Machine Learning for EV Startup Market Segmentation

Nakul Bisen

17Th February, 2024

Abstract: *This project utilizes machine learning to segment the electric vehicle (EV) market for a startup. By applying clustering and predictive modeling techniques to diverse data sources including demographics, geography, and behaviour, distinct customer segments are identified. Emphasizing interpretability, the approach provides actionable insights for tailored marketing campaigns and product strategies. This enables the startup to effectively target and retain customers in the competitive EV market.*

GitHub Link: https://github.com/nakul2070

1. Introduction:

Background and context of the problem:

- The electric vehicle (EV) market is booming due to rising environmental concerns, government incentives, and technological advancements. However, increased competition necessitates differentiation for EV startups. Traditional demographic-based segmentation methods are inadequate for this diverse market. Hence, there's a rising demand for sophisticated segmentation strategies like machine learning.
- Machine learning analyzes vast datasets to reveal subtle patterns and relationships, allowing startups to identify distinct market segments based on demographics, psychographics, and behavior. This enables tailored marketing campaigns and product offerings. Moreover, the interpretability of machine learning models provides actionable insights, aiding in strategy development.
- Overall, applying machine learning for EV startup market segmentation provides a competitive edge by accurately identifying customer segments, enabling strategic positioning, effective engagement, and sustainable growth amidst fierce competition.

Objectives of the project:

- Identify Optimal EV Type: Determine the ideal electric vehicle (EV) model for launch by analyzing market trends, consumer preferences, and technological feasibility.
- Segment Customer Base: Utilize machine learning to identify distinct customer segments within the EV market based on demographics, behavior, and psychographics.

2. Data Collection and Preprocessing:

Data collection process & Description of the dataset used

To kickstart our EV startup's market segmentation analysis for the upcoming launch in India, I began by focusing on data acquisition. This involved extensive research across multiple online sources to gather pertinent and suitable data for our project. This thorough data gathering process forms the foundation for the next pivotal phase: pinpointing the most lucrative segment to ensure a successful entry into India's dynamic and burgeoning EV market.

Resources used for research:

- https://www.kaggle.com/
- https://data.gov.in/
- https://cea.nic.in/electric-vehicle-charging-reports/?lang=en
- https://dataspace.mobi/dataset/electric-vehicle-charging-station-list
- https://www.statista.com/statistics/1264923/india-electric-passenger-vehicle-sales-by-manufacturers/
- https://vahan.parivahan.gov.in/vahan4dashboard/vahan/view/reportview.xhtml
- https://datasetsearch.research.google.com/

https://www.kaggle.com/datasets/saketpradhan/electric-vehicle-charging-stations-in-india

- This dataset comprises information on charging stations operational across India. It encompasses data on states, cities, and precise addresses of each charging station, along with latitude and longitude coordinates pinpointing their locations.
- Additionally, the dataset provides insights into the types of charging stations available, offering a comprehensive overview of the infrastructure supporting electric vehicle adoption throughout the country.
- Such detailed information is invaluable for stakeholders in the electric vehicle industry, aiding in strategic planning, infrastructure development, and market analysis to facilitate the transition towards sustainable mobility in India.

Data preprocessing steps

- **Datatype change**

In [6]:
```
ev_df['lattitude'] = ev_df['lattitude'].astype('float64')
```

In [7]:
```
ev_df.info(verbose=1)
```

- **Standardizing Column values**

In [9]:
```
# Function to capitalize only the first letter of each word and convert all others to lowercase
def capitalize_first_letter(word):
    return ' '.join([w.capitalize() for w in word.split()])

# Apply the function to the 'state' column
ev_df['state'] = ev_df['state'].apply(capitalize_first_letter)
```

- **Null Values handling**

In [20]:
```
# Drop rows with null values in 'longitude', 'latitude', and 'type' columns
ev_df = ev_df.dropna(subset=['longitude', 'lattitude', 'type'])


# Fill null values in 'address' column with combined 'city' and 'state'
ev_df['address'].fillna(ev_df['city'] + ', ' + ev_df['state'], inplace=True)
```

In [21]:
```
# Check for null values in each column again
null_values_per_column = ev_df.isnull().sum()
null_values_per_column
```

- **Duplicate entries handling**

In [23]:
```
# Check for duplicate rows
duplicate_rows = ev_df[ev_df.duplicated()]

print(duplicate_rows , len(duplicate_rows))
```

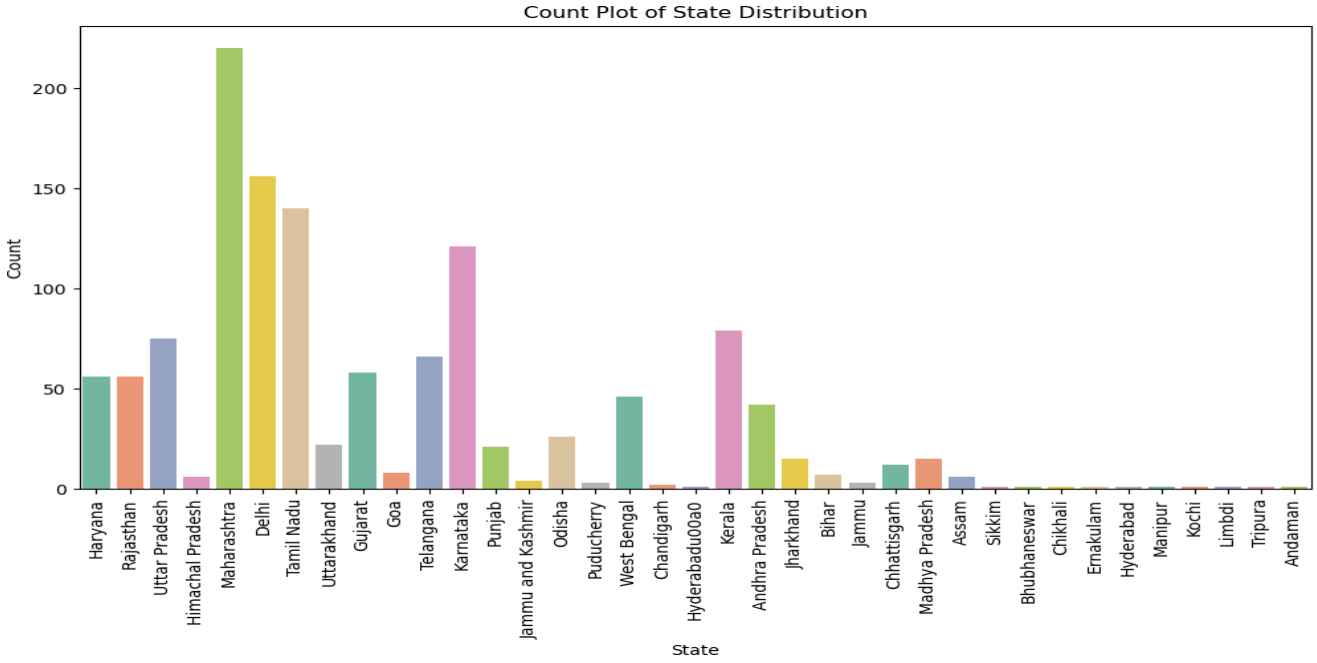
In [24]:
```
# Remove duplicate rows
ev_df = ev_df.drop_duplicates()
ev_df.shape
```

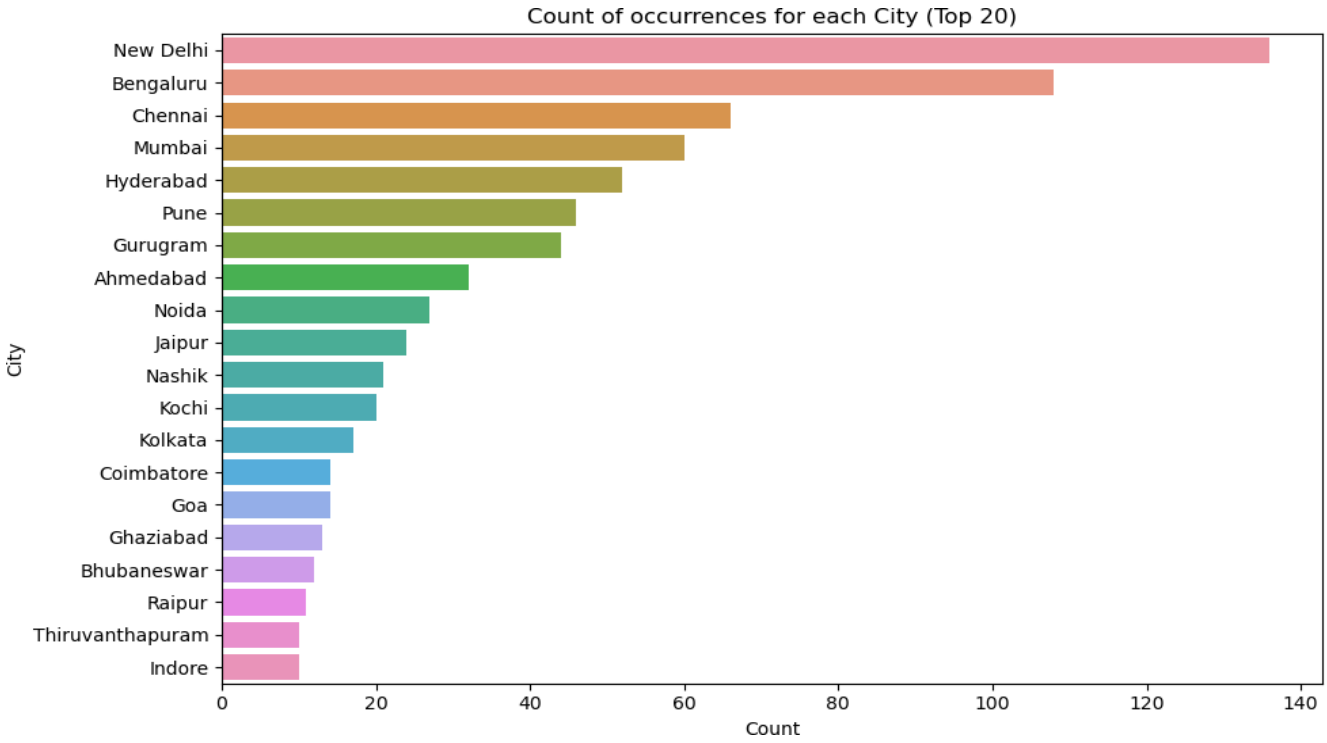
- **Outliers handling**

In [26]:
```
# Calculate the first and third quartiles
Q1_latitude = ev_df['lattitude'].quantile(0.03)
Q3_latitude = ev_df['lattitude'].quantile(0.97)
Q1_longitude = ev_df['longitude'].quantile(0.03)
Q3_longitude = ev_df['longitude'].quantile(0.97)

# Calculate the IQR for latitude and longitude
IQR_latitude = Q3_latitude - Q1_latitude
IQR_longitude = Q3_longitude - Q1_longitude

# Define the outlier step as 1.5 times the IQR
outlier_step_latitude = 1.5 * IQR_latitude
outlier_step_longitude = 1.5 * IQR_longitude

# Filter rows where latitude and longitude values are within the acceptable range
filtered_ev_df = ev_df[
    (ev_df['lattitude'] >= Q1_latitude - outlier_step_latitude) &
    (ev_df['lattitude'] <= Q3_latitude + outlier_step_latitude) &
    (ev_df['longitude'] >= Q1_longitude - outlier_step_longitude) &
    (ev_df['longitude'] <= Q3_longitude + outlier_step_longitude)
]
```

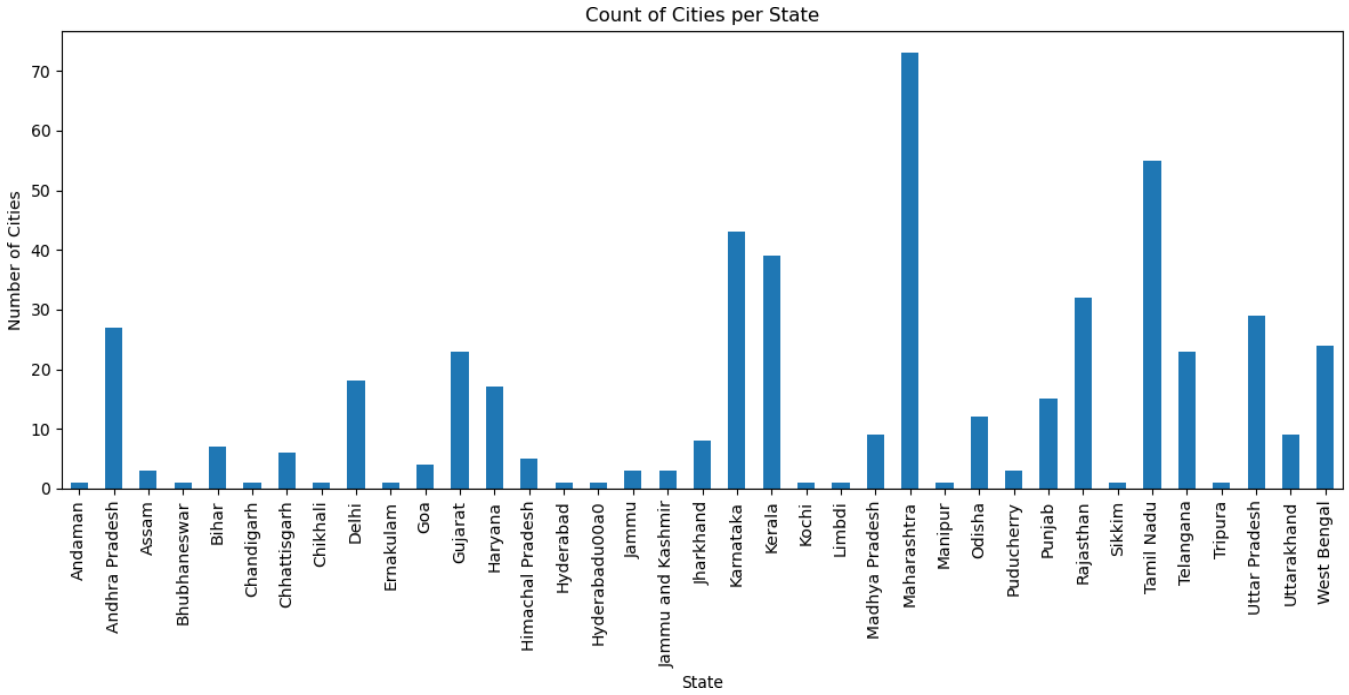
In [27]:
```
ev_df = filtered_ev_df
```

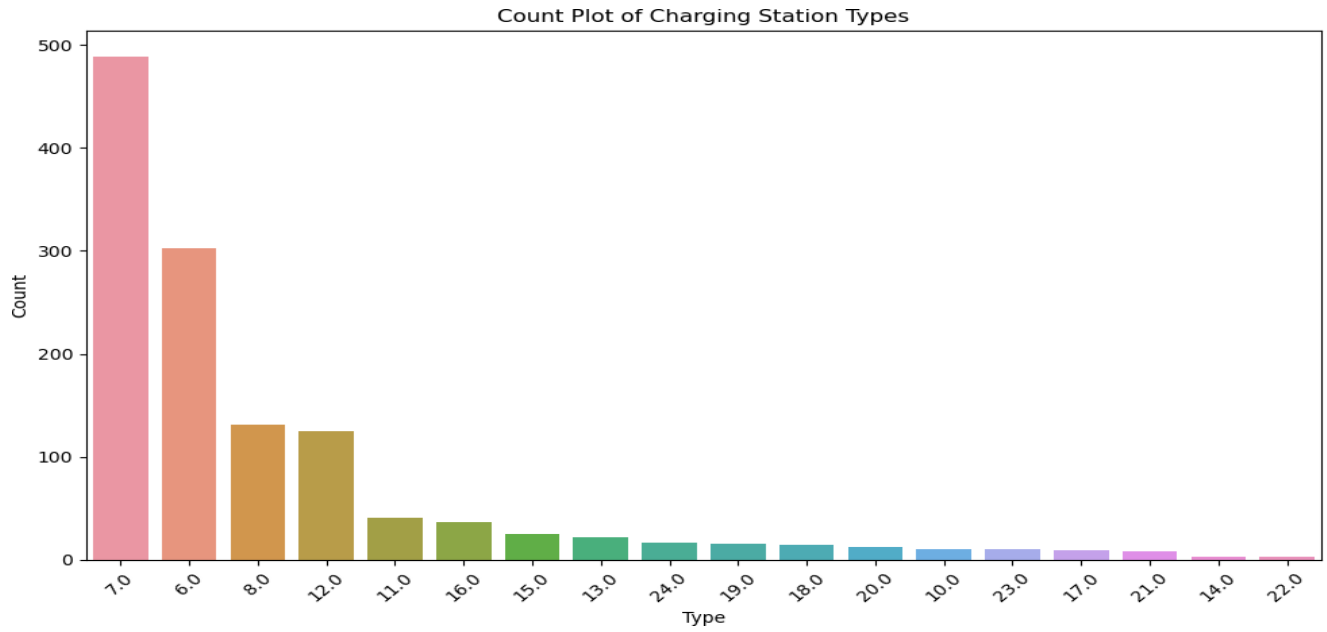
3. Exploratory Data Analysis (EDA):



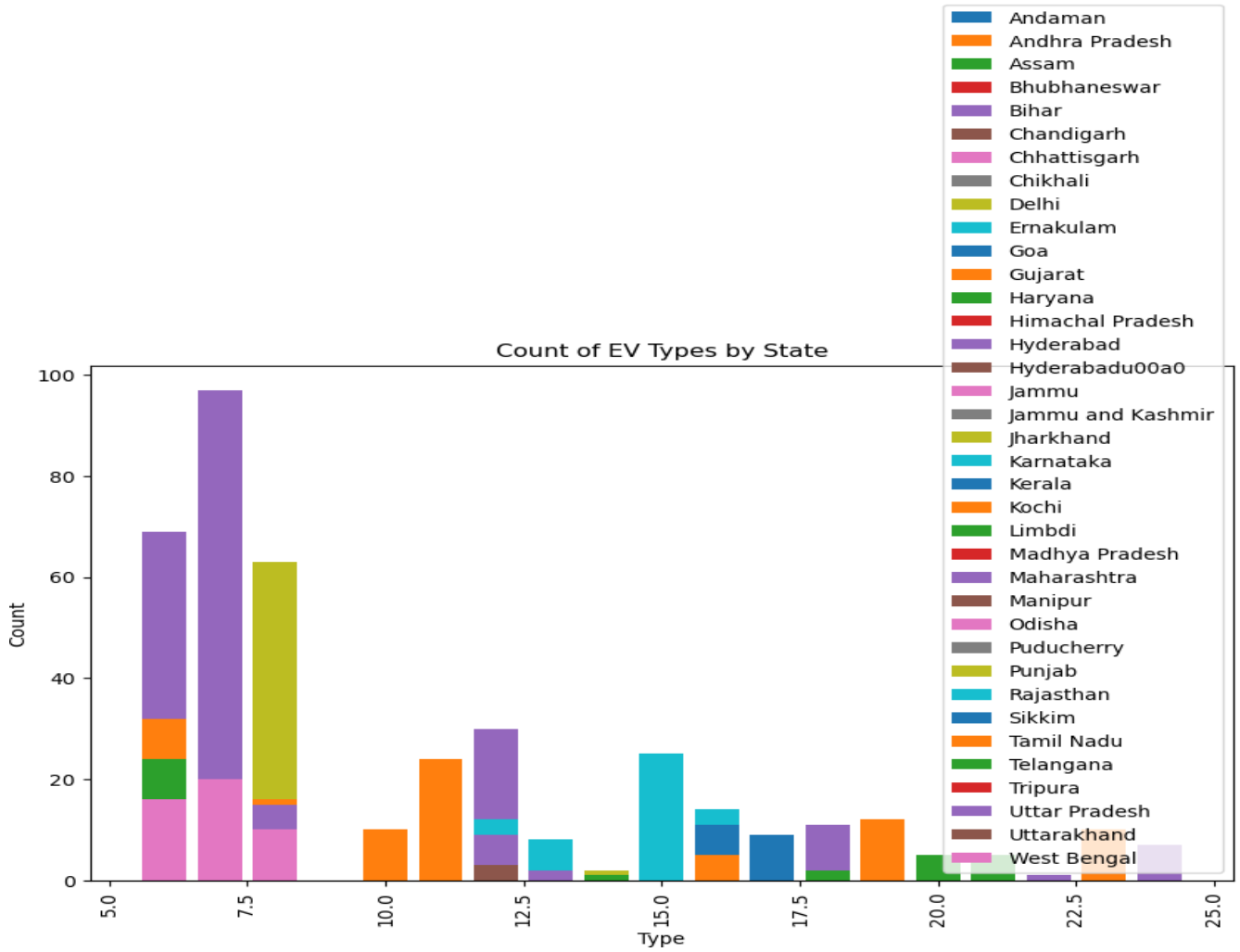


- Upon analysis of the state column, it's evident that Maharashtra boasts the highest count of EV charging stations, followed by Delhi.
- Tamil Nadu and Karnataka also exhibit notable numbers of EV charging stations, albeit slightly lower than Maharashtra and Delhi.
- Half of all charging stations are concentrated within the states of Maharashtra, Delhi, Tamil Nadu, Karnataka, and Kerala in India.
- In terms of cities, New Delhi exhibits the highest number of EV charging stations, followed by Bengaluru, Chennai, Mumbai, and Hyderabad, in descending order of station counts.





- High prevalence of 7 kWh EV charging stations.
- significant numbers also observed for 6, 8, and 12 kWh variants
- In Maharashtra, the majority of charging stations offer 6 and 7 kilowatt-hour (kWh) charging capacity.
- Charging stations in Gujarat, Tamil Nadu, and Kerala predominantly provide charging capacities of 10, 11, and 19 kWh.
- Karnataka and Rajasthan feature charging stations with a capacity of 15 kWh.
- Telangana and Haryana are equipped with charging stations offering 20 and 21 kWh charging capacities.



4. Methodology & Modeling:

Machine learning algorithms used:

In [42]:

```
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score

# Creating a list to store silhouette scores
silhouette_scores = []

# Creating a list to store inertia values
inertia_values = []

# Defining the range of clusters
k_range = range(2, 8)

# Iterate through each number of clusters
for k in k_range:
    # Initialize KMeans with k clusters
    kmeans = KMeans(n_clusters=k, random_state=42)

    # Fit KMeans to the data
    kmeans.fit(ev_df)

    # Calculate silhouette score
    silhouette_avg = silhouette_score(ev_df, kmeans.labels_)
    silhouette_scores.append(silhouette_avg)

    # Append the inertia to the list
    inertia_values.append(kmeans.inertia_)
```

Feature Engineering Techniques

- **One-hot Encoding**

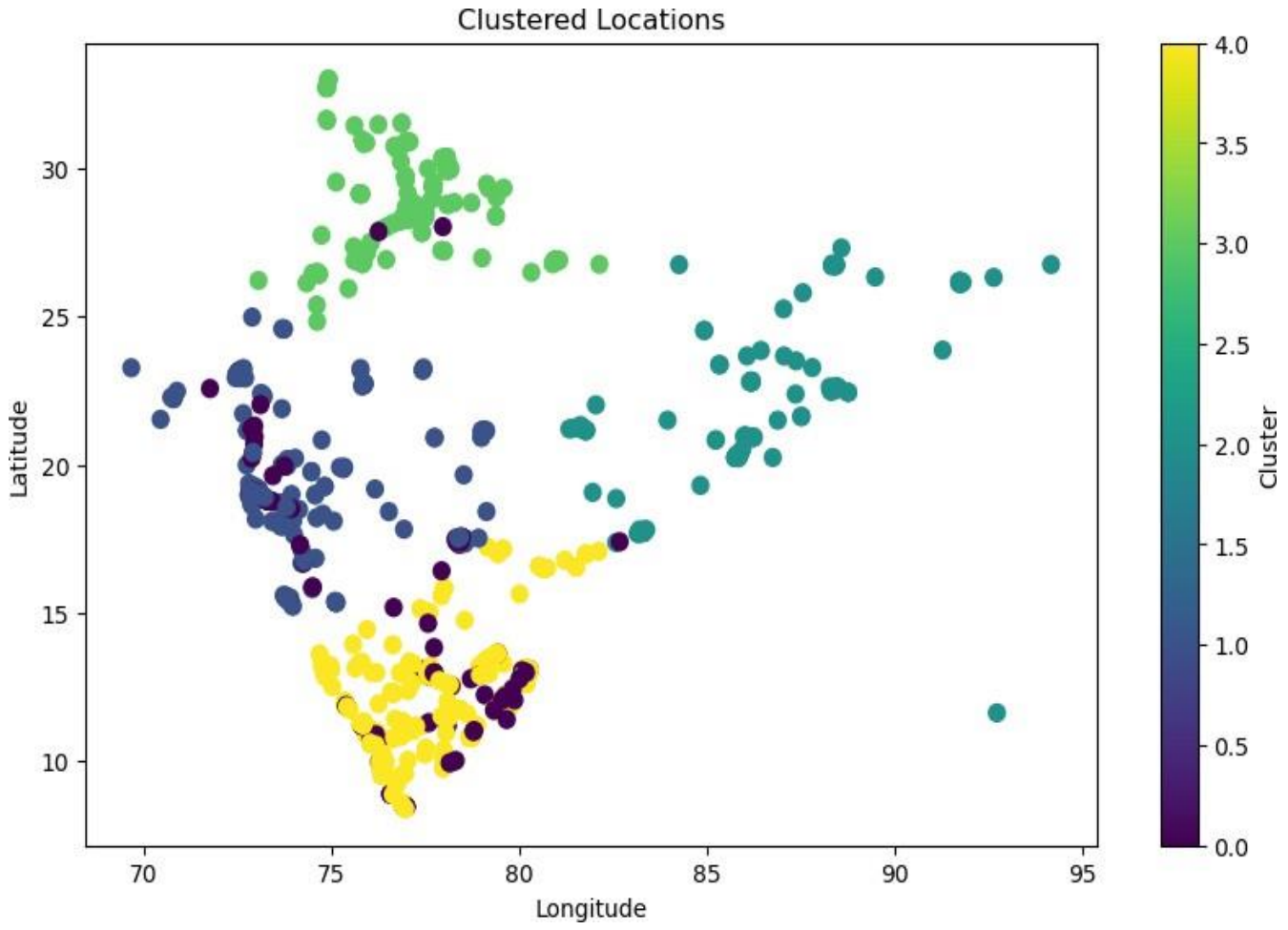
In [37]:
```
# Perform one-hot encoding for 'state'
state_dummies = pd.get_dummies(ev_df['state'], prefix='state', drop_first=True)

# Perform one-hot encoding for 'city'
city_dummies = pd.get_dummies(ev_df['city'], prefix='city', drop_first=True)

# Concatenate dummy variables with original dataframe
ev_df_encoded = pd.concat([ev_df, state_dummies, city_dummies], axis=1)

# Drop the original categorical columns
ev_df_encoded.drop(['state', 'city'], axis=1, inplace=True)
```

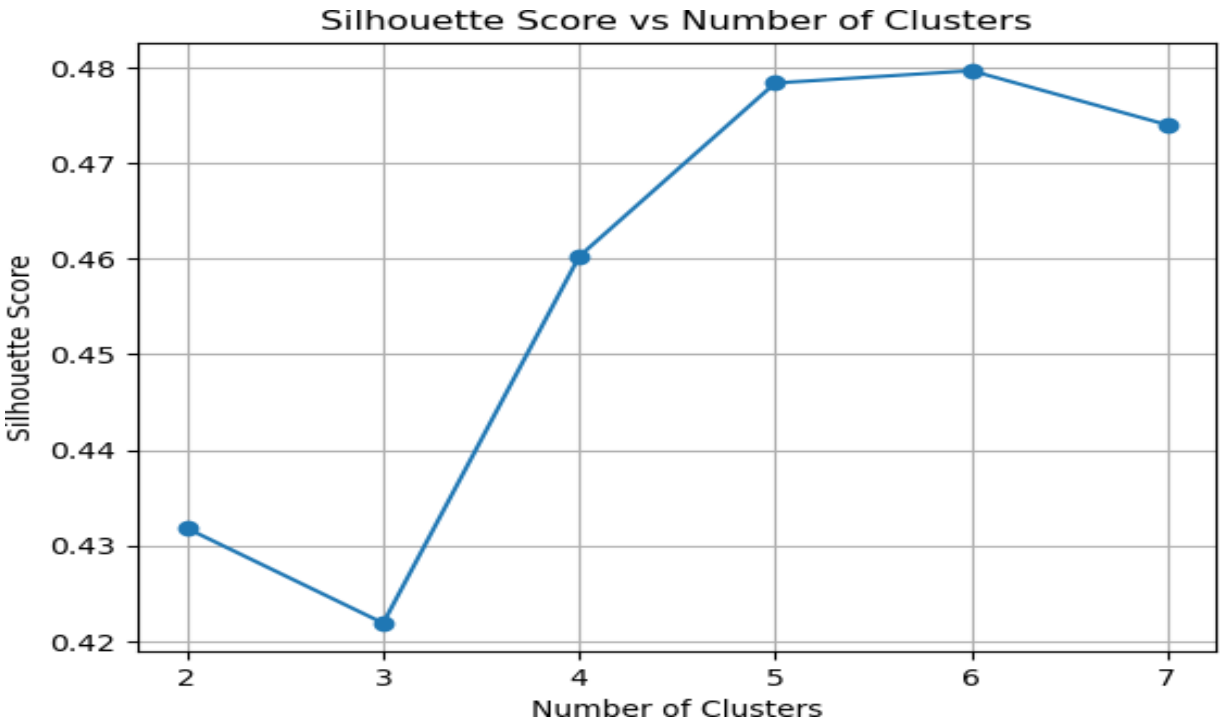
In [38]:
```
ev_df = ev_df_encoded.drop(['name', 'address', 'Charging Type'], axis=1)
```

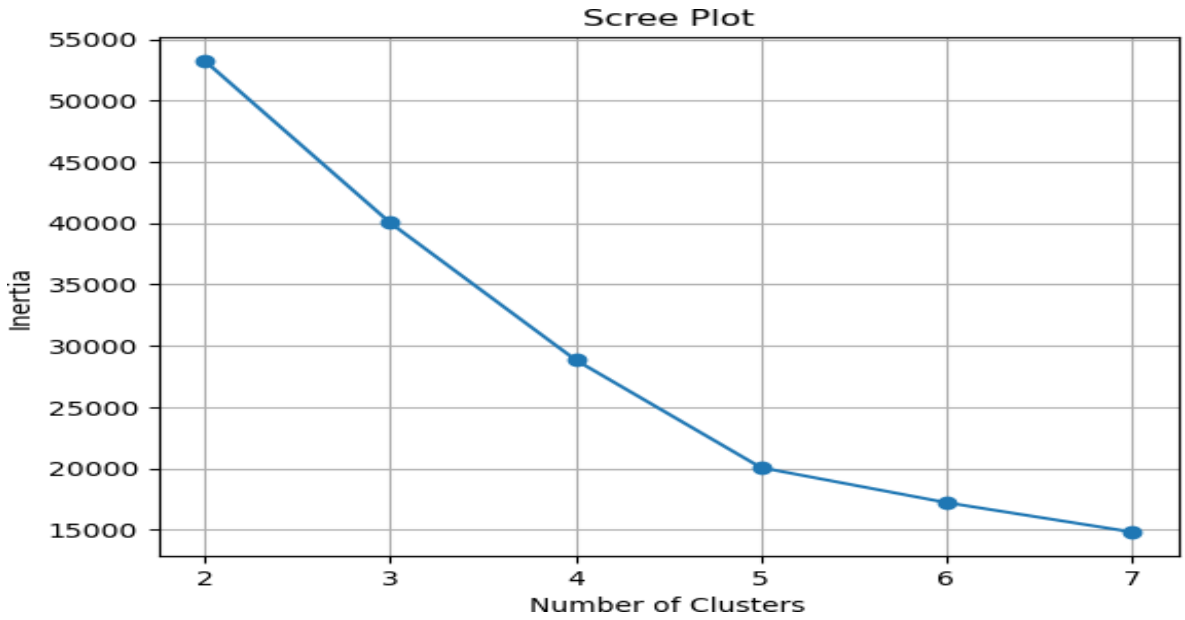
- **PCA**

In [39]:
```
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

# Standardize the numerical features
scaler = StandardScaler()
scaled_features = scaler.fit_transform(ev_df)

pca = PCA(n_components=300)
pca.fit(scaled_features)
explained_variance_ratio = pca.explained_variance_ratio_
cumulative_variance = explained_variance_ratio.cumsum()
```

Model evaluation metrics







- At around k=5, there's a noticeable and abrupt change in the rate at which the distortion score decreases. This indicates that increasing the number of clusters beyond k=5 doesn't result in a significant enhancement in the quality of clustering.
- According to the elbow method, the most suitable number of clusters for the dataset appears to be k=5. This implies that the KMeans algorithm can efficiently organize the data points into five separate clusters with minimal distortion.

5. Conclusion:

- Maharashtra leads in EV charging stations, followed closely by Delhi, Tamil Nadu, and Karnataka, collectively hosting half of India's stations.
- New Delhi tops city counts, followed by Bengaluru, Chennai, Mumbai, and Hyderabad.
- 7 kWh stations are prevalent, with significant numbers also for 6, 8, and 12 kWh variants.
- Clustering analysis reveals well-defined clusters, with five showing favorable scores, but the scree plot suggests the potential for more than five clusters.
- Overall, this analysis highlights regional distribution patterns and charging station capacities in India.

Market Segmentation and Targeting Strategy for an Electric Vehicle Startup in India

17Th February, 2024

1. Problem Statement:

An electric vehicle (EV) startup in India faces a crucial decision: who to target. The diverse Indian market offers numerous possibilities, but choosing the right segment is essential for success. This project tackles this challenge through market segmentation analysis. By understanding different customer groups (geographic, demographic, etc.), we will assess their needs and the competition. This will help us identify the "sweet spot": the segment most likely to embrace the startup's EVs. Considering data limitations and market dynamics, we will develop a targeted entry strategy, positioning the startup for long-term growth in the electrifying Indian market.

2. Data Collection:

To initialize the implementation of the market segmentation analysis for our EV startup's Indian launch, I started with the data acquisition efforts. Through meticulous research, I delved into various data sources available on the internet to gather appropriate and relevant data for the project. This comprehensive data collection exercise lays the groundwork for the next crucial step: identifying the most promising segment for our startup's successful entry into the electrifying Indian EV market.

Websites used for researching:

- https://www.kaggle.com/
- https://data.gov.in/
- https://datasetsearch.research.google.com/
- https://trends.google.com/trends/explore

The datasets I worked on for the project:

1. https://www.kaggle.com/datasets/prasenjitsharma/fuel-type-wise-vehicle-registration-india/data

The above dataset specifies the total no. of vehicles registered in India from January 2014- July, 2023. The has been categorized into fuel variant of the vehicle registered. Analyzing this data would give an idea about the purchasing trends in the Indian Market and how it has changed over the years.

2. https://pib.gov.in/Pressreleaseshare.aspx?PRID=1808115

This dataset specifies the sanctioned EV Charging Stations in India:

a. State-wise sanctioned EV Charging Stations
b. City-wise sanctioned EV Charging Stations
c. Sanctioned EV Charging Stations on Expressways and Highways

Analyzing this data would help in making proper decisions about which states/cities to target those have already established/planned supporting infrastructure for the electric vehicles.

Code Implementation:

Importing all the neccesarry libraries

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
import warnings
warnings.filterwarnings('ignore')
```

1. Fuel Type wise vehicle registration in India

Data Loading and Preprocessing

```
In [7]: print(pd.isnull(datasetFuelType).sum())
```

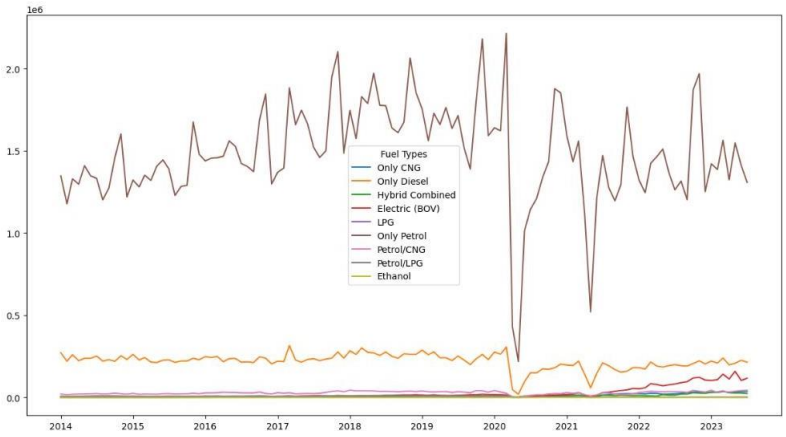
```
Month                0
CNG ONLY             0
DIESEL               0
DIESEL/HYBRID        0
DUAL DIESEL/CNG      0
ELECTRIC(BOV)        0
ETHANOL              0
LPG ONLY             0
NOT APPLICABLE       0
PETROL               0
PETROL/CNG           0
PETROL/ETHANOL       0
PETROL/HYBRID        0
PETROL/LPG           0
SOLAR                0
FUEL CELL HYDROGEN   0
LNG                  0
METHANOL             0
DUAL DIESEL/LNG      0
dtype: int64
```

```
In [8]: datasetFuelType["Month"]=pd.to_datetime(datasetFuelType["Month"], format='%b-%y')
        datasetFuelType["Month"].head()
```

```
Out[8]: 0   2014-01-01
        1   2014-02-01
        2   2014-03-01
        3   2014-04-01
        4   2014-05-01
        Name: Month, dtype: datetime64[ns]
```

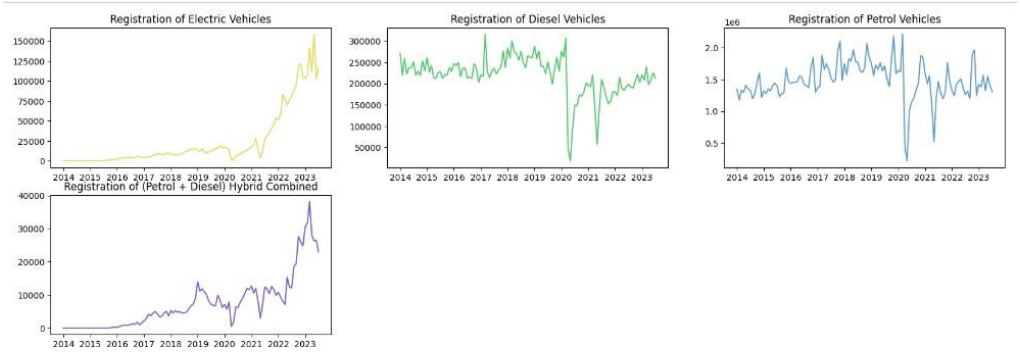
```
In [9]: datasetFuelType['HYBRID COMBINED'] = datasetFuelType['PETROL/HYBRID'] + datasetFuelType['DIESEL/HYBRID']

        # Dropping 'PETROL/HYBRID' and 'DIESEL/HYBRID' columns
        datasetFuelType.drop(['PETROL/HYBRID', 'DIESEL/HYBRID'], axis=1, inplace=True)
```

Exploring the Data

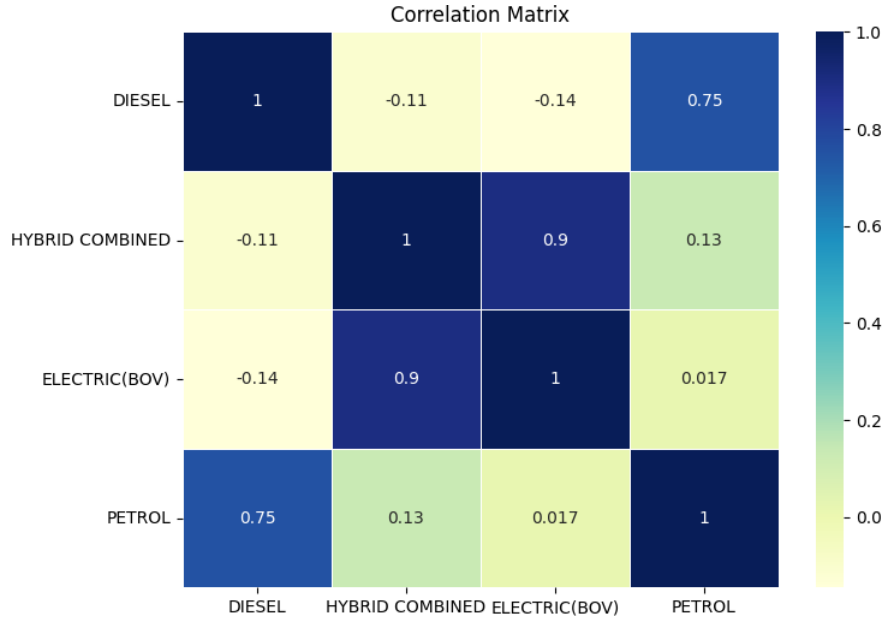


Based upon the graph, we can see that:

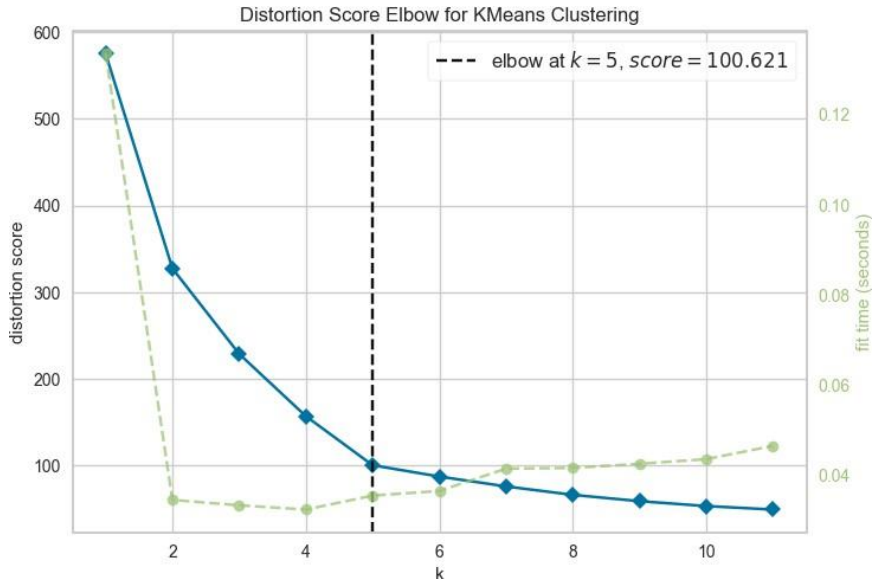
- Petrol vehicles consistently have the highest registrations throughout the time period, indicating their dominance in the market.
- Diesel vehicle registrations remain relatively stable over time, suggesting a consistent demand for diesel-powered vehicles.
- Electric vehicles show an increasing trend in registrations over time, indicating a growing interest or adoption of electric mobility, from late 2021 onwards.
- The increasing trend in EV registrations suggests a potential shift in the market toward cleaner and more sustainable transportation options.



- Out of all the four variants of fuel type, the registrations of Electric Vehicles and Hybrid Vehicles show a noticeable increasing trend over the time period and seems to suggest a growing market in the near future.
- The upward trend may indicate a shift in consumer preferences towards more sustainable and environmentally friendly transportation options.
- The consistent demand for the remaining vehicles suggests a stable market presence, possibly driven by specific industry requirements or consumer preferences.



- **Strong positive correlations** exist between "DIESEL" and "PETROL" (0.87), suggesting that these fuel types tend to move together in terms of their values.
- **Moderate positive correlation** exists between "HYBRID COMBINED" and "PETROL" (0.54) and "HYBRID COMBINED" and "ELECTRIC(BOV)" (0.44), indicating some degree of co-occurrence.
- **A weak positive correlation** exists between "Year" and "ELECTRIC(BOV)" (0.23), suggesting a possible increase in electric vehicles over time.
- **A strong negative correlation** exists between "ELECTRIC(BOV)" and "DIESEL" (-0.84), indicating that an increase in electric vehicles is associated with a decrease in diesel usage.

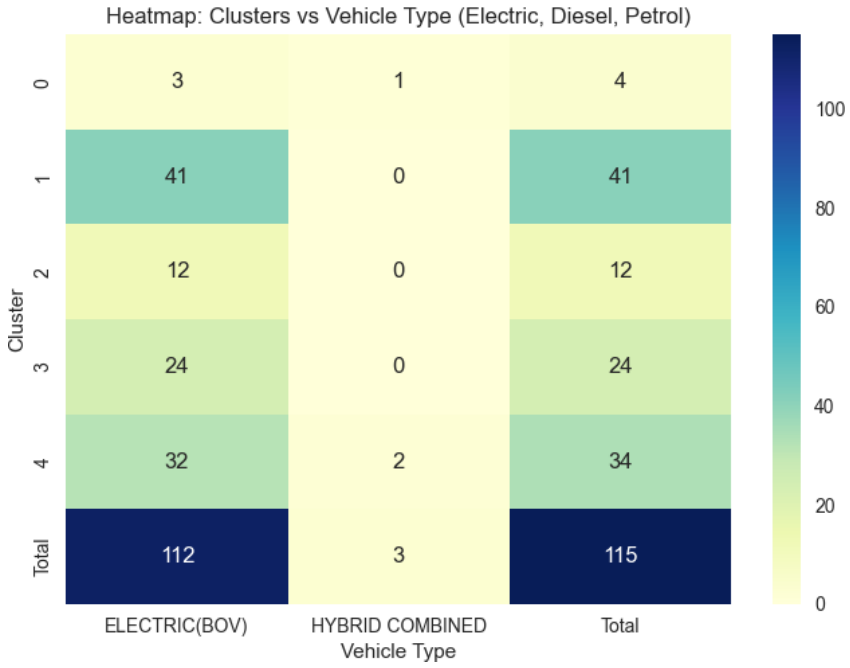


- There is a clear and sharp elbow around k=5, where the decrease in distortion score starts to slow down significantly. This suggests that adding more clusters beyond k=5 does not provide a substantial improvement in clustering quality.
- Based on the elbow method, the optimal number of clusters for the given data is likely **k=5.** This means that the KMeans algorithm can effectively group the data points into five distinct clusters with minimal distortion.

```
In [36]: # Applying K-means clustering
kmeans = KMeans(n_clusters=5)
finalDatasetFuelType['Cluster'] = kmeans.fit_predict(scaledData)

# Visualize or analyze the clusters
print(finalDatasetFuelType['Cluster'].value_counts())
```

```
1    41
4    34
3    24
2    12
0     4
Name: Cluster, dtype: int64
```

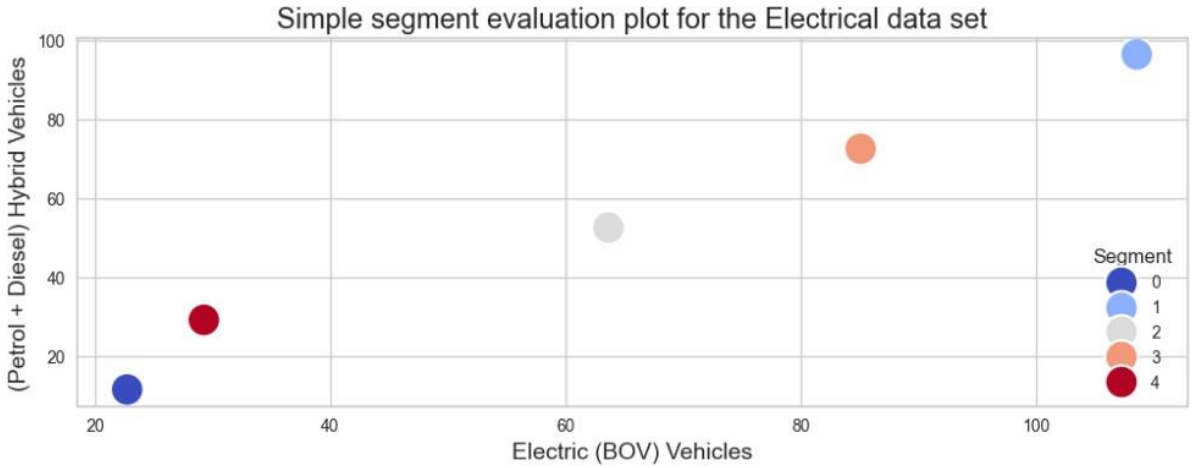
More registerations belong to **Cluster 1 & 4**



Based on the above heatmap, here are some observations:

- Cluster 0: This cluster is dominated by Electric vehicles, with a significantly higher count compared to Diesel and Petrol vehicles.

- Cluster 1: This cluster has a more balanced distribution of vehicle types, with Petrol vehicles having the highest count, followed by Electric and Diesel vehicles.
- Cluster 2: This cluster is primarily composed of Diesel vehicles, with a very low count of Electric and Petrol vehicles.
- Cluster 3: This cluster has a moderate count of Electric vehicles, followed by Diesel and Petrol vehicles.



Electric (BOV) Vehicles: This refers to Battery Operated Vehicles, which encompass pure electric cars and electric two-wheelers.

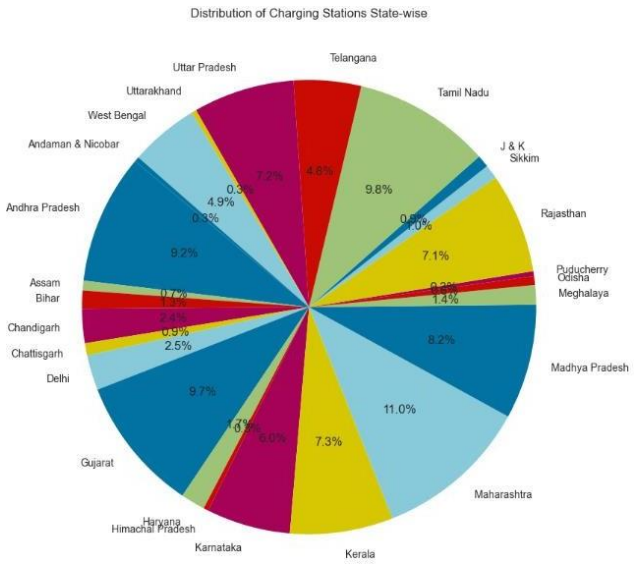
(Petrol + Diesel) Hybrid Vehicles: These are vehicles that combine an electric motor with a traditional gasoline or diesel engine.

- The data points are spread across all four quadrants of the graph, suggesting that there is a diversity in terms of both electric and hybrid vehicle adoption across the identified segments.
- There seems to be a concentration of points in the lower left and upper right quadrants. This could imply that some segments have a preference for either electric or hybrid vehicles, while others have a more balanced mix.

Target Segments:

- Target Segment 0: Prioritize marketing EVs' environmental benefits and lower running costs. Ensure easy access to charging infrastructure information.
- Segment 4: Address price concerns and highlight the flexibility of having both hybrid and electric options.
- Segments 1 and 5: Offer a diverse range of EV and hybrid options cater to various needs and budgets. Emphasize fuel efficiency and the evolving charging infrastructure landscape.
- Segments 2 and 3: Focus on the practicality and reliability of hybrids while acknowledging the growing appeal of EVs. Address range anxiety concerns.

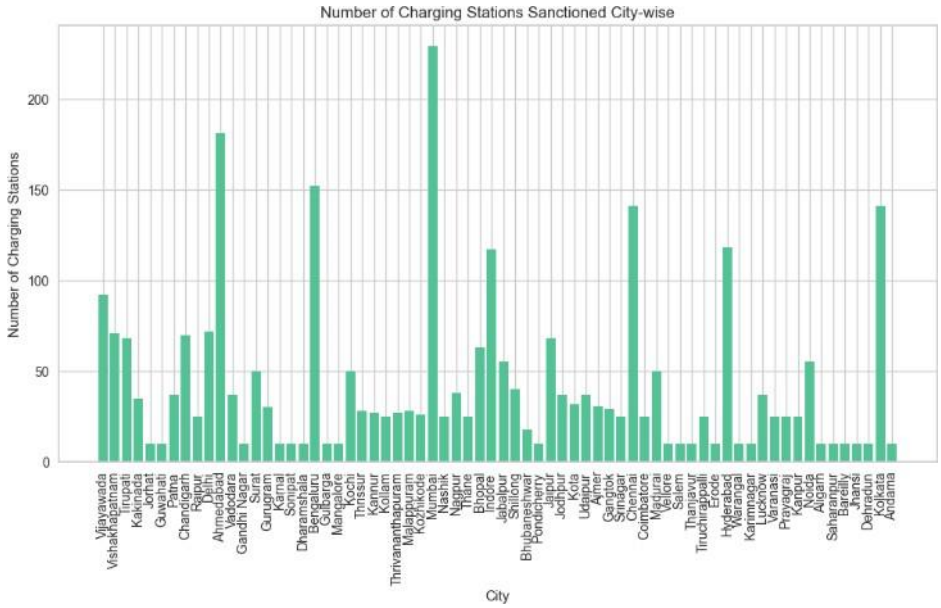
2. State-wise Charging Station Sanctioned



Based on the following graph we can see that the states with maximum number of sanctioned charging stations are:

1. Maharashtra
2. Tamil Nadu
3. Gujarat
4. Andhra Pradesh
5. Madhya Pradesh

3. City-wise Charging Station Sanctioned



Based on the following graph we can see that the cities with maximum number of sanctioned charging stations are:

1. Mumbai
2. Ahmedabad
3. Bengaluru
4. Kolkata
5. Chennai

Conclusion:

The Indian mobility landscape is undergoing a dynamic shift, fueled by rising environmental consciousness and evolving consumer preferences. While petrol retains dominance, electric vehicles (EVs) are experiencing a surge, particularly evident in states like Maharashtra, Tamil Nadu, and Gujarat, and major cities like Mumbai, Ahmedabad, and Bengaluru. This presents a lucrative opportunity for our EV startup to target two key segments:

1. Environmentally Conscious Early Adopters: Cluster 0, concentrated in these key regions, prioritizes sustainability and embraces EVs. Focus messaging on environmental benefits, address charging infrastructure concerns, and showcase innovative features.

2. Price-Conscious Hybrid-Open Consumers: Segment 4 represents potential converts open to both EVs and hybrids. Tailor messaging by emphasizing price competitiveness, flexibility, and fuel efficiency.

By focusing on these distinct segments and adapting marketing strategies accordingly, the startup can capitalize on the expanding Indian EV market, ensuring long-term success in this electrifying landscape.

- The complete code along with the dataset is available at **GitHub Repository**.

Market Segmentation of EV Vehicles in India

17Th February, 2024

Abstract - In the multifaceted landscape of the Electric Vehicle (EV) market, effective segmentation based on efficiency, seating capacity, and price is paramount to address the varied preferences of consumers. Efficiency, measured by factors such as range per charge and energy consumption, ensures that EV options cater to both the environmentally conscious urban commuter and those with longer travel needs, mitigating concerns related to range anxiety. Simultaneously, considering seating capacity accommodates diverse lifestyles, with compact EVs catering to smaller households or urban dwellers, while larger vehicles with extended seating capacity meet the requirements of families or larger groups.

Furthermore, price segmentation plays a pivotal role in shaping consumer choices. By offering EVs across different price ranges, manufacturers can tap into a broader market spectrum, providing entry-level options for cost-conscious consumers and premium models with advanced features for those seeking enhanced performance and luxury. This strategic segmentation approach not only facilitates a better understanding of consumer needs but also allows manufacturers to tailor their offerings, fostering widespread adoption and contributing to the overall success and growth of the dynamic EV industry.

Data Collection - The data was collected from the online statistics platform and has been fact checked and used multiple times by trusted users.

Data Description -

```
df = pd.read_csv("EV_Dataset.csv")
df.dropna(inplace=True)
df.shape
```

✓ 0.0s Python

```
(103, 15)
```

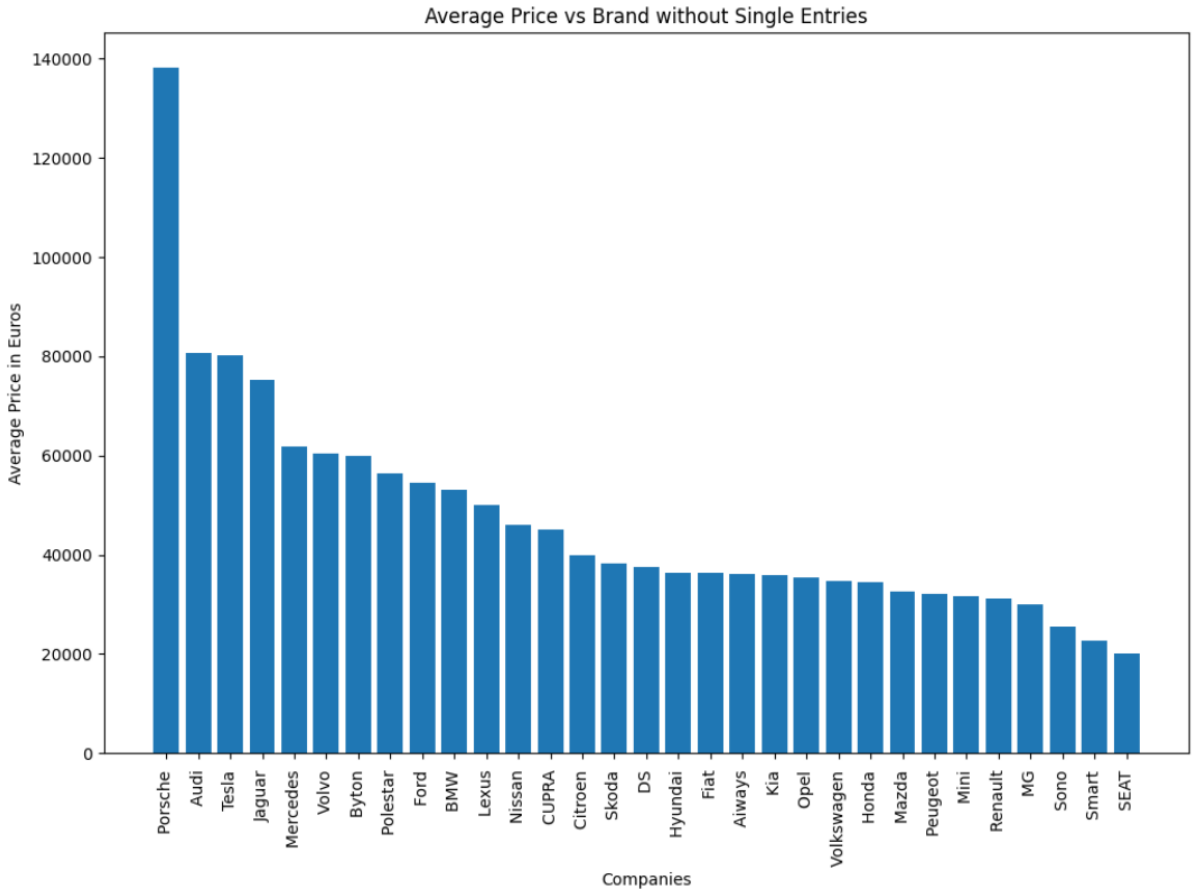
+ Code + Markdown

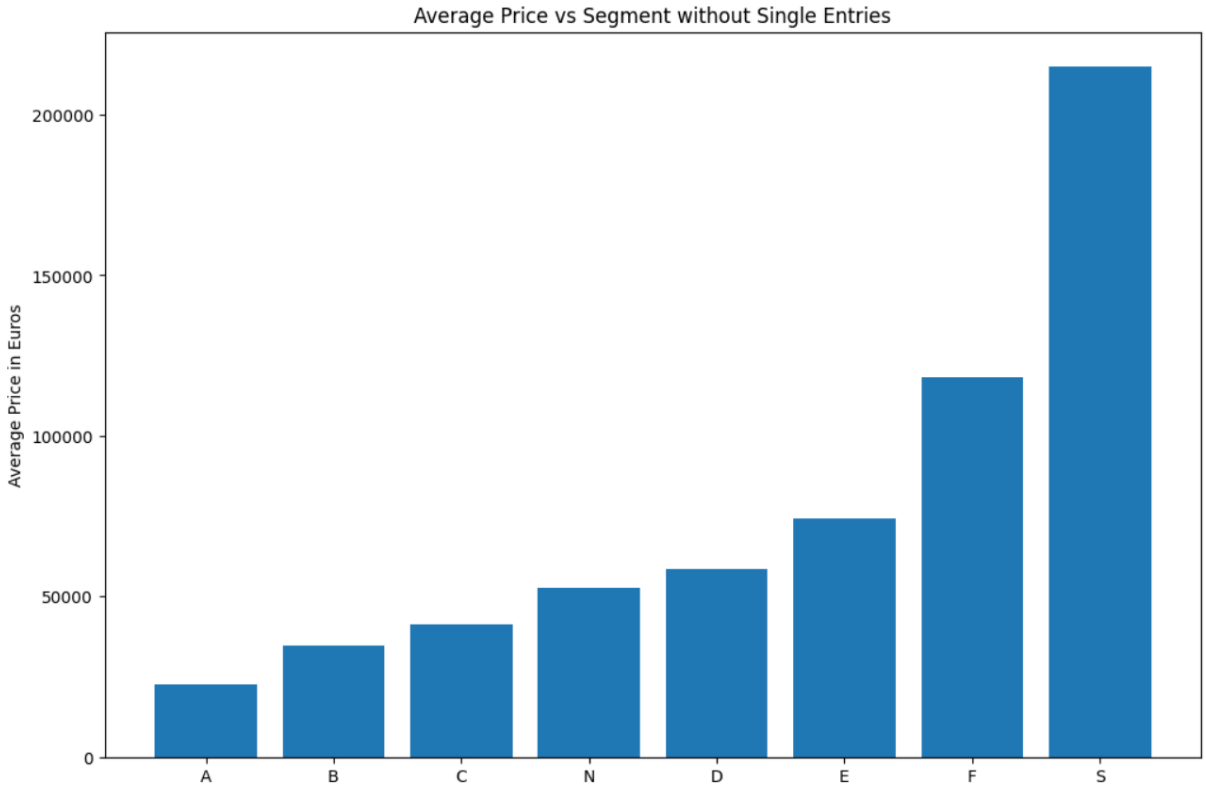
```
df.describe()
```

✓ 0.0s Python

	Unnamed: 0	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	Seats	PriceEuro
count	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000	101.000000
mean	51.485149	7.415842	178.782178	333.762376	190.099010	441.584158	4.881188	54401.891089
std	29.816644	3.001257	43.333037	121.976239	28.590734	204.986500	0.803581	32793.840327
min	0.000000	2.100000	123.000000	95.000000	153.000000	170.000000	2.000000	20129.000000
25%	26.000000	5.100000	150.000000	250.000000	168.000000	260.000000	5.000000	34400.000000
50%	52.000000	7.300000	160.000000	340.000000	181.000000	430.000000	5.000000	45000.000000
75%	77.000000	9.000000	200.000000	400.000000	206.000000	550.000000	5.000000	64000.000000
max	102.000000	22.400000	410.000000	970.000000	273.000000	940.000000	7.000000	215000.000000

Basic EDA - Performed Checking of Average prices of different brands and how they change with seat number, segments, Body Styles, etc. This gave us a idea of the trends that are to be expected from the data and the positive/negative correlation between the features and target variable.





Data Cleaning - Removed unwanted data rows with only one entry, removed unwanted data features for PCA and removed data nulls that could be present in the data set.

Data Cleaning

```
mod_df = df.drop(columns=["Brand","Model","RapidCharge","PowerTrain","PlugType","Segment"])
dummy_dic = {}
for index, name in enumerate(df.BodyStyle.unique()):
(variable) encoded_BodyStyle: list

encoded_BodyStyle = []
for i in df.BodyStyle:
    encoded_BodyStyle.append(dummy_dic[i])

mod_df['BodyStyle'] = encoded_BodyStyle
```

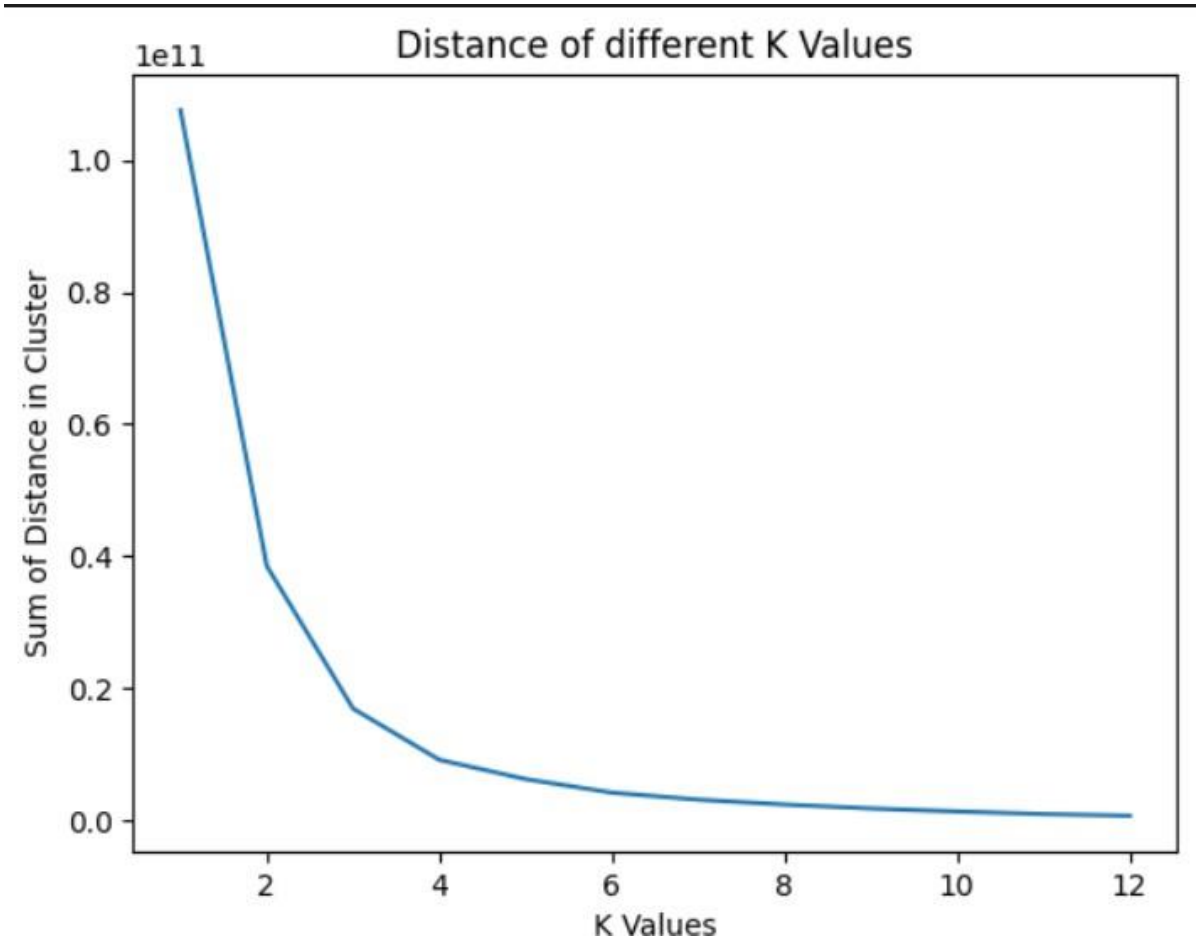
[133] ✓ 0.0s

```
mod_df.drop(columns=["Unnamed: 0"],axis= 1, inplace=True)
mod_df.head()
```

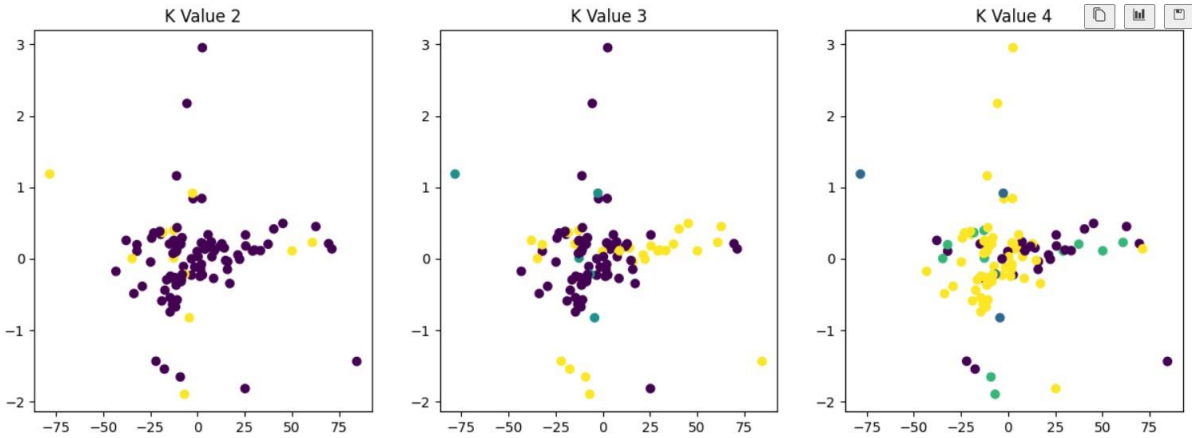
[135] ✓ 0.0s

	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	BodyStyle	Seats	PriceEuro
0	4.6	233	450	161	940	0	5	55480
1	10.0	160	270	167	250	1	5	30000
2	4.7	210	400	181	620	2	5	56440
3	6.8	180	360	206	560	3	5	68040
4	9.5	145	170	168	190	1	4	32997

K Means Clustering - Performed elbow test with different k values after taking the principal components of the data, Clusters of 4 were found to be the best in this case.



Performed visualizations of different clusters and how they develop clusters as the value of the clusters increase in number.



Model Development and Score - The data was divided into to 80 : 20 for training and testing respectively, the linear regression model and the KNN regressor both were tried out to check which model performs the best. The KNN was put through many values of k, in the end k = 2 worked the best with 0.83 score verses 0.78 of the linear regression.

K = 3 can be better if a larger set is taken accounting to lesser chance of over localisation of the value but it is upto testing.

Price Prediction

```
X_train, X_test, Y_train, Y_test = train_test_split(mod_df.drop(columns=["PriceEuro"]), mod_df.PriceEuro, test_size=0.2, random_state= 100)

knn_model_score = {}
k_reg_val = range(2,10)

for k in k_reg_val:
    knn_model = KNeighborsRegressor(n_neighbors=k)
    knn_model.fit(X_train,Y_train)
    knn_model_score[k] = (knn_model.score(X_test,Y_test))

knn_model_score
```

✓ 0.0s

```
{2: 0.8375493109894968,
 3: 0.7960343662021281,
 4: 0.688674677267638,
 5: 0.568994180443227,
 6: 0.5142981785425516,
 7: 0.5885400127798038,
 8: 0.5595977406705857,
 9: 0.5151705887723222}
```

```
lr_model = LinearRegression().fit(X_train,Y_train)
lr_model.score(X_test,Y_test)
```

✓ 0.0s

```
0.7887598101043176
```

Conclusions -

- 2 and 5 seat vehicles seem to be the best in the market segment when accounting for cost to passenger ration.
- Most EV companies are trying to target the 20 lakhs on the lower end and 35 to 40 lakhs on the upper range for normal affordable use of the vehicles.
- Efficency and Charging speed are key components of the price and desirablitiy
- There seems to 4 to 5 types of EV vehicles that are being developed at this current stage.

Market Segmentation Analysis Electric Vehicle Market In India

Date: 16/02/2024

***Abstract*---**In the realm of emerging transportation technologies like electric vehicles (EVs), market segmentation emerges as a pivotal strategy for facilitating widespread adoption, particularly in burgeoning markets. With the promise of reduced emissions and operational costs, EVs are poised for significant growth, prompting considerable academic interest. This study aims to delineate distinct buyer segments for EVs, utilizing a comprehensive research framework encompassing perceived benefits, attitudes, and intentions. Leveraging robust analytical techniques such as cluster analysis, multiple discriminant analysis, and Chi-square tests, the study scrutinized data from 563 respondents via a cross-sectional online survey. The results unveil three distinct consumer groups among potential EV buyers, namely 'Conservatives', 'Indifferent', and 'Enthusiasts', each exhibiting unique psychographic, behavioural, and socio-economic characteristics. These findings hold implications for scholars and policymakers seeking to promote EV adoption within the evolving landscape of sustainable transportation markets, offering valuable insights to inform strategic initiatives.

1. Data collection and Preprocessing:

Data Collection Process

The data collection process was conducted manually, sourcing information from various platforms including: Kaggle Datasets (https://www.kaggle.com/datasets)

Data- Preprocessing:

The collected data, condensed in format, serves dual purposes: visualization and clustering. The workflow relies on Python libraries like NumPy, Pandas, Scikit-Learn, and SciPy to process the data. Special attention is given to ensuring reproducibility in the obtained results, facilitating robustness and reliability in the analysis.

```
df = pd.read_csv('data.csv')
df.drop('Unnamed: 0', axis=1, inplace=True)
df['inr(10e3)'] = df['PriceEuro']*0.08320
df['RapidCharge'].replace(to_replace=['No','Yes'],value=[0, 1],inplace=True)
df.head()
```

	Brand	Model	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	PowerTrain	PlugType	BodyStyle	Segment	Seats	PriceEuro	inr(10e3)
0	Tesla	Model 3 Long Range Dual Motor	4.6000	233	450	161	940	1	AWD	Type 2 CCS	Sedan	D	5	55480	4615.9360
1	Volkswagen	ID.3 Pure	10.0000	160	270	167	250	0	RWD	Type 2 CCS	Hatchback	C	5	30000	2496.0000
2	Polestar	2	4.7000	210	400	181	620	1	AWD	Type 2 CCS	Liftback	D	5	56440	4695.8080
3	BMW	iX3	6.8000	180	360	206	560	1	RWD	Type 2 CCS	SUV	D	5	68040	5660.9280
4	Honda	e	9.5000	145	170	168	190	1	RWD	Type 2 CCS	Hatchback	B	4	32997	2745.3504

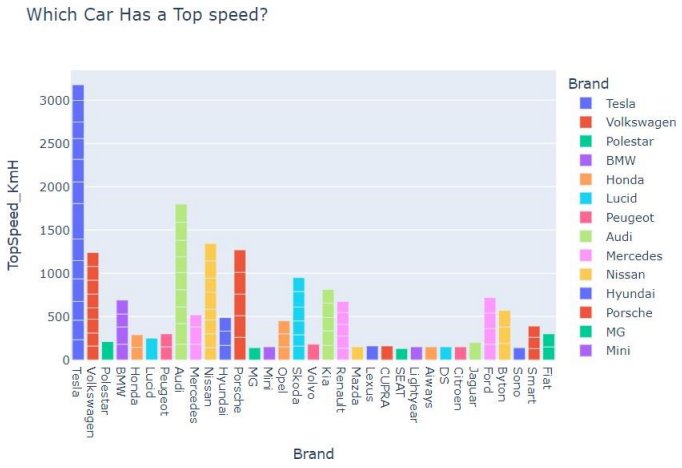
```
df.describe()
```

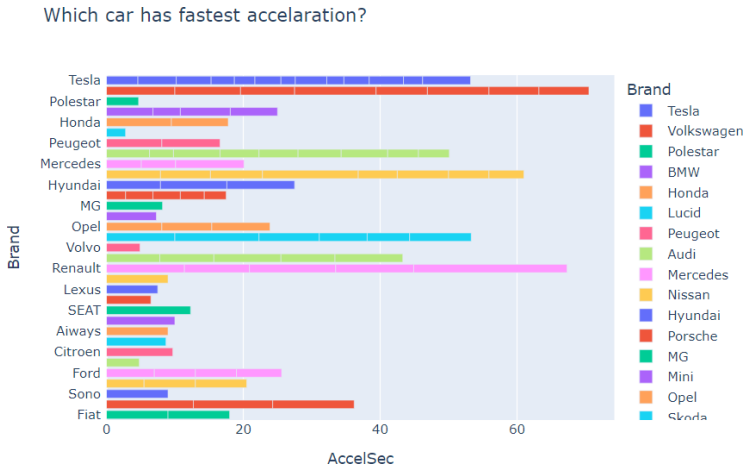
	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	Seats	PriceEuro	inr(10e3)
count	103.0000	103.0000	103.0000	103.0000	103.0000	103.0000	103.0000	103.0000	103.0000
mean	7.3961	179.1942	338.7864	189.1650	444.2718	0.7476	4.8835	55811.5631	4643.5221
std	3.0174	43.5730	126.0144	29.5668	203.9493	0.4365	0.7958	34134.6653	2840.0042
min	2.1000	123.0000	95.0000	104.0000	170.0000	0.0000	2.0000	20129.0000	1674.7328
25%	5.1000	150.0000	250.0000	168.0000	260.0000	0.5000	5.0000	34429.5000	2864.5344
50%	7.3000	160.0000	340.0000	180.0000	440.0000	1.0000	5.0000	45000.0000	3744.0000
75%	9.0000	200.0000	400.0000	203.0000	555.0000	1.0000	5.0000	65000.0000	5408.0000
max	22.4000	410.0000	970.0000	273.0000	940.0000	1.0000	7.0000	215000.0000	17888.0000

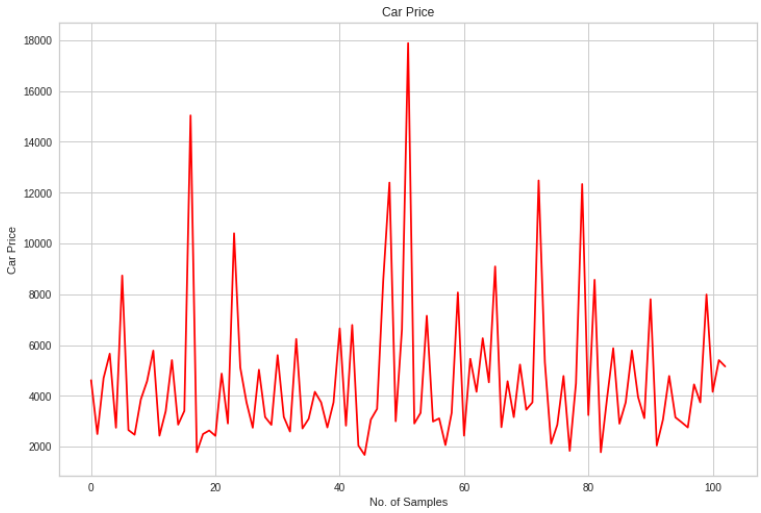
2. Exploratory Data Analysis (EDA):

Initiate the Exploratory Data Analysis (EDA) by conducting analysis on the dataset, both with and without Principal Component Analysis (PCA). PCA is a statistical technique used to transform correlated features into a set of linearly uncorrelated features through orthogonal transformation. These newly transformed features, known as Principal Components, aid in reducing the dimensionality of the data, thereby enhancing the cost-effectiveness of classification, regression, or any machine learning process.

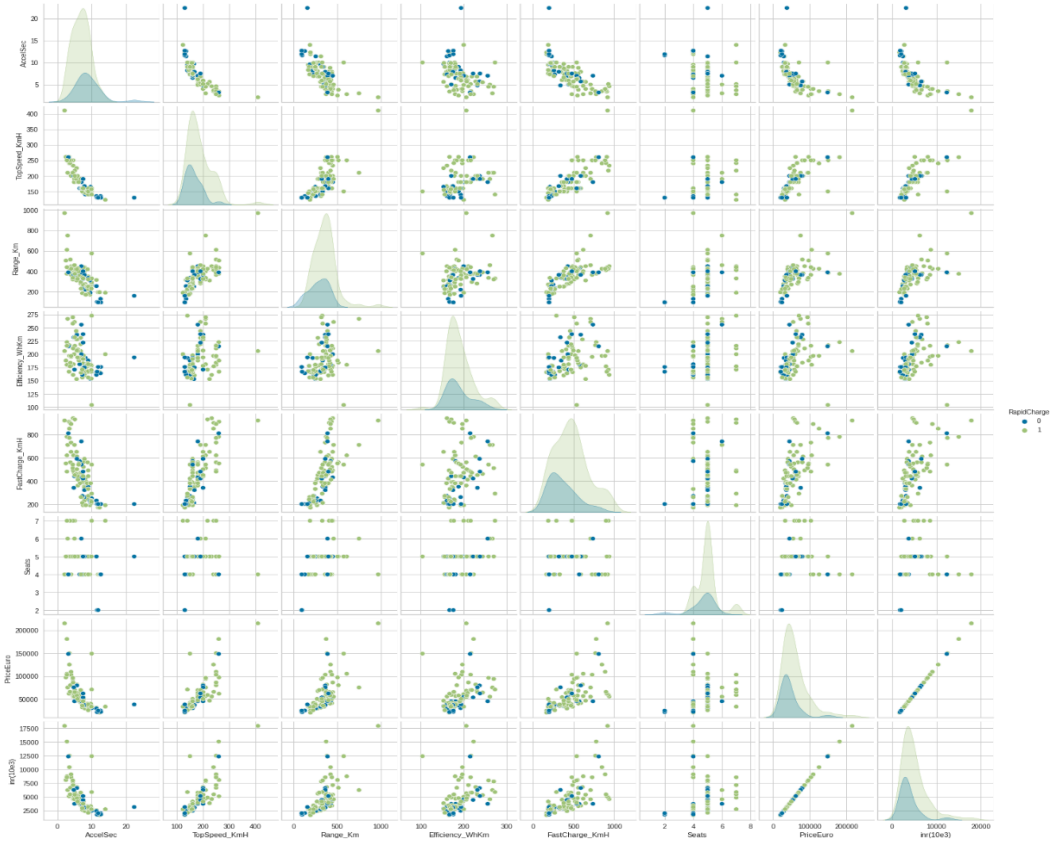
comparison of cars in our data





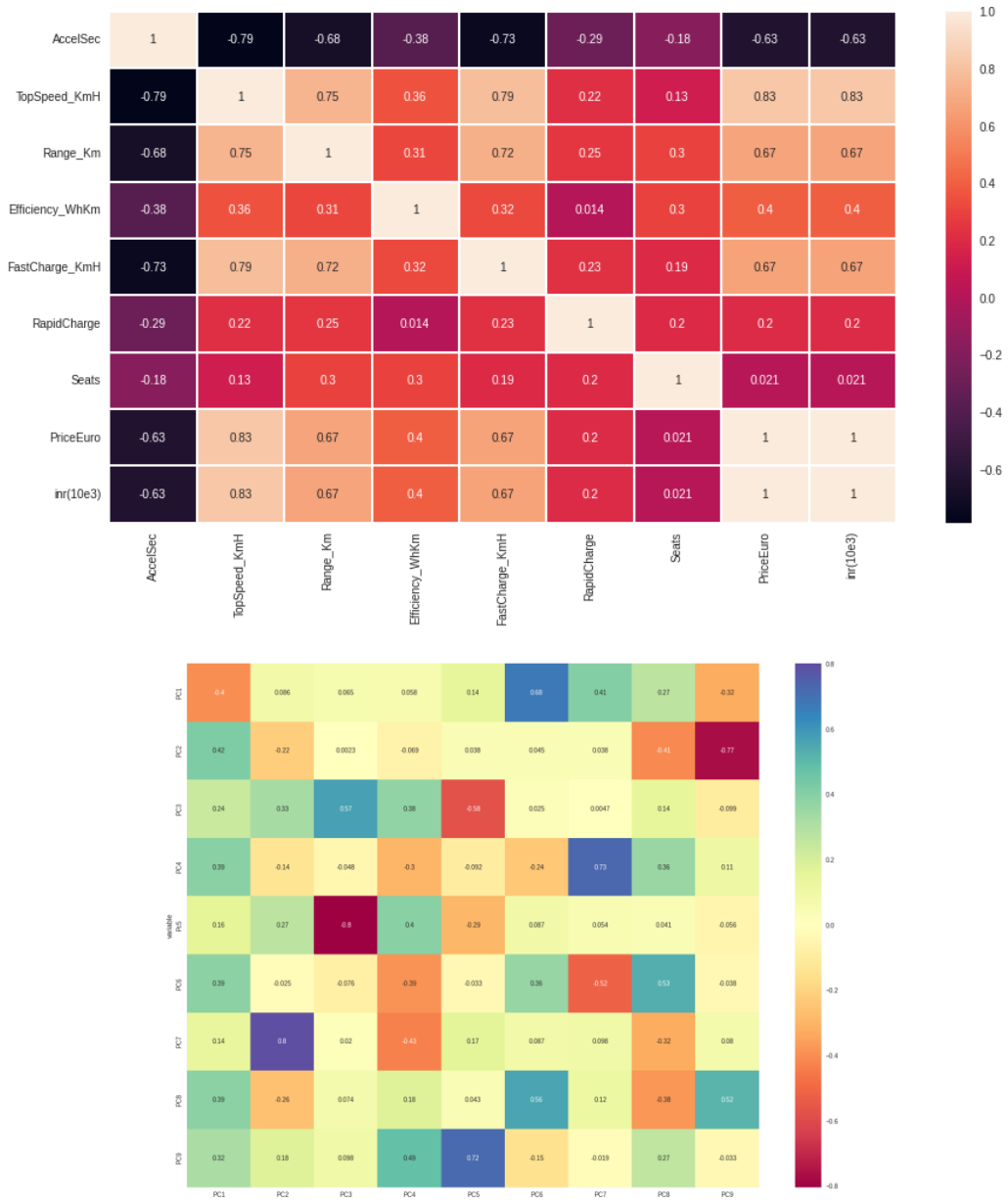


***For Electric Vehicle Market one of the most important keys is Charging*:**



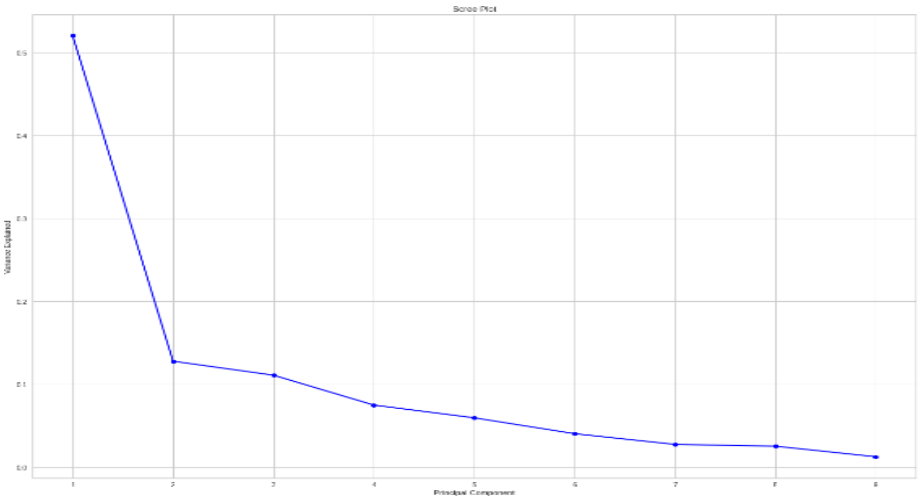
Correlation Matrix

A correlation matrix is essentially a tabular representation showcasing the correlation between variables. It's most effective when applied to variables demonstrating a linear relationship. The coefficients in the matrix illustrate the correlations among different variables. This correlation matrix is typically visualized using a heatmap, as depicted in the figure below. In general, a correlation coefficient value exceeding 0.7 is indicative of a strong relationship between two variables.



The Scree Plot serves as a visual aid in determining the optimal number of Principal Components (PCs) to retain. This plot is a simple line graph displaying the eigenvalues of each individual PC, with eigenvalues depicted on the y-axis and the number of factors on the x-axis. Typically, the plot exhibits a downward curve, starting high on the left, declining rapidly, and then plateauing at some point.

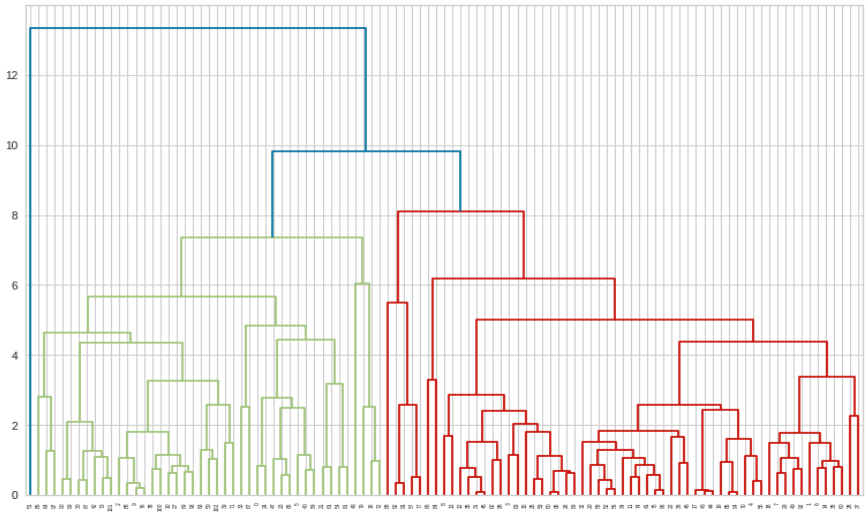
The Scree Plot criterion involves identifying the "elbow" in the curve, where the decline levels off. This point indicates the optimal number of PCs to retain. Additionally, the Proportion of Variance Plot is used to ensure that the selected PCs collectively account for at least 80% of the total variance in the data.



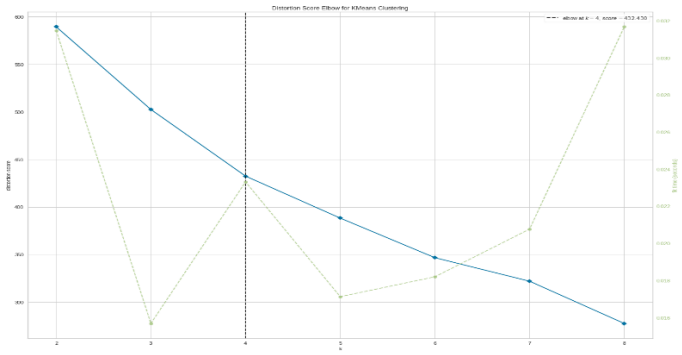
Extracting Segments

- ***Dendrogram***

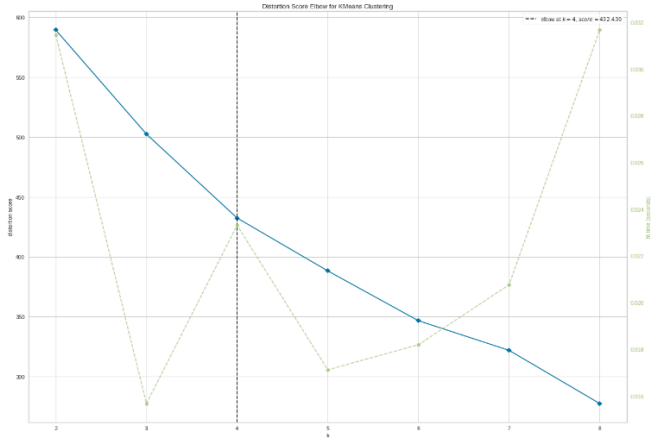
This technique is specifically tailored for the agglomerative hierarchical method of clustering. In this method, each data point starts as its own cluster, and clusters are progressively merged based on their distances in a hierarchical manner. To determine the optimal number of clusters for hierarchical clustering, we utilize a dendrogram—a tree-like chart illustrating the sequence of cluster merges or splits. In line with other cluster validation metrics, the agglomerative hierarchical method often suggests considering four to five clusters for effective clustering.



The Elbow Method is a widely used technique for determining the optimal number of clusters in a dataset. It operates by calculating the Within-Cluster-Sum of Squared Errors (WSS) for different numbers of clusters (k) and identifying the point at which the change in WSS diminishes significantly. Additionally, the function provides insights into the time required to generate models for various cluster numbers, represented by the green line.



Evaluating the clusters suing Distortion

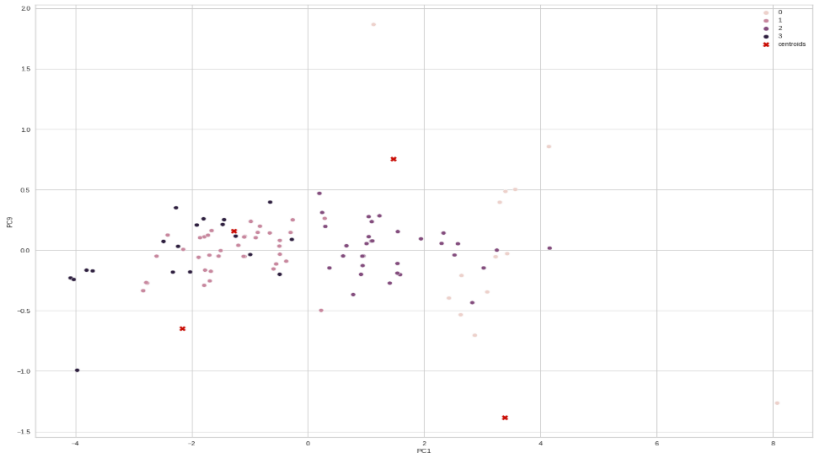


Evaluating the clusters suing silhouette

3. **Methodology**:

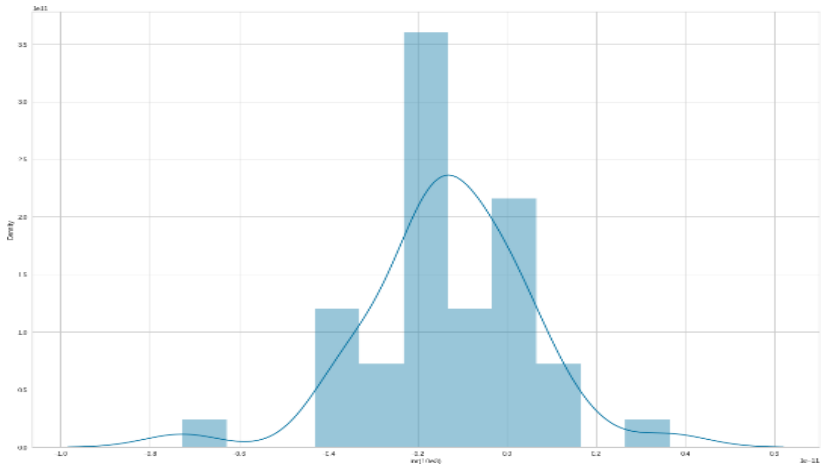
K-Means Algorithm

K Means algorithm is an iterative algorithm that tries to partition the dataset into pre-defined distinct non-overlapping subgroups where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid is at the minimum. The less variation we have within clusters, the more homogeneous the data points are within the same cluster.



Prediction of Prices most used cars

After completion of training the model process, we test the remaining 60% of data on the model. The obtained results are checked using a scatter plot between predicted values and the original test data set for the dependent variable and acquired similar to a straight line as shown in the figure and the density function is also normally distributed.



4. Model Development:

Training/testing data split.

LinearRegression(). fit(Xtrain,ytrain) command is used to fit the data set into model. The values of intercept, coefficient, and cumulative distribution function (CDF) are described in the figure

```
X=data2[['PC1', 'PC2','PC3','PC4','Pc5','PC6', 'PC7','PC8','PC9']] y=df['inr(10e3)']

X_train, X_test, y_train, y_test = train_test_split(X, y,test_size=0.4, random_state=101)
lm=LinearRegression().fit(X_train,y_train)

print(lm.intercept_)

4643.522050485438

lm.coef_

 array([ 1101.58721,  -741.20904,   208.53617,   508.32246,   122.3533 ,
        1579.00686,   333.61147, -1079.99512,  1461.72269])

X_train.columns

 Index(['PC1', 'PC2', 'PC3', 'PC4', 'Pc5', 'PC6', 'PC7', 'PC8', 'PC9'], dtype='object')

cdf=pd.DataFrame(lm.coef_, X.columns, columns=['Coeff']) cdf
```

	Coeff
PC1	1101.5872
PC2	-741.2090
PC3	208.5362
PC4	508.3225
Pc5	122.3533
PC6	1579.0069
PC7	333.6115
PC8	-1079.9951
PC9	1461.7227

5. Results:

Performance metrics of the models

```
In [79]:
print('MAE:',metrics.mean_absolute_error(y_test,predictions))
print('MSE:',metrics.mean_squared_error(y_test,predictions))
print('RMSE:',np.sqrt(metrics.mean_squared_error(y_test,predictions)))
MAE: 1.6674069532503684e-12
MSE: 4.854762404626698e-24
RMSE: 2.2033525375270063e-12

In [80]:
metrics.mean_absolute_error(y_test,predictions)
Out[80]:
1.6674069532503684e-12

In [81]:
metrics.mean_squared_error(y_test,predictions)
Out[81]:
4.854762404626698e-24

In [82]:
np.sqrt(metrics.mean_squared_error(y_test,predictions))
Out[82]:
2.2033525375270063e-12
```

6. Conclusion:

So, from the analysis we can see that the optimum targeted segment should be belonging to the following categories:

Behavioural: Mostly from our analysis there are cars with 5 seats.

Demographic:

- **Top Speed & Range:** With a large area of market the cost is dependent on Top speeds and Maximum range of cars.
- **Efficiency:** Mostly the segments are with most efficiency

Psychographic:

Price: From the above analysis, the price range is between 16,00,000 to 1,80,00,000.

Finally, our target segment should contain cars with most Efficiency, contains Top Speed and price between 16 to 180 lakhs with mostly with 5 seats

Market Segmentation and Targeting Strategy for an Electric Vehicle Startup in India

Date: 17/02/2024

1. Abstract: This project addresses the strategic challenge faced by an electric vehicle (EV) startup in India: identifying the optimal target market for its offerings. Through comprehensive market segmentation analysis, the project aims to provide actionable insights to guide the startup's entry strategy into the diverse Indian market. Leveraging machine learning techniques, customer reviews and associated data were collected and analyzed to identify distinct customer segments with unique preferences and needs. Key aspects such as ratings, visual appeal, reliability, performance, service experience, extra features, comfort, maintenance cost, and value for money were examined to understand customer perceptions. The analysis revealed two distinct clusters of customers with varying levels of satisfaction and preferences. By uncovering insights into customer preferences, satisfaction levels, and areas for improvement, the project enables the EV startup to refine its product offerings, tailor marketing strategies, and strategically position itself for long-term growth and success in the dynamic Indian EV market.

2. Problem Statement:

The electric vehicle (EV) startup in India seeks to optimize its market entry strategy by effectively targeting customer segments most likely to embrace its offerings. With the Indian market offering diverse possibilities, selecting the right segment is imperative for the startup's success. This project addresses this challenge through comprehensive market segmentation analysis. By leveraging machine learning techniques to cluster customers based on their reviews and preferences, the goal is to identify distinct customer segments with unique needs and preferences. Additionally, the analysis aims to uncover insights into customer perceptions, satisfaction levels, and areas for improvement. Ultimately, the project aims to provide actionable recommendations to the EV startup, enabling it to refine its product offerings, tailor its marketing strategies, and strategically position itself within the Indian EV market for long-term growth and success.

3. Scope and limitations :

Scope: The scope of this project includes conducting market segmentation analysis to assist an electric vehicle (EV) startup in India in identifying the most promising target market segment for its offerings. The analysis involves collecting and analyzing customer reviews and associated data points, utilizing machine learning techniques to identify distinct customer segments based on preferences and needs. Key aspects such as ratings, visual appeal, reliability, performance, service experience, extra features, comfort, maintenance cost, and value for money are examined to gain insights into customer perceptions. The project aims to provide actionable recommendations to guide the startup's entry strategy, marketing efforts, and product development initiatives.

Limitations: Several limitations should be considered in the context of this project. Firstly, the analysis relies on available customer review data, which may not fully capture the diverse range of customer preferences and experiences. Additionally, the scope of the analysis is limited to the information contained within the collected data set, and external factors such as market dynamics and competitor strategies may not be fully accounted for. Furthermore, the accuracy and representativeness of the identified customer segments depend on the quality and quantity of the

data available. Finally, while the project aims to provide valuable insights, the ultimate success of the startup's market entry strategy may be influenced by various external factors beyond the scope of this analysis.

4. Data Collection and preprocessing:

In the context of the electric vehicle (EV) startup project, data collection involves gathering relevant information about customer reviews, ratings, and associated data points from various sources such as online platforms, surveys, and customer feedback channels. This process entails systematically gathering structured and unstructured data to gain a comprehensive understanding of customer perceptions, preferences, and behaviors related to EVs. By collecting diverse and representative data, the project aims to obtain valuable insights into customer segments, market dynamics, and areas for improvement, ultimately informing strategic decision-making and enhancing the startup's positioning in the competitive EV market landscape.

Websites used for researching:

- https://www.kaggle.com/

The datasets I worked on for the project:

1. https://www.kaggle.com/datasets/deadprstkrish/ev-cars-user-reviews-india/data
   a. 2-wheeler-EV-bikewale.csv
   b. 4-wheeler-EV-cardekho.csv
   The data has been scraped from many popular websites like carDekho, carwale and bikewale. First dataset contains model of EV bikes already available in india, offering valuable insights into the experiences of different users of different models along woth highlighting what are the main concerns associated and ratings. Models of ev cars available in india, user reviews based on various attributes associated. This can give the EV startup an insight growth trends, consumer preference and competitor analysis.

   To perform customer/market segmentation, we first need to clean and preprocess the data. Here's how we can do it:

1. Remove unnecessary punctuation and special characters.
2. Tokenize the text into individual words.
3. Convert text to lowercase.
4. Remove stopwords.
5. Lemmatize words to reduce inflectional forms.

```
def clean_text(text):
    if isinstance(text, str):  # Check if the input is a string
        text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation
        text = text.lower()  # Convert text to lowercase
        tokens = word_tokenize(text)  # Tokenize text
        tokens = [word for word in tokens if word not in stopwords.words('english')]  # Remove stopwords
        lemmatizer = WordNetLemmatizer()
        tokens = [lemmatizer.lemmatize(word) for word in tokens]  # Lemmatize words
        return ' '.join(tokens)
    else:
        return ''  # Return empty string if input is not a string
```

```
[ ]  # Clean and preprocess the 'review' column
     df['clean_review'] = df['review'].apply(clean_text)
```

5. Exploratory Data Analysis (EDA):

```
# Check the data types of columns in the dataframe
print(df.dtypes)
```

```
review                 object
Used it for            object
Owned for              object
Ridden for             object
rating                  int64
Visual Appeal         float64
Reliability           float64
Performance           float64
Service Experience    float64
Extra Features        float64
Comfort               float64
Maintenance cost      float64
Value for Money       float64
Model Name             object
dtype: object
```

6. Methodology:

Now, we have cleaned and pre processed the text data. We can proceed with customer/market segmentation using machine learning techniques such as clustering. We can use methods like TF-IDF vectorization followed by clustering algorithms like KMeans or hierarchical clustering to segment the customers based on their reviews.

Here's how TF-IDF vectorization works:

1. **Term Frequency (TF)**: This component measures the frequency of a term (word) in a document. It is calculated by dividing the number of times a term appears in a document by the total number of terms in the document. The idea is to identify the importance of a term within a specific document.
2. **Inverse Document Frequency (IDF)**: This component measures the significance of a term across a collection of documents. It is calculated by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of the result. The IDF value decreases as the term appears in more documents, indicating that the term is less important or informative.
3. **TF-IDF Vectorization**: To calculate the TF-IDF score for a term in a document, we multiply its TF score by its IDF score. This process assigns higher weights to terms that are frequent in a specific document but rare across all documents, thus highlighting terms that are important and unique to that document.

Once TF-IDF vectorization is performed on the textual data, the resulting TF-IDF matrix serves as input for clustering algorithms like KMeans. KMeans is an unsupervised learning algorithm used for clustering data points into a specified number of clusters. Here's how KMeans clustering works:

7. Model development :

I am proceeding with customer segmentation using TF-IDF vectorization followed by KMeans clustering. TF-IDF (Term Frequency-Inverse Document Frequency) is a numerical statistic that reflects the importance of a word in a document relative to a

collection of documents. KMeans clustering is an unsupervised learning algorithm that partitions the data into k clusters based on similarities.

- Initialization: KMeans starts by randomly selecting the centroids of the clusters.

- Assignment: Each data point is assigned to the nearest centroid based on a distance metric (commonly Euclidean distance).

- Update: After all data points are assigned, the centroids are updated by taking the mean of all data points assigned to each cluster.

- Iteration: Steps 2 and 3 are repeated iteratively until convergence, i.e., until the centroids no longer change significantly or a predefined number of iterations is reached.

- Result: Once convergence is achieved, KMeans produces clusters where data points within the same cluster are more similar to each other compared to data points in different clusters.

```
#proceeding with customer segmentation using TF-IDF vectorization followed by KMeans clustering.
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
```

```
# Vectorize the text data using TF-IDF
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(df['clean_review'])
```

```
# Determine the optimal number of clusters using silhouette score
max_clusters = 10
best_score = -1
best_k = 2
for k in range(2, max_clusters + 1):
    kmeans = KMeans(n_clusters=k, random_state=42)
    kmeans.fit(X)
    labels = kmeans.labels_
    score = silhouette_score(X, labels)
    if score > best_score:
```

```
# Perform KMeans clustering with the optimal number of clusters
kmeans = KMeans(n_clusters=best_k, random_state=42)
kmeans.fit(X)
df['cluster'] = kmeans.labels_
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1
  warnings.warn(
```

```
# Display the clusters
print(df[['clean_review', 'cluster']])
```

```
                                          clean_review  cluster
0    checked bike capacity 150 km 1 full charge giv...        1
1    performance poor bike charging problem big thi...        0
2    purchased april 2022 sale staff clueless new v...        1
3    issue come scooty part available service centr...        0
4    dont buy vehicle unless near tv iqube service ...        0
..                                                 ...      ...
839  scooty ok 250 motor power le scooter power one...        1
840  superb scooty good look many color option le s...        1
841  2 year condition good 2 year scooter stopped m...        0
```

```
[ ]  3    issue come scooty part available service centr...        0
     4    dont buy vehicle unless near tv iqube service ...        0
     ..                                                  ...      ...
     839  scooty ok 250 motor power le scooter power one...        1
     840  superb scooty good look many color option le s...        1
     841  2 year condition good 2 year scooter stopped m...        0
     842  compare scooter best bike comfortable seat dri...        1
     843  bike good segment use person aged 12 plus spee...        1

     [844 rows x 2 columns]
```

8. Results:

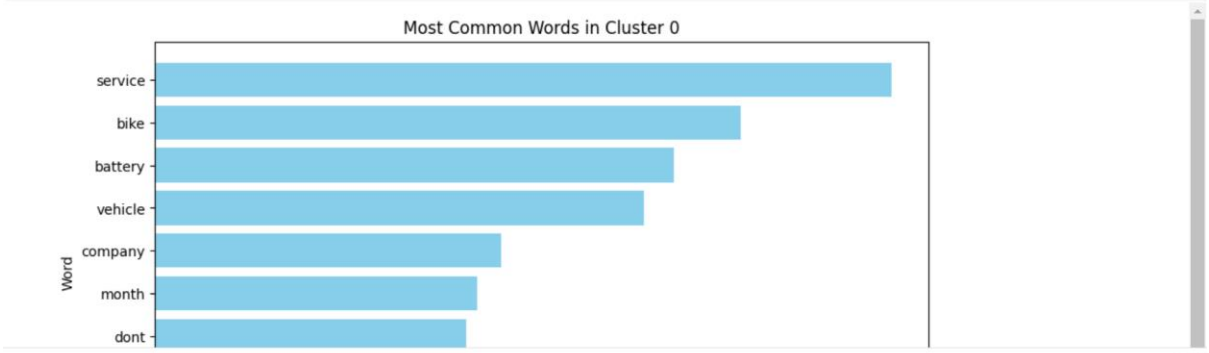
```
[ ] #performing some analysis on the clustered data.
```

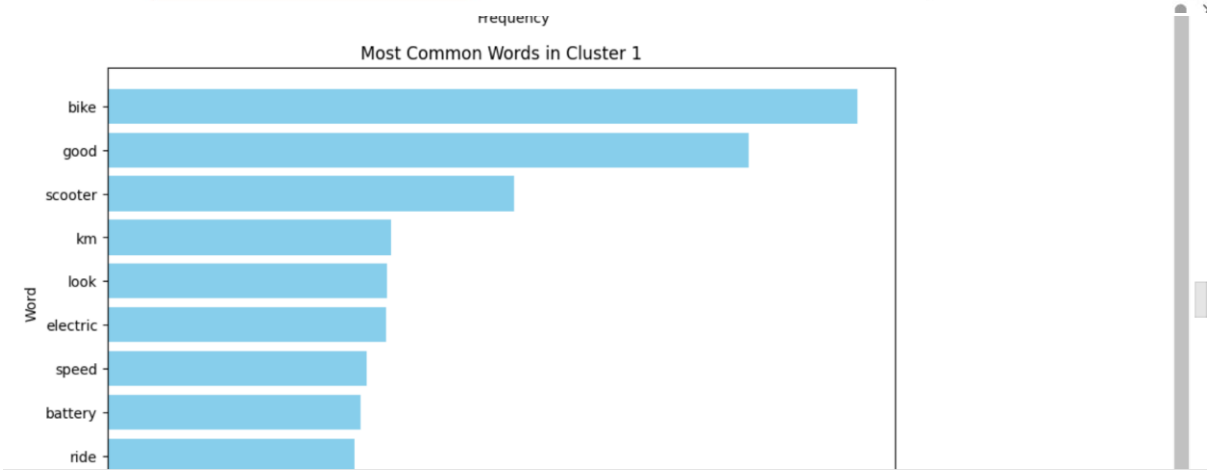
```
[ ] from collections import Counter
    import matplotlib.pyplot as plt
```

```
[ ] # Function to get the most common words in a cluster
    def get_most_common_words(cluster_id, n_words=10):
        cluster_reviews = df[df['cluster'] == cluster_id]['clean_review']
        all_words = ' '.join(cluster_reviews).split()
        word_counts = Counter(all_words)
        return word_counts.most_common(n_words)
```

```
[ ] # Function to plot the most common words in each cluster
    def plot_most_common_words(cluster_id):
        word_freq = get_most_common_words(cluster_id)
        words, frequencies = zip(*word_freq)
        plt.figure(figsize=(10, 6))
        plt.barh(words, frequencies, color='skyblue')
        plt.xlabel('Frequency')
        plt.ylabel('Word')
        plt.title(f'Most Common Words in Cluster {cluster_id}')
```

```
[ ] # Plot the most common words in each cluster
    for cluster_id in range(best_k):
        plot_most_common_words(cluster_id)
```





```
[ ]  # Summary statistics for each cluster
     cluster_stats = df.groupby('cluster').agg({
         'rating': ['mean', 'median'],
         'Visual Appeal': ['mean', 'median'],
         'Reliability': ['mean', 'median'],
         'Performance': ['mean', 'median'],
         'Service Experience': ['mean', 'median'],
         'Extra Features': ['mean', 'median'],
         'Comfort': ['mean', 'median'],
         'Maintenance cost': ['mean', 'median'],
         'Value for Money': ['mean', 'median']
     })
     print(cluster_stats)
```

```
            rating        Visual Appeal        Reliability        Performance  \
              mean median          mean median        mean median        mean
cluster
0         1.551237    1.0      2.662551    3.0    1.726141    1.0    2.110169
1         4.278075    5.0      4.326613    5.0    4.120000    4.0    4.264317

                Service Experience        Extra Features          Comfort  \
           median               mean median           mean median     mean
```

Based on the summary statistics provided, it appears that the data has been segmented into two clusters. Let's analyze the key insights from the summary statistics:

1. **Rating**:
    - Cluster 0 has significantly lower mean and median ratings compared to Cluster 1. This suggests that customers in Cluster 0 generally rated their experience lower across various aspects.
2. **Visual Appeal, Reliability, Performance, Service Experience, Extra Features, Comfort, Maintenance Cost, Value for Money**:
    - For all these aspects, Cluster 1 consistently has higher mean and median ratings compared to Cluster 0.
    - Cluster 1 generally perceives the product more positively across these aspects compared to Cluster 0.
    - Notably, the difference in ratings is most pronounced in Visual Appeal, Reliability, and Value for Money, where Cluster 1 has substantially higher ratings.

These insights suggest that there are distinct differences in customer perceptions and preferences between the two clusters. Cluster 1 appears to be more satisfied with the product across various dimensions, while Cluster 0 is less satisfied. This segmentation could be indicative of different customer segments with varying needs, expectations, and experiences with the product.

Further analysis and exploration could involve understanding the characteristics of customers in each cluster, identifying potential factors driving the differences in perceptions, and devising targeted strategies to address the needs of each segment effectively.

9. Conclusion:

The analysis of customer segmentation based on summary statistics reveals distinct differences in perceptions and preferences between two clusters. Cluster 1 consistently exhibits higher mean and median ratings across various aspects such as visual appeal, reliability, performance, service experience, extra features, comfort, maintenance cost, and value for money compared to Cluster 0. Notably, the most significant disparities are observed in visual appeal, reliability, and value for money, where Cluster 1 shows substantially higher ratings. These findings indicate the

presence of different customer segments with varying levels of satisfaction and preferences. Further exploration into the characteristics of customers in each cluster and the factors driving these differences is warranted to inform targeted strategies for addressing the needs of each segment effectively.

Understanding customer segmentation and the distinct preferences of each cluster can provide valuable insights for the EV startup. By tailoring its marketing efforts, product development initiatives, and customer service strategies to cater to the specific needs and preferences of each segment, the startup can enhance customer satisfaction, loyalty, and brand perception. For example, targeting Cluster 1, which exhibits higher satisfaction levels, with promotional campaigns highlighting the product's visual appeal, reliability, and value for money could help attract and retain customers in this segment. Conversely, for Cluster 0, addressing any identified shortcomings in these areas and implementing targeted improvements can help mitigate dissatisfaction and improve overall customer experience. Ultimately, leveraging insights from customer segmentation can enable the EV startup to optimize its resources, maximize market penetration, and establish a competitive edge in the dynamic EV market landscape.