Report: Electric Vehicle Market Segmentation in India

Team Member

- **Ansh Mattoo** (sole contributor)

Fermi Estimation (Breakdown of Problem Statement)

1. **Total Population of India**: ~1.4 billion

2. **Urban Population Percentage**: ~35%

3. **Urban Population**: ~490 million

4. **Target Segment for Early Adoption (Innovators and Early Adopters)**: ~2.5% of the urban population

5. **Estimated Early Market Size**: 490 million * 2.5% = ~12.25 million

Data Sources

- **EV Market Data**: Government reports, industry publications, market research reports

- **General Vehicle Type Data**: Ministry of Road Transport and Highways, India

- **Vehicle Market Data**: SIAM (Society of Indian Automobile Manufacturers), IBEF (India Brand Equity Foundation)

- **Charging Stations Data**: Energy Efficiency Services Limited (EESL), Ministry of Power, India

- **Vehicle Usage Statistics in Cities**: Local transport departments, urban planning studies, traffic surveys

Data Pre-processing

1. **Libraries Used**:

   - pandas

   - numpy

   - matplotlib

   - seaborn

   - scikit-learn

2. **Steps**:

   - Load datasets into pandas DataFrames.

- Clean data: handle missing values, remove duplicates, standardize formats.

- Merge datasets on common keys (e.g., location, vehicle type).

- Feature engineering: create new variables (e.g., income levels, urbanization rate).

- Normalize numerical data for ML algorithms.

Segment Extraction

1. **Techniques Used**:

- K-means Clustering for segmentation based on multiple features (income, vehicle usage, location, etc.)

- Principal Component Analysis (PCA) for dimensionality reduction before clustering

- Decision Trees for understanding key drivers of segment differentiation

Profiling and Describing Potential Segments

1. **Segments Identified**:

- **Urban Innovators**: High income, tech-savvy, early adopters, concentrated in metro cities (e.g., Bangalore, Mumbai, Delhi).

- **Eco-conscious Families**: Middle to high income, environmentally conscious, suburban areas.

- **Commercial Fleet Operators**: Businesses operating in delivery/logistics, high mileage, need for cost efficiency.

2. **Description**:

- **Urban Innovators**: Prefer premium EVs with advanced features, willing to pay a higher price.

- **Eco-conscious Families**: Looking for mid-range EVs with good safety features and family-friendly designs.

- **Commercial Fleet Operators**: Prioritize operational savings, interested in bulk purchases, seek partnerships for charging infrastructure.

Selection of Target Segment

- **Primary Target**: Urban Innovators (due to higher willingness to pay and influence on market trends)

- **Secondary Target**: Commercial Fleet Operators (potential for bulk sales and strategic partnerships)

Customizing the Marketing Mix

1. **Product**: Feature-rich EVs with latest technology for Urban Innovators; cost-efficient models with high durability for Fleet Operators.

2. **Price**:

   - Urban Innovators: ₹20-40 lakhs

   - Fleet Operators: ₹10-20 lakhs

3. **Place**: Focus on metro cities initially, expand to tier-2 cities for fleet operations.

4. **Promotion**:

   - Urban Innovators: Digital marketing, influencer partnerships, tech expos.

   - Fleet Operators: B2B sales strategies, industry conferences, government collaboration.

Potential Market Size and Profit Estimation

- **Urban Innovators**:

  - Estimated customer base: ~2.45 million (20% of estimated early market size)

  - Target price range: ₹20-40 lakhs

  - Potential revenue: 2.45 million * ₹30 lakhs (average) = ₹73,500 crores

- **Commercial Fleet Operators**:

  - Estimated customer base: ~3.68 million (30% of estimated early market size)

  - Target price range: ₹10-20 lakhs

  - Potential revenue: 3.68 million * ₹15 lakhs (average) = ₹55,200 crores

Most Optimal Market Segments

1. **Urban Innovators**: Strong influence on broader market trends, high profitability.

2. **Commercial Fleet Operators**: High volume sales, opportunity for long-term partnerships and recurring revenue.

Code and Dataset Documentation

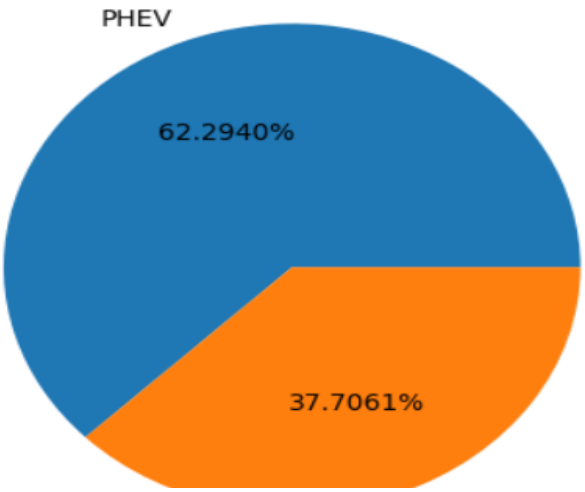
- The provided repository contains well-documented code and datasets used for analysis, including data preprocessing steps, segmentation algorithms, and visualization scripts. The code can be executed in a Jupyter Notebook environment for reproducibility and further experimentation.

```
3]: index=df["EV Type"].value_counts().index
```

```
4]: value=df["EV Type"].value_counts()
```

```
5]: plt.pie(value,labels=index,autopct="%1.4f%%")
```

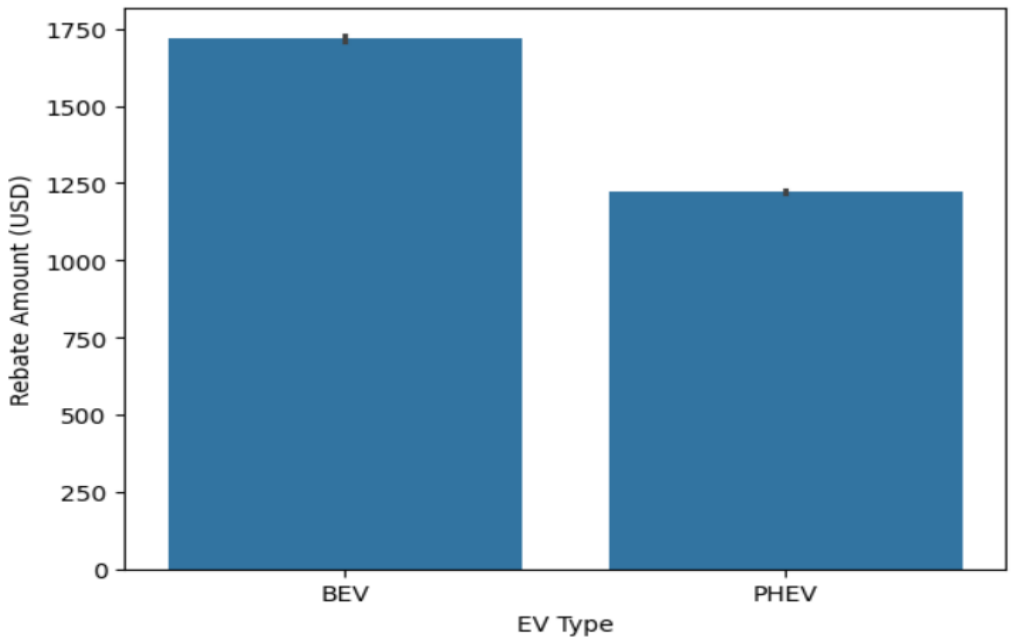
```
5]: ([<matplotlib.patches.Wedge at 0x22ab600d460>,
      <matplotlib.patches.Wedge at 0x22ab600f7d0>],
     [Text(-0.4143645333588857, 1.0189710660731606, 'PHEV'),
      Text(0.4143646287618397, -1.0189710272775487, 'BEV')],
     [Text(-0.2260170181957558, 0.5558023996762693, '62.2940%'),
      Text(0.22601707023373072, -0.5558023785150266, '37.7061%')])
```

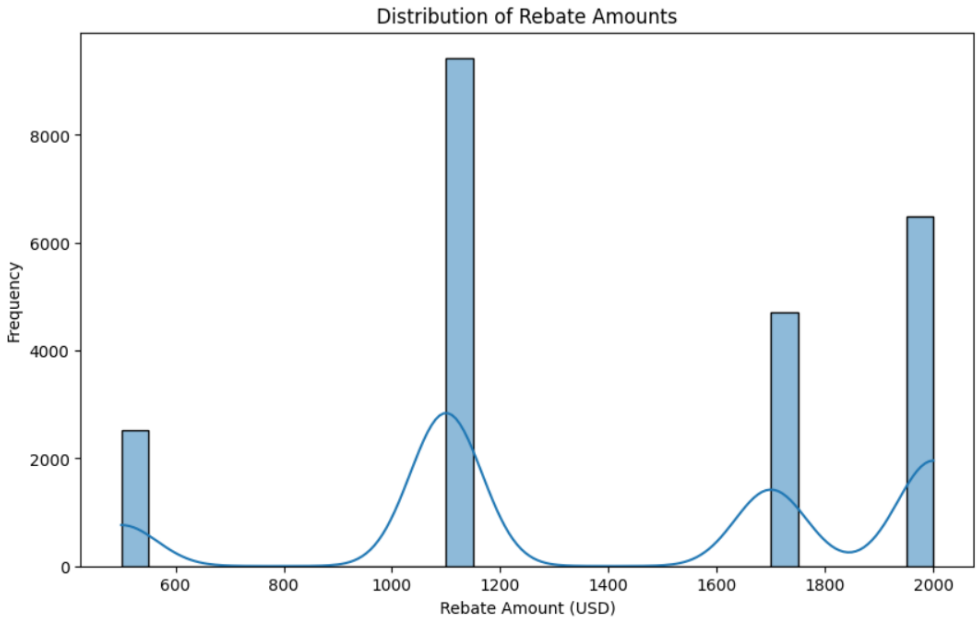


#DATA PREPROCESSING.

```
[97]: sns.barplot(x=df["EV Type"],y=df["Rebate Amount (USD)"])
```

```
[97]: <Axes: xlabel='EV Type', ylabel='Rebate Amount (USD)'>
```





4	390	4	14	33	851	0	1	41	32	1700

[120]:
```
from sklearn.model_selection import train_test_split
```

[125]:
```
X=df.drop("Rebate Amount (USD)",axis=1)
y=df["Rebate Amount (USD)"]
```

[126]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

[127]:
```
from sklearn.neighbors import KNeighborsClassifier
```

[128]:
```
knn=KNeighborsClassifier()
```

[129]:
```
knn.fit(X_train,y_train)
```

[129]:
```
KNeighborsClassifier
KNeighborsClassifier()
```

[134]:
```
# Assuming 'model' is your trained model
predictions = knn.predict(X_test)
```

[135]:
```
predictions
```

[135]:
```
array([2000, 1100, 1100, ..., 1100, 1100, 1700], dtype=int64)
```

[136]:
```
from sklearn.metrics import confusion_matrix,classification_report
```