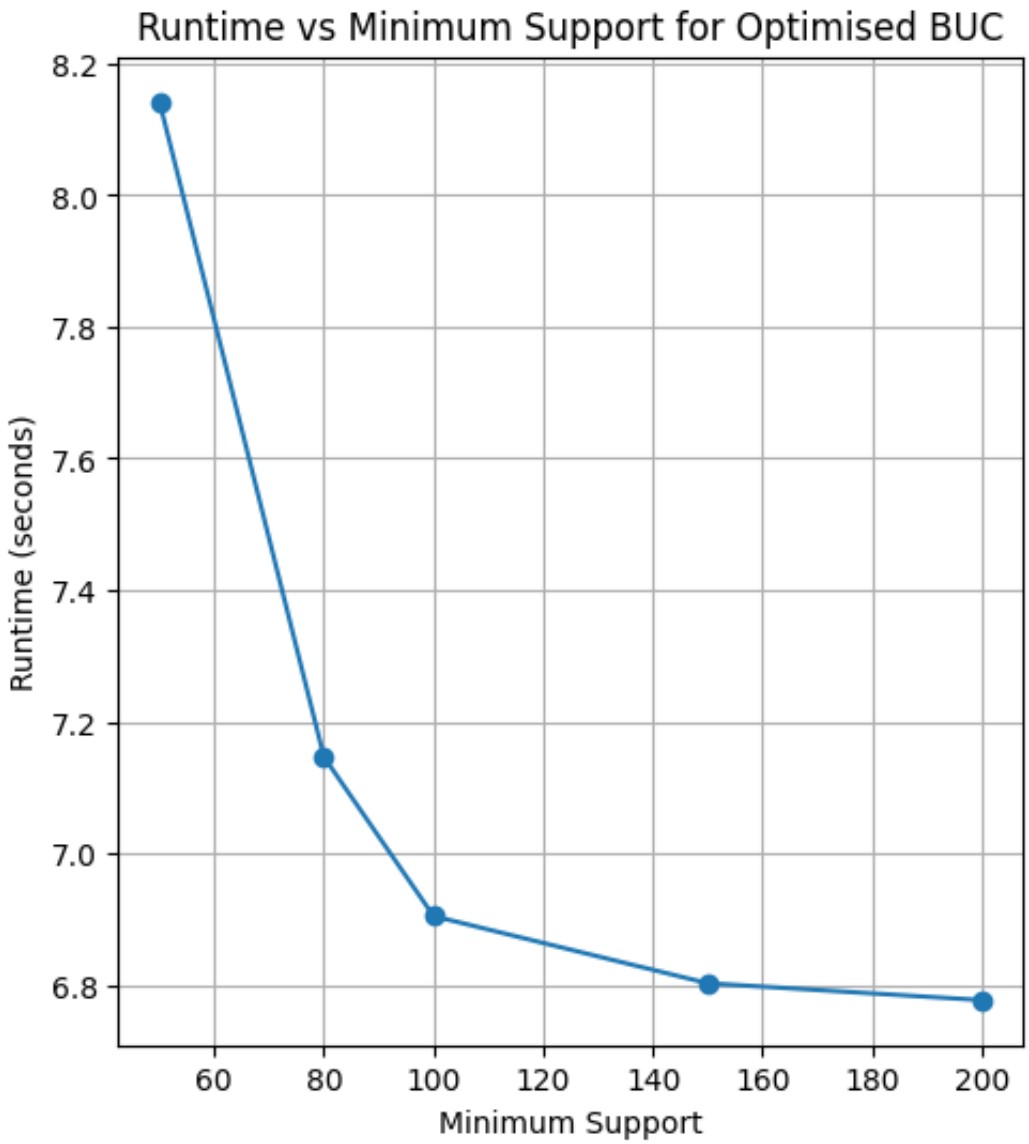
We have added two optimisations to our BUC implementation: caching and multiprocessing.

1. **Caching**:
    - A partition cache (`self.partition_cache`) is introduced to store the results of partitioning operations for subsets of the input data. This helps to avoid recalculating groupings of data that have already been processed, reducing redundant computations.
2. **Multiprocessing**:
    - Tasks generated during the BUC process are parallelized using Python's `multiprocessing.Pool`. The tasks, which involve processing subsets of the data, are distributed across multiple CPU cores. Each core runs the `process_task` function, which in turn invokes `run_buc` on smaller partitions of the data.

Impact on Performance

- **Reduced Redundancy**: By caching partition results, the algorithm avoids repeating expensive partitioning operations, leading to faster execution, especially for datasets with overlapping data subsets.
- **Parallel Processing**: Leveraging multiprocessing enables concurrent processing of independent tasks, which significantly reduces overall runtime,

especially for large datasets with many potential partitions.



As seen in the graph, the optimizations have made the runtime significantly shorter compared to a typical BUC implementation, which had much higher runtimes due to repeated computations and single-threaded execution.

Comparison between BUC and AOI

a. Primary Purposes and Use Cases

- **BUC**: Used for multidimensional data cube computation in OLAP tasks, generating aggregates across dimensions for complex analytical queries. It's ideal for applications like sales analysis and market trends.
- **AOI**: Simplifies data by generalizing attributes to extract characteristic rules for knowledge discovery. It’s best suited for conceptual pattern discovery like customer behavior analysis.

b. Types of Insights

- **BUC**: Best suited for discovering aggregate patterns such as summarized metrics, average sales per product category, or trends across regions. The insights it provides are multidimensional aggregates, revealing hidden correlations across various dimensions.
- **AOI**: Uncovers generalized patterns and conceptual hierarchies. It is ideal for finding characteristic rules which provide summarized knowledge rather than numerical aggregates, offering insights about typical behaviors and trends based on generalized data.

c. Computational Efficiency and Scalability

- **BUC**: Can become computationally expensive, especially with high-dimensional datasets and low minimum support thresholds. Its complexity grows as it explores all dimension combinations and computes aggregates for each. To enhance efficiency, BUC can incorporate techniques like pruning or multiprocessing, but it may still suffer from performance issues when scaling.
- **AOI**: Generally more efficient than BUC because it generalizes attributes early, reducing the dimensionality of the dataset. It can handle larger datasets by limiting detailed analysis to concept hierarchies. However, it may still face performance bottlenecks if the generalization process requires significant preprocessing or involves complex hierarchies.

d. Interpretability of Results

- **BUC**: The results produced are data cubes, which highly interpretable for data analysts familiar with OLAP concepts. The results consist of aggregates such as sums, averages, and counts, which are easy to interpret in a business

context. However, interpreting them without visualization tools can be challenging due to the volume of results.

- **AOI**: Produces highly interpretable characteristic rules. The rules are in the form of simple if-then statements, making them intuitive for business users and decision-makers. Since AOI focuses on generalization, its results are concise and easy to understand.

e. Preferable Scenarios

- **BUC**: More suitable when you need a comprehensive, multidimensional view of your data. If you're trying to analyze trends such as the Base MSRP of electric vehicles by City, State, and Electric Vehicle Type, BUC allows you to explore these combinations and produce granular results.
- **AOI**: Preferable when you want to extract general patterns or characteristic rules from the data, without focusing on detailed aggregations. This approach is especially useful when trying to summarize data at a higher level and provide easily interpretable rules. In scenarios where quick, actionable insights are needed, such as identifying the key characteristics of CAFV-eligible vehicles, AOI is more efficient.

Examples from implementation

- BUC provides detailed multidimensional aggregates, allowing us to analyze electric vehicle data from different angles, such as vehicle price trends by City or Model Year, and how Electric Range varies across States. It’s powerful when detailed insights are needed.
- AOI is better suited for generalized insights and rule extraction, making it ideal for identifying broad trends and patterns, such as how CAFV eligibility correlates with vehicle characteristics or how certain types of electric vehicles are distributed geographically. It produces results that are intuitive and easier for non-technical stakeholders to understand.