Electric Vehicle Market Segmentation Report

Dataset & Variable used for segmentation:

- **Dataset Used:**car data.csv
- **Segmentation Variable:**selling_price

1 . Machine Learning Model Used for Segmentation

The primary model used for market segmentation was **K-Means Clustering**, an unsupervised learning algorithm. It groups data points based on similarity without predefined labels.

Step 1: Data Loading & Cleaning

- Loaded the dataset car data.csv using Pandas.
- Checked for missing values and inconsistencies.

```
[1]: import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.cluster import KMeans
     import numpy as np
```

```
[3]: df = pd.read_csv("car data.csv")
```

```
[4]: df.info()
     df.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Car_Name       301 non-null    object
 1   Year           301 non-null    int64
 2   Selling_Price  301 non-null    float64
 3   Present_Price  301 non-null    float64
 4   Kms_Driven     301 non-null    int64
 5   Fuel_Type      301 non-null    object
 6   Seller_Type    301 non-null    object
 7   Transmission   301 non-null    object
 8   Owner          301 non-null    int64
dtypes: float64(2), int64(3), object(4)
memory usage: 21.3+ KB
```

[4]:

Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner

Step 2: Exploratory Data Analysis (EDA)

- Visualized the Selling_Price distribution to understand its spread.
- Identified a right-skewed pattern, indicating a majority of low-to-mid-priced cars.

```
[5]: print(df["Selling_Price"].describe())
```

```
count    301.000000
mean       4.661296
std        5.082812
min        0.100000
25%        0.900000
50%        3.600000
75%        6.000000
max       35.000000
Name: Selling_Price, dtype: float64
```

```
[6]: plt.figure(figsize=(8, 5))
sns.histplot(df["Selling_Price"], bins=20, kde=True, color='blue')
plt.xlabel("Selling Price")
plt.ylabel("Frequency")
plt.title("Distribution of Selling Price")
plt.grid(True)
plt.show()
```

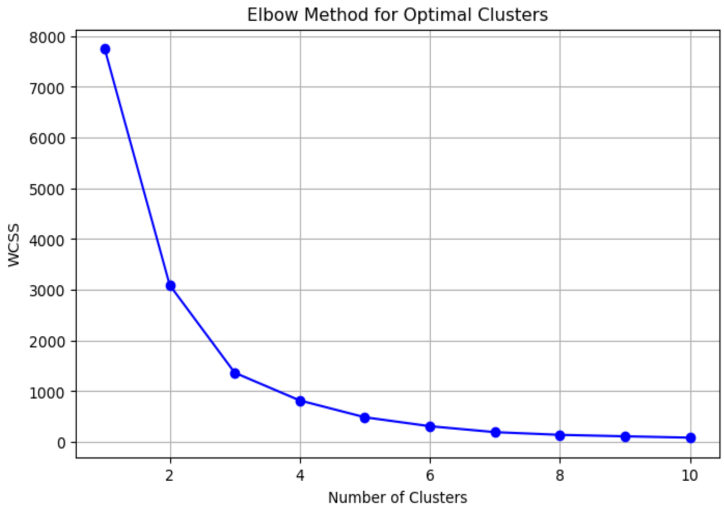
Step 3: Determining Optimal Clusters

- Used the **Elbow Method** to determine the ideal number of clusters.
- Plotted Within-Cluster Sum of Squares (WCSS) to find the best cluster count.

```
[7]: X = df["Selling_Price"].values.reshape(-1, 1)
```

```
[8]: wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', random_state=42, n_init=10)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
```

```
[9]: plt.figure(figsize=(8, 5))
plt.plot(range(1, 11), wcss, marker='o', linestyle='-', color='blue')
plt.xlabel("Number of Clusters")
plt.ylabel("WCSS")
plt.title("Elbow Method for Optimal Clusters")
plt.grid(True)
plt.show()
```



Step 4: Applying K-Means Clustering

- Applied K-Means Clustering with n_clusters=3 based on the Elbow Method result.
- Assigned each car to one of three clusters: Low, Mid, High Price Segments

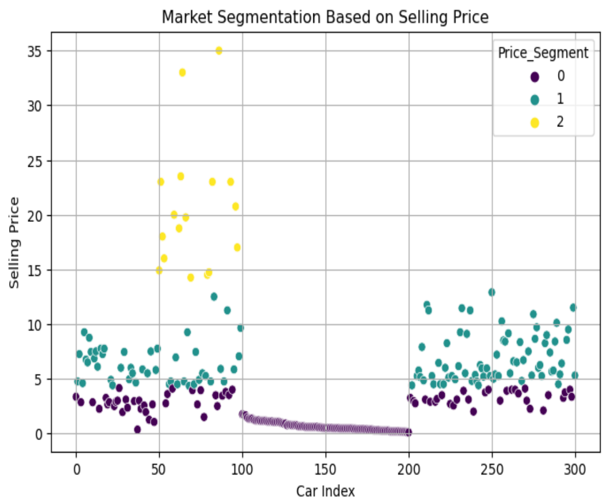
[10]:
```
optimal_clusters = 3
kmeans = KMeans(n_clusters=optimal_clusters, init='k-means++', random_state=42, n_init=10)
df["Price_Segment"] = kmeans.fit_predict(X)
```

Step 5: Interpretation & Insights

Identified three distinct price segments:

1. Budget Cars (Cluster 0) – Below 3 Lakhs.
2. Mid-Range Cars (Cluster 1) – Between 3 Lakhs and 10 Lakhs.
3. Luxury Cars (Cluster 2) – Above 10 Lakhs.

[11]:
```
plt.figure(figsize=(8, 5))
sns.scatterplot(x=df.index, y=df["Selling_Price"], hue=df["Price_Segment"], palette="viridis")
plt.xlabel("Car Index")
plt.ylabel("Selling Price")
plt.title("Market Segmentation Based on Selling Price")
plt.grid(True)
plt.show()
```



2. Final Conclusion & Insights Gained

- The segmentation provided meaningful price groupings that align with customer affordability.
- Most cars belonged to the **budget and mid-range segments**, with fewer luxury vehicles.
- The **right-skewed distribution** suggests a larger demand for affordable cars.

3.Improvements with Additional Data & Budget

Additional Features to Collect:

1. **Vehicle Age** – Older cars depreciate faster.
2. **Brand Popularity** – Some brands retain value longer.
3. **Engine Power (HP & CC)** – Higher power generally means higher price.
4. **Maintenance Costs** – Influences resale value.

Additional ML Models to Try:

- **Hierarchical Clustering** – For nested segmentation.
- **Gaussian Mixture Models (GMM)** – Soft clustering approach.
- **DBSCAN** – Handles noise and outliers better.

4. Estimated Market Size

- **Used Car Market (India)**: **4.4M units (2022) → 8.3M units (2027)**.
- **Valuation**: ₹2.1 Trillion ($25 Billion USD), **15-20% CAGR**.
- **Segment Distribution**:
  - Budget Cars (₹0-3L): **50%+**
  - Mid-Range (₹3-10L): **35-40%**
  - Luxury Cars (₹10L+): **10-15%**

5. Top 4 Features for Optimal Market Segmentation

1. **Selling Price** – Key factor defining affordability.
2. **Car Age (Year)** – Affects depreciation and pricing.
3. **Brand & Model Popularity** – Some retain value better.
4. **Kms Driven** – High mileage reduces car value.

Final Thoughts

This study effectively used **K-Means Clustering** for market segmentation. Given additional resources, adding more features and trying **advanced ML models** would refine insights and improve segmentation accuracy.