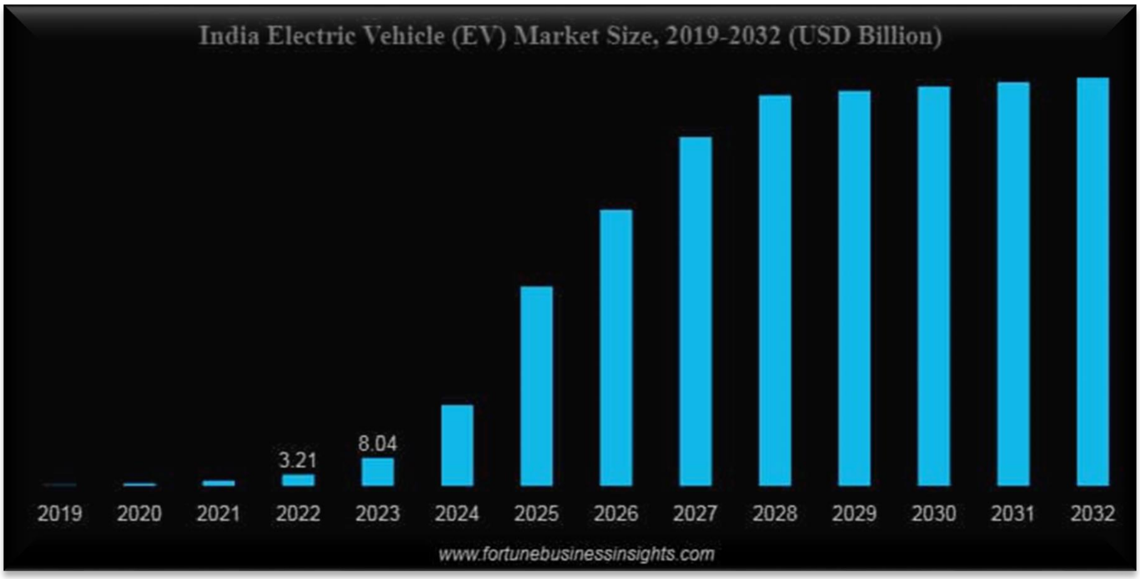
ELECTRIC VEHICLE MARKET SEGMENTATION

PAYAL CHAUHAN

Abstract

India’s electric vehicle (EV) market is experiencing rapid growth, driven by government incentives, technological advancements, and increasing consumer awareness. Despite challenges such as high upfront costs and limited charging infrastructure, the EV market in India is poised for significant expansion. The Indian Electric vehicle market size was valued at USD 32.1 Billion in 2023 and is expected to reach USD 150.2 Billion by 2032, at a CAGR of 25.1% during the forecast period 2023 - 2032. The attractive incentives being offered by the Indian government on the production and purchase of electric vehicles to

encourage the adoption of electric vehicles are anticipated to derive the growth of the marker over the forecast period.



Market segmentation has been become a crucial tool for evolving transportation technology such as electric vehicles in emerging markets to explore and implement for extensive adoption. This study aims to segment the Indian electric vehicle market based on key consumer characteristics and preferences. By understanding the diverse needs and behaviours of EV buyers in INDIA, manufactures and policymakers can develop targeted strategies to accelerate EV adoption. Ev adoption is expected to grow phenomenally in near future as low emission and low operating cost and hence it derives a considerable amount of upcoming academic research curiosity. The segmentation analysis is based on various factors, including demographics, psychographics, purchase behaviour, and EV usage patterns. The findings of this study offer valuable insights into the Indian EV market, identifying potential market niches and opportunities for growth.

In this report with the help of Fermi Estimation we are going to analyse the data and solve the problem.

Market Segmentation

Market Segmentation is the process of dividing market into distinct groups of customers who have similar needs, wants or behaviours. This allows business to tailor their marketing strategies and product offerings to specific segments, increasing their chances of success.

In the context of electric vehicle (EV) industry, market segmentation is crucial for several reasons:

1. **Understanding customer preferences:**
   By identifying different groups of EV buyers with unique needs and preferences, manufactures can develop products that caters to each segment.

2. **Identifying growth opportunities:**
   Market segmentation can reveal untapped market niches or emerging growths that can drive growth.

3. **Developing targeted marketing campaigns:**
   Tailored marketing messages can be more effective in reaching and persuading potential EV buyers.

4. **Optimizing resource allocation:**
   Understanding market segments can help companies allocate resources more effectively by focusing on the most promising segments.

Fermi Estimation:

A market segmentation analysis for the electric vehicle market in India, keeping in mind the available data and the feasibility of targeting different customer segments. Based on the categories of segments mentioned, we'll analyze the Indian EV market and propose a feasible strategy for your startup to enter the market.

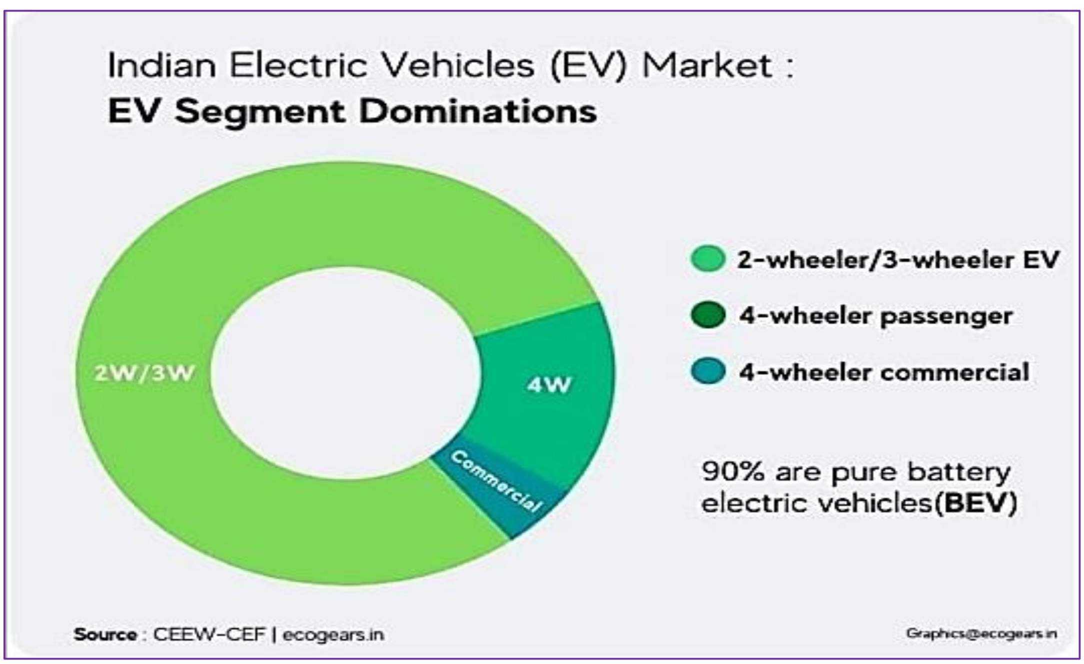
1. **Customer/Usage Segmentation:**

Understanding how potential customers will use electric vehicles can be crucial. Consider segments based on customer needs and vehicle usage patterns.

2. **Demographic Segmentation:**
   While demographic data might be more challenging to obtain, you can still consider some broad categories like age group, income levels, etc.
3. **Geographic Segmentation:**
   India is a diverse country with varying geographic characteristics. You can consider segments based on regions, cities, and urban area vs. rural areas.
4. **Vehicle type segmentation:**
   Segmenting based on the type of EVs can help you focus on the most relevant offerings.

Future:

The future of electric vehicles (EVs) in India looks bright. With continued government support, technological advancements, and increasing consumer acceptance, the Indian EV market is poised for significant growth. Even India has the potential to become a global leader in electric mobility, contributing to a cleaner and more sustainable future.



Detailed Analysis:

Collecting relevant data for electric vehicle (EV) segmentation is crucial for making informed business decisions. Data collection might involve a combination of primary data (collected directly) and secondary data (existing sources). Collaborating with research organizations, academic institutions, and industry associations can also provide access to valuable data. Ensure that any data you collect or use adheres to privacy regulations and ethical considerations.

It's important to conduct thorough research to gather accurate and up-to-date data that reflects the specific market conditions and trends in the Indian EV market. This data will serve as the foundation for your segmentation analysis and subsequent business strategy.

Implementation

Packages/Tools used:

1. Numpy: To calculate various calculations related to arrays.
2. Pandas: To read or load the datasets.
3. SKLearn: We have used LabelEncoder() to encode our values.

Data-Preprocessing Data Cleaning:

The data collected is compact and is partly used for visualization purposes and partly for clustering. Python libraries such as NumPy, Pandas, Scikit-Learn, and SciPy are used for the workflow, and the results obtained are ensured to be reproducible.

A View of Dataset:

```python
ev = pd.read_csv(r"/content/ev.csv")
ev.drop('Unnamed: 0', inplace = True, axis = 1)
```

```python
ev
```

	Brand	Model	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	PowerTrain	PlugType	BodyStyle	Segment	Seats	PriceEuro	INR
0	Tesla	Model 3 Long Range Dual Motor	4.6	233	450	161	940	Yes	AWD	Type 2 CCS	Sedan	D	5	55480	4540988.068
1	Volkswagen	ID.3 Pure	10.0	160	270	167	250	No	RWD	Type 2 CCS	Hatchback	C	5	30000	2455473.000
2	Polestar	2	4.7	210	400	181	620	Yes	AWD	Type 2 CCS	Liftback	D	5	56440	4619563.204
3	BMW	iX3	6.8	180	360	206	560	Yes	RWD	Type 2 CCS	SUV	D	5	68040	5569012.764
4	Honda	e	9.5	145	170	168	190	Yes	RWD	Type 2 CCS	Hatchback	B	4	32997	2700774.753
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
98	Nissan	Ariya 63kWh	7.5	160	330	191	440	Yes	FWD	Type 2 CCS	Hatchback	C	5	45000	3683209.500
99	Audi	e-tron S Sportback 55 quattro	4.5	210	335	258	540	Yes	AWD	Type 2 CCS	SUV	E	5	96050	7861606.055
100	Nissan	Ariya e-4ORCE 63kWh	5.9	200	325	194	440	Yes	AWD	Type 2 CCS	Hatchback	C	5	50000	4092455.000
101	Nissan	Ariya e-4ORCE	5.1	200	375	232	450	Yes	AWD	Type 2	Hatchback	C	5	65000	5320191.500

DataFrame consists of 103 rows and 14 columns, with data related to various car models and on the basis of model, the speed, acceleration, range, efficiency, etc. details achieved by the car is provided. It also tells us about charger type, body style, number of seats, and price of car in euro as well as in INR. All columns are fully populated with no missing values, and the data types consist of integers for numerical values and objects for categorical or textual data.

```
ev.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 103 entries, 0 to 102
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   Brand            103 non-null    object 
 1   Model            103 non-null    object 
 2   AccelSec         103 non-null    float64
 3   TopSpeed_KmH     103 non-null    int64  
 4   Range_Km         103 non-null    int64  
 5   Efficiency_WhKm  103 non-null    int64  
 6   FastCharge_KmH   103 non-null    int64  
 7   RapidCharge      103 non-null    object 
 8   PowerTrain       103 non-null    object 
 9   PlugType         103 non-null    object 
 10  BodyStyle        103 non-null    object 
 11  Segment          103 non-null    object 
 12  Seats            103 non-null    int64  
 13  PriceEuro        103 non-null    int64  
 14  INR              103 non-null    float64
dtypes: float64(2), int64(6), object(7)
memory usage: 12.2+ KB
```

Descriptive Statistics of the dataset

```
[ ] ev.describe()
```

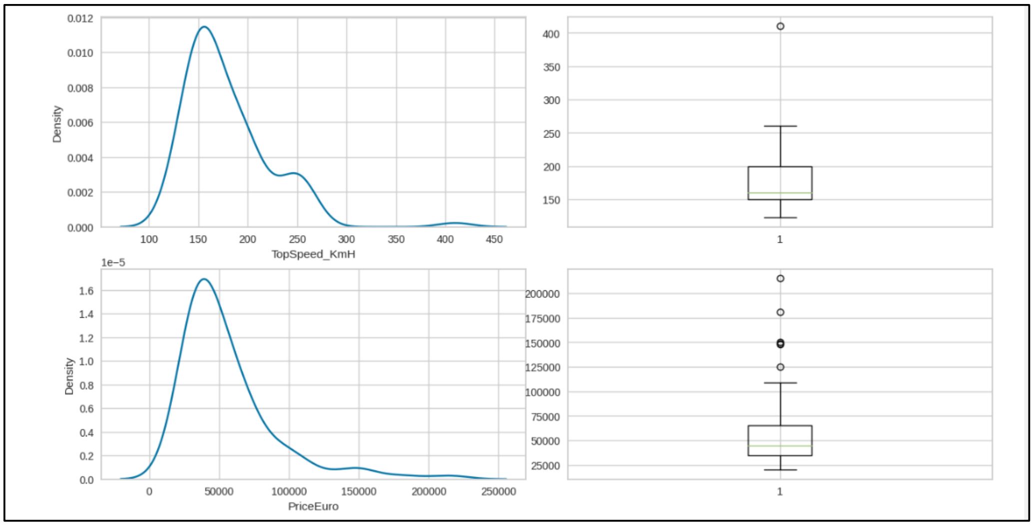
	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	Seats	PriceEuro	INR
count	103.000000	103.000000	103.000000	103.000000	103.000000	103.000000	103.000000	1.030000e+02
mean	7.396117	179.194175	338.786408	189.165049	444.271845	4.883495	55811.563107	4.983453e+06
std	3.017430	43.573030	126.014444	29.566839	203.949253	0.795834	34134.665280	5.032279e+06
min	2.100000	123.000000	95.000000	104.000000	170.000000	2.000000	20129.000000	1.647541e+06
25%	5.100000	150.000000	250.000000	168.000000	260.000000	5.000000	34429.500000	2.818024e+06
50%	7.300000	160.000000	340.000000	180.000000	440.000000	5.000000	45000.000000	3.683210e+06
75%	9.000000	200.000000	400.000000	203.000000	555.000000	5.000000	65000.000000	5.345565e+06
max	22.400000	410.000000	970.000000	273.000000	940.000000	7.000000	215000.000000	4.706232e+07

EDA

The code use Matplotlib, seaborn and plotly to visualize data. We start the Exploratory Data Analysis with some data analysis drawn from the data without Principal Component Analysis and with Principal Component Analysis. Principal Component Analysis (PCA) is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the

Principal Components. The process helps in reducing dimensions of the data to make the process of classification/regression or any form of machine learning, cost-effective.

Data Visualisation



From above graph we can infer the following from the above two graphs:

1.From Hist plot:

- Most of the cars from dataset has top speed 150 kmph.
- Most of the cars has price 50000 in Euro.

2.From boxplots:

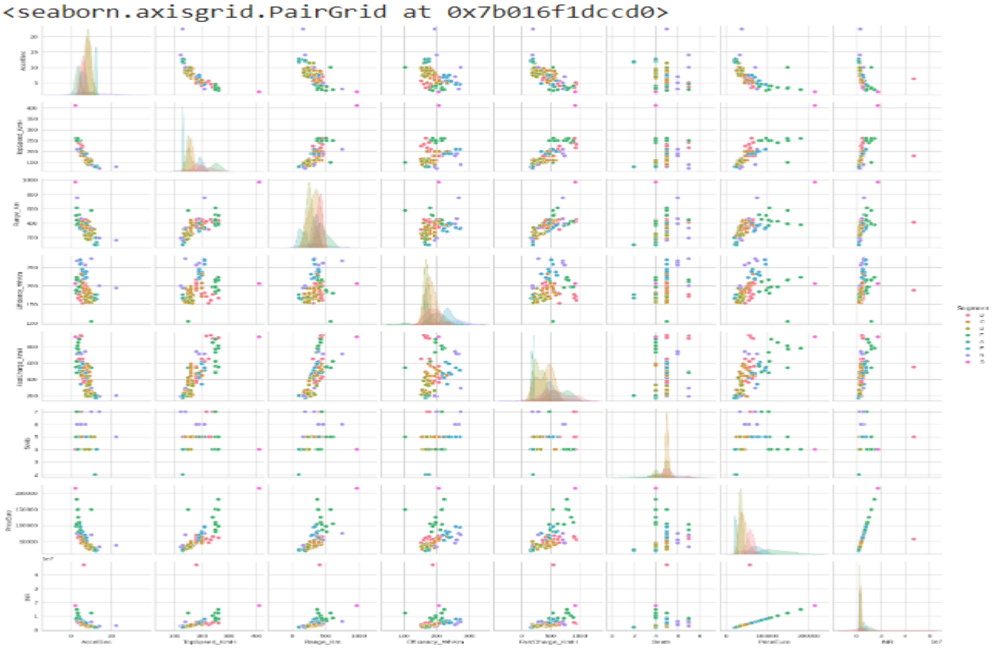
- Boxplot on 'TopSpeed_KmH' shows that cars with speed more than 250 is rare, and there is 1 such outlier.
- Boxplot on ‘PriceEuro’ shows that price rates of car above 125000 are rare, and there are 4 such outliers.

Pair Plot

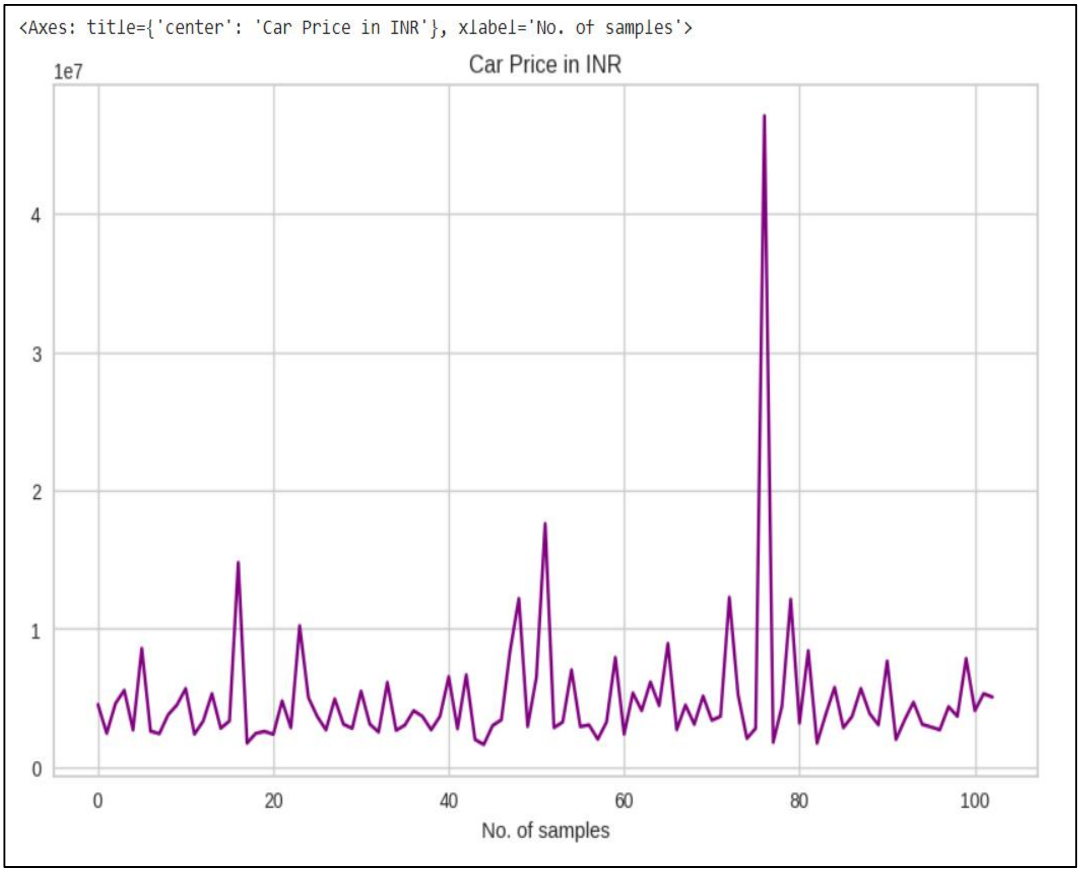
We used pairplot() function, to plot multiple pairwise bivariate distribution in a dataset.

The diagonal plots are the univariate plots, and this displays the relationship for the (n, 2) combination of variables in a DataFrame as a matrix of plots. Through this pairplot we can depict the relation between the correlation between the datasets.

```
sns.pairplot(ev, hue = 'Segment')
<seaborn.axisgrid.PairGrid at 0x7b016f1dccd0>
```



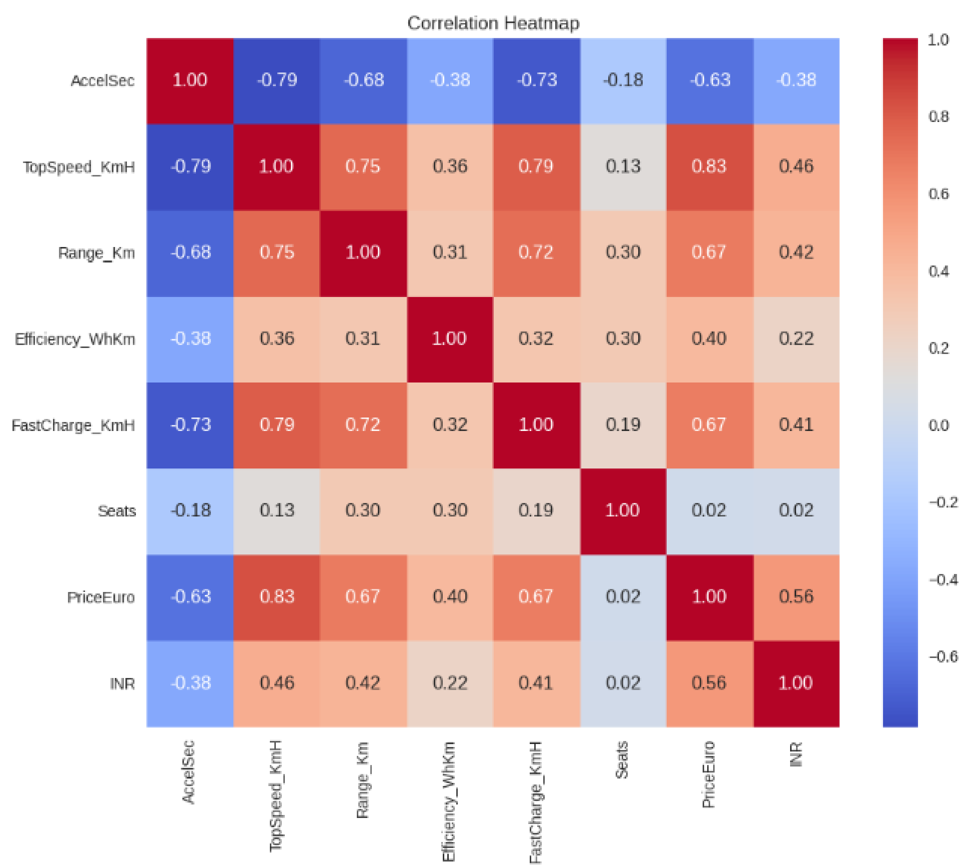
For Electric Vehicle Market one of the most important key is Charging.



HeatMap:

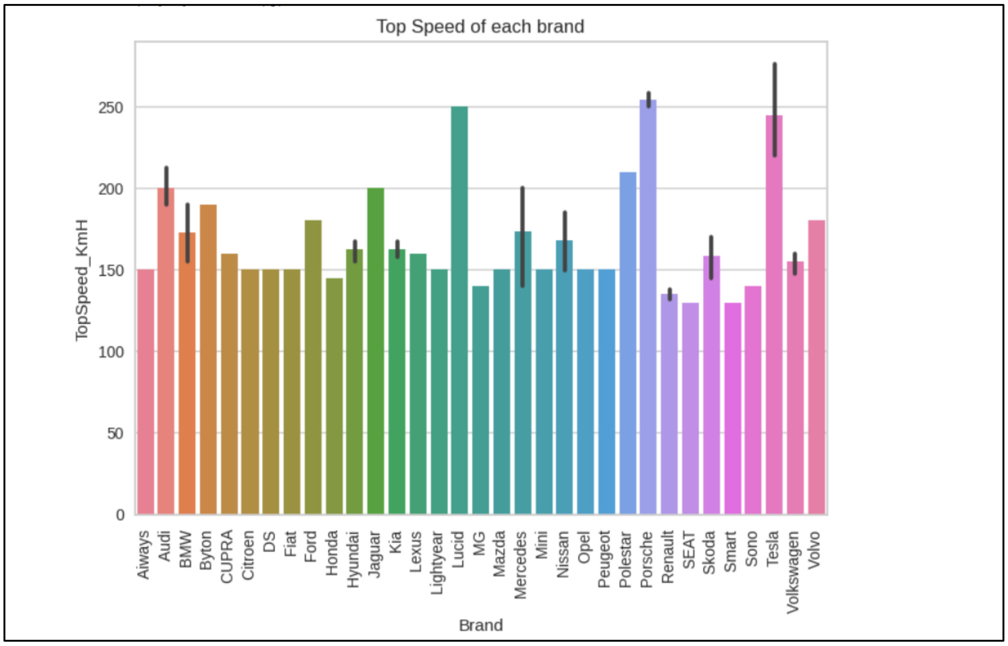
We also plotted a heatmap to find correlation between the datapoints of the dataset. It is best used in variables that demonstrate a linear relationship between each other. Coefficients for different variables. The matrix depicts the correlation between all the possible pairs of values through the heatmap in the below figure.

```
# Plot the heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='
plt.title('Correlation Heatmap')
plt.show()
```

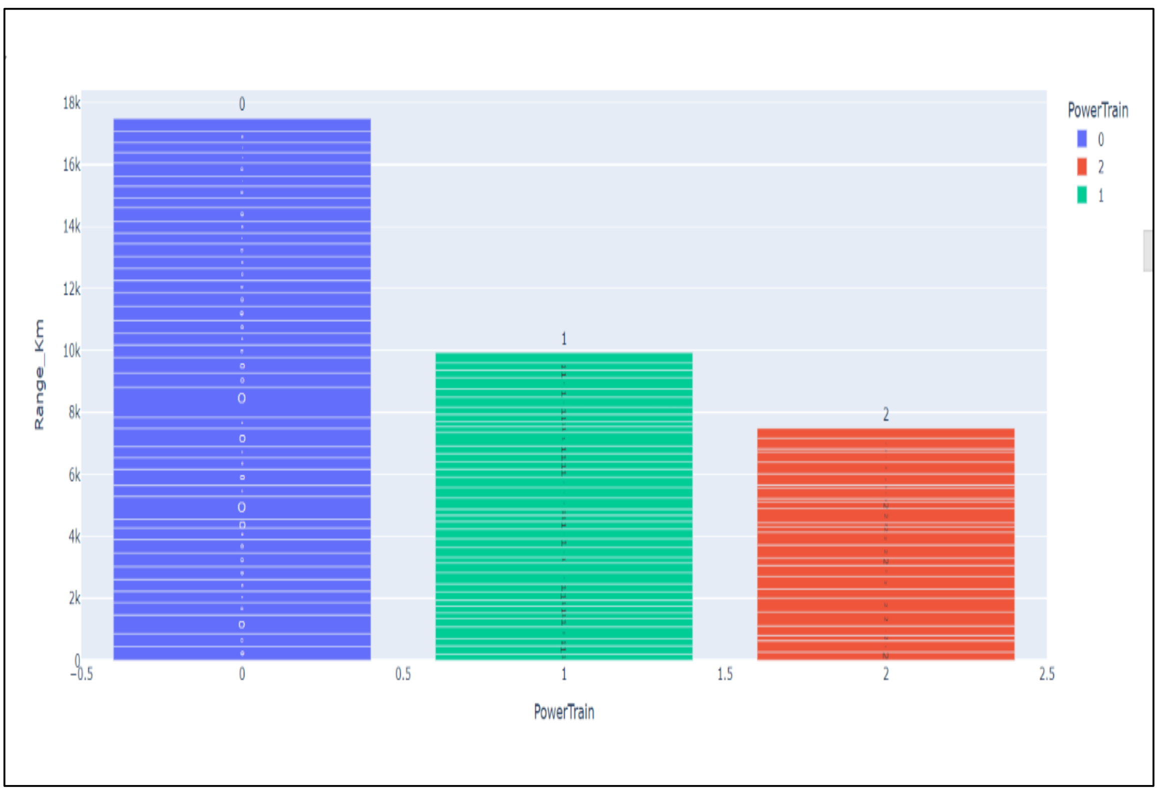


It can be clearly stated that there is a high correlation between the top speed and PriceEuro, range and FastCharge, TopSpeed and FastCharge of the car. It is also visible that the range and battery pack of the car are also related.

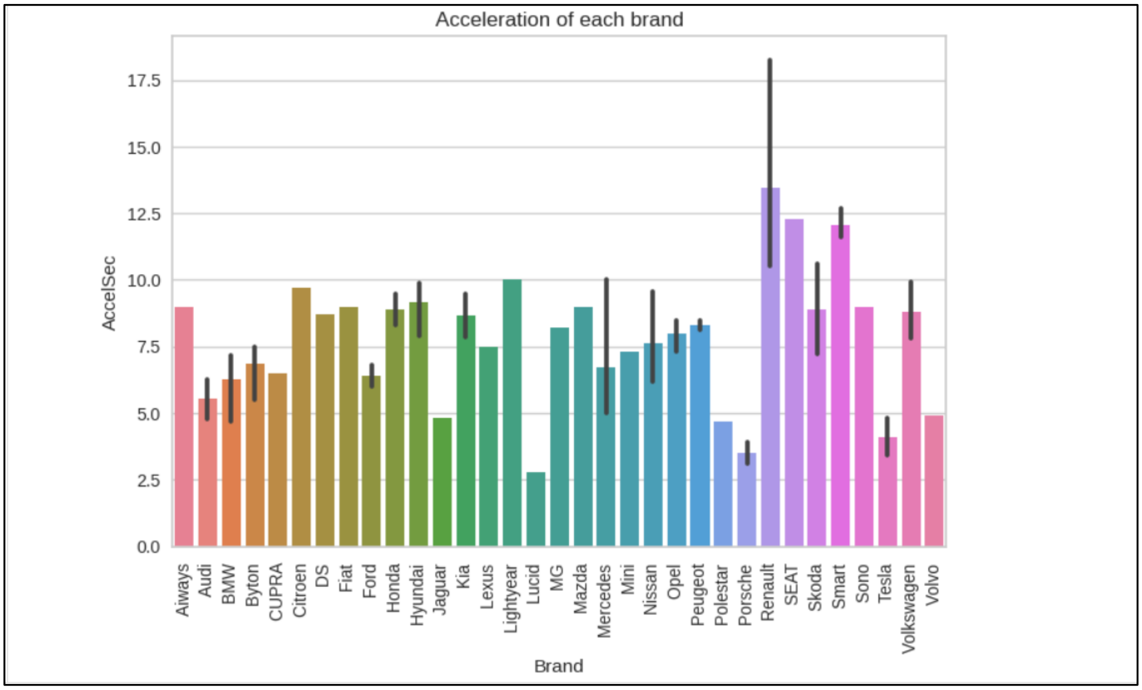
Which car has fastest Speed?



From above graph it is clear that Porche, Lucid and Tesla has produced cars with fastest speed and Smart with lowest speed.



Which car has Fastest acceleration ?

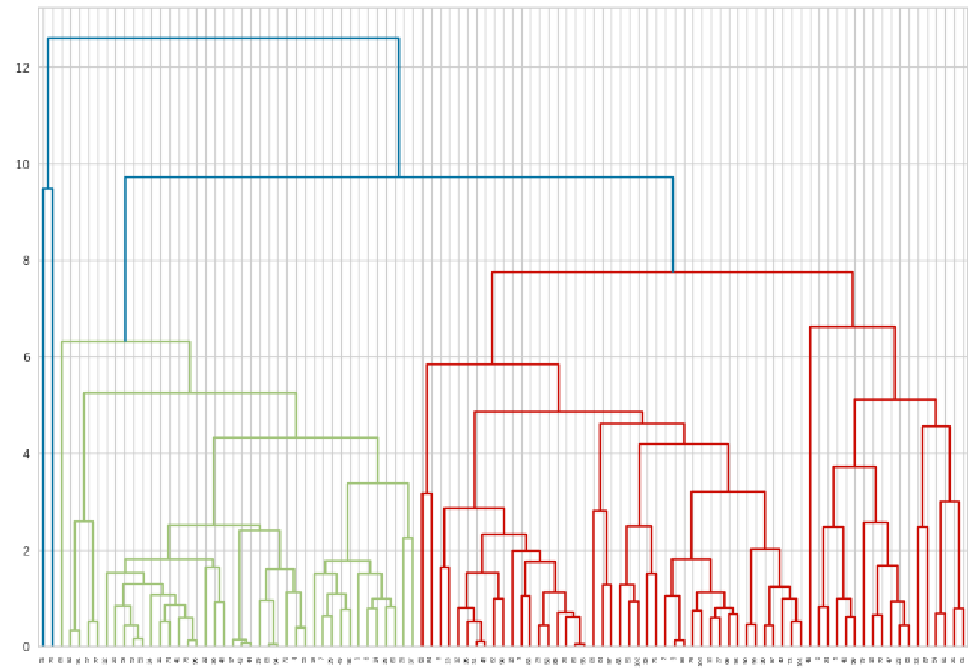


Dendrogram

Dendrogram is a tree like diagram often used in hierarchical clustering to visualize the relationships between different clusters. In the context of EV market segmentation, a dendrogram can help visualize the similarities and differences between different groups of EV consumers.

If two clusters are merged, the dendrogram will join them in a graph and the height of the join will be the distance between those clusters. As shown in Figure, we can choose the optimal number of clusters based on hierarchical structure of the dendrogram. As highlighted by other cluster validation metrics, four to five clusters can be considered for the agglomerative hierarchical as well

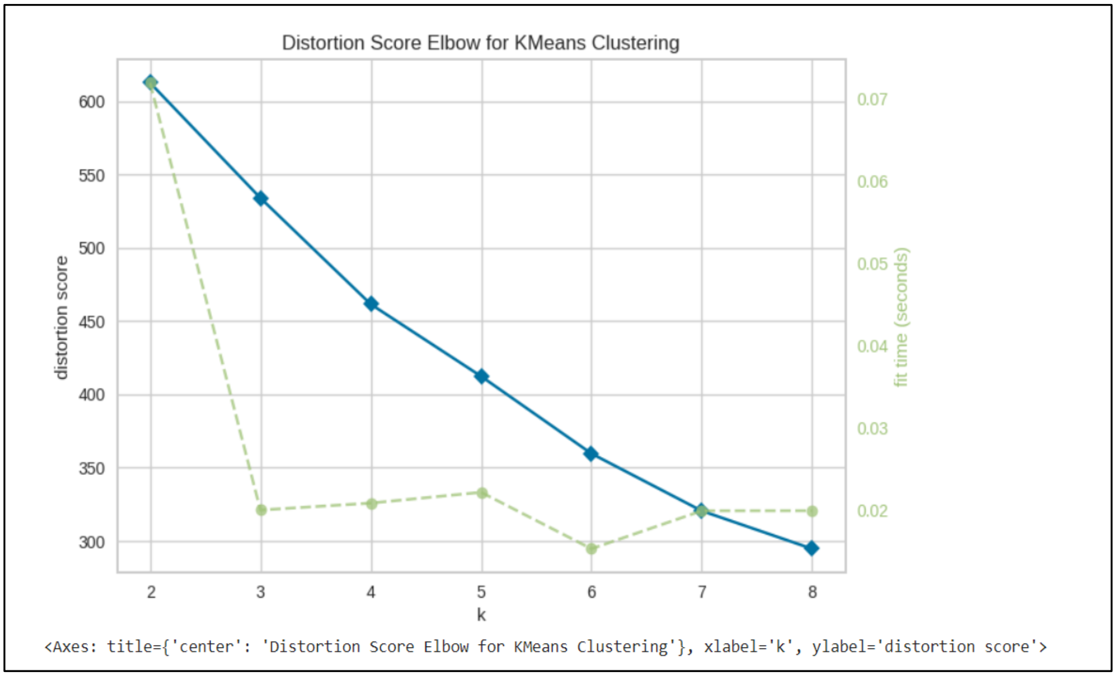
```python
linked = linkage(data, 'complete')
plt.figure(figsize=(13, 9))
dendrogram(linked, orientation='top')
plt.show()
```



Elbow Method

The elbow method is a heuristic used to determine the optimal number of clusters in cluster analysis. It involves plotting the variance explained by each cluster against the number of clusters. The plot typically shows an elbow-shaped curve, where the rate of increase in explained variance starts to decrease significantly after a certain number of clusters. This point is considered the 'elbow' and is often used to determine the optimal number of clusters.

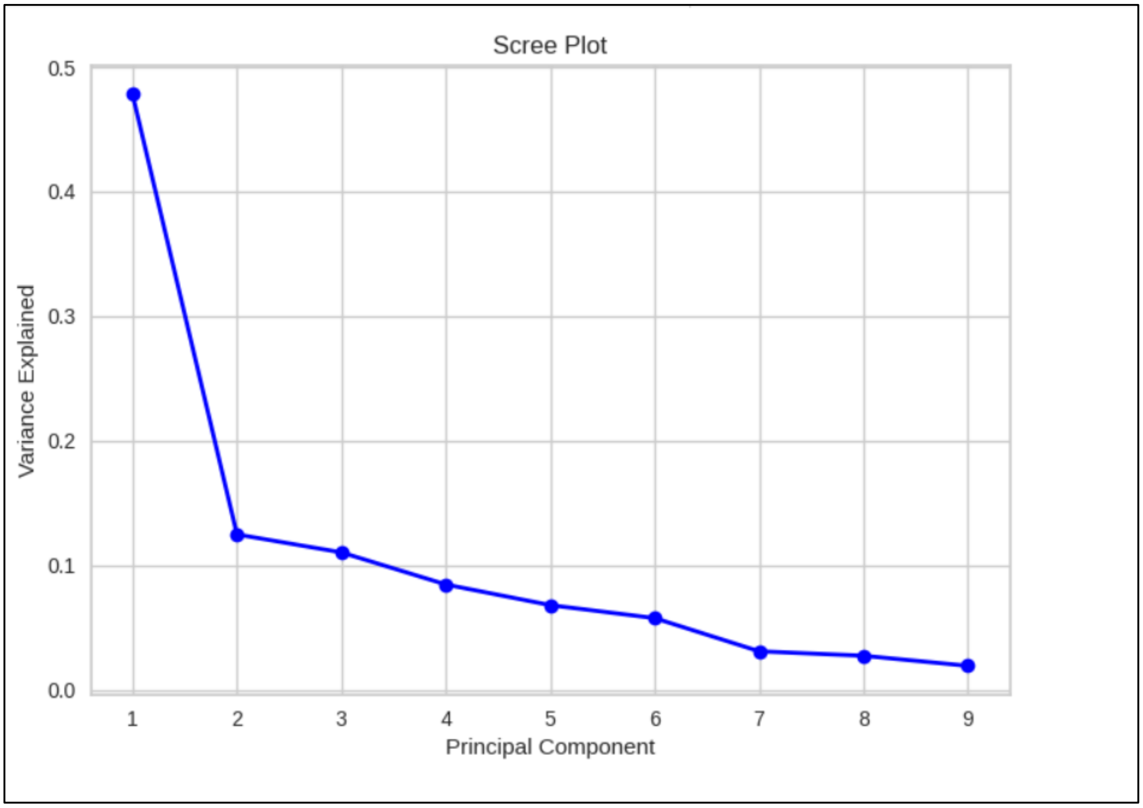
The KElbowVisualizer function fits the KMeans model for a range of clusters values between 2 to 8. As shown in Figure, the elbow point is achieved which is highlighted by the function itself. The function also informs us about how much time was needed to plot models for various numbers of clusters through the green line.



Scree Plot

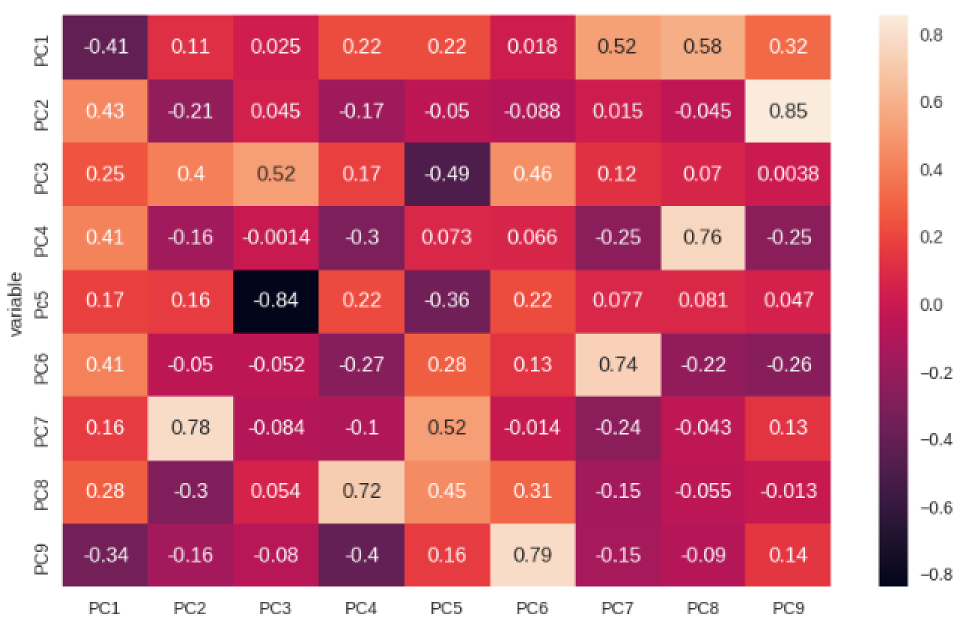
A scree plot is a graphical representation of the eigenvalues of a principal component analysis (PCA). It is used to determine the optimal number of principal components to retain in dataset. In the context of EV market segmentation, a scree plot can help to identify the most important factors or dimensions that explain the variation in consumer behaviour and preferences.

Scree plot is a valuable tool for EV market segmentation, as they help identify the optimal number of principal components and the underlying dimensions that drive consumer behaviour. The scree plot criterion looks for the “elbow” in the curve and selects all components just before the line flattens out. The proportion of variance plot: The selected PCs should be able to describe at least 80% of the variance.



Correlation matrix plot for loadings:

```
plt.rcParams['figure.figsize'] = (10, 6)
ax = sns.heatmap(loadings_df, annot = True)
plt.show()
```



Regression Analysis:

Through regression we can find the relationship between the independent variables such as top speed, range, and efficiency with that of the dependent variable price.

Linear Regression:

```
[ ] x = ev[['AccelSec','Range_Km','TopSpeed_KmH','Efficiency_WhKm', 'RapidCharge','PowerTrain']]
    y = ev['INR']
```

```
[ ] x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3, random_state = 1)
    x_train.shape, y_train.shape

((72, 7), (72,))
```

LinearRegression

```
[ ] lr = LinearRegression()
    lr.fit(x_train, y_train)

▾ LinearRegression
LinearRegression()

[ ] lr.score(x_train, y_train), lr.score(x_test, y_test)

(0.19852972039316608, 0.5324089029581873)

y_pred = lr.predict(x_test)
r2 = r2_score(y_test, y_pred)
r2

0.5324089029581873
```

The accuracy of the model is calculated using r2_score that depicts how close the estimated value is to the actual data values. The coefficient of determination, or R2, is a measure that provides information about the goodness of fit of a model. In the context of regression it is a statistical measure of how well the regression line approximates the actual data.

We can clearly say that our model is giving an accuracy of 53%.

Principal Component Analysis (PCA):

Principal Component Analysis (PCA) is the analysis of principal features of the data. The analysis is done by reducing the dimensionality of the feature space. In other words, it is a tool to reduce the features from the data to get only the required features or principal components for the learner. PCA has three major components which help to reduce dimensionality:

• The covariance matrix is the measure of how much the variables are associated with each other.

• The eigenvectors are the directors in which the data is dispersed.

• The eigenvalues are the relative importance of the directions.

```
pca = PCA(n_components=9)
t = pca.fit_transform(x)
data = pd.DataFrame(t, columns=['PC1', 'PC2','PC3','PC4','Pc5','PC6', 'PC7', 'PC8','PC9'])
data
```

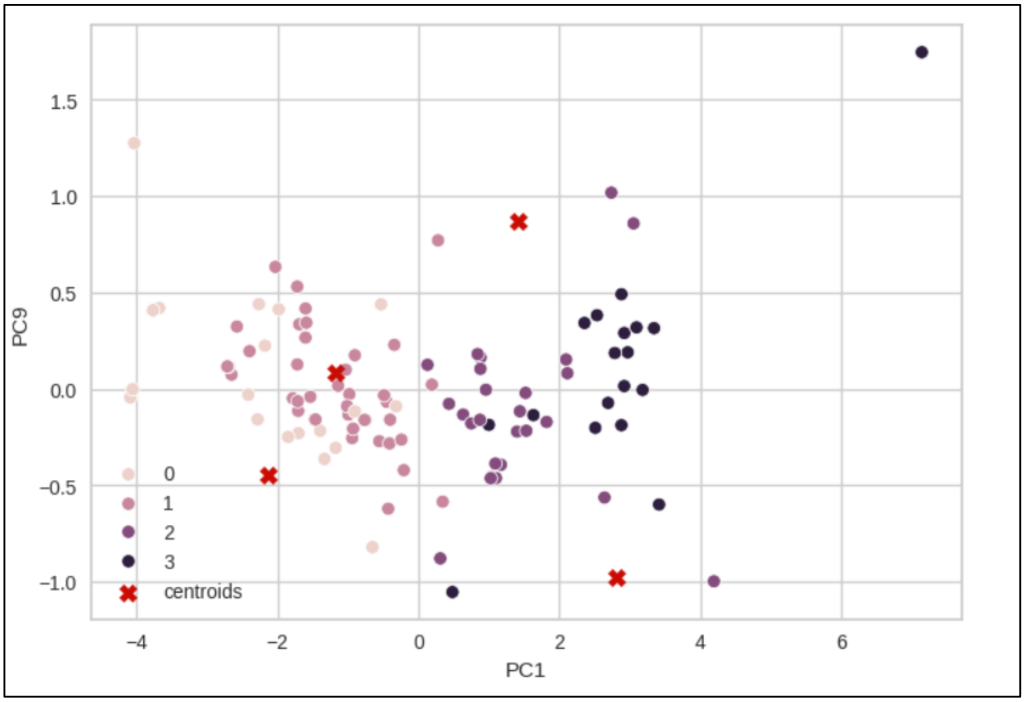
	PC1	PC2	PC3	PC4	Pc5	PC6	PC7	PC8	PC9
0	2.510948	-0.791509	-0.935325	-1.079538	0.284896	-1.042819	-0.351339	1.138032	-0.202522
1	-2.264692	-0.169049	0.928566	-0.765169	1.066840	0.195739	-0.121962	-0.382264	0.438876
2	1.632305	-0.130327	-0.580394	-0.271792	-0.237282	-0.830073	-0.167520	0.115280	-0.136624
3	0.190819	0.045366	-0.328135	-0.557701	-0.077354	1.634599	-0.271246	0.222239	0.021528
4	-2.576951	-0.645104	-0.858211	0.227355	-0.667724	0.658759	-0.252411	-0.270942	0.322600
...	...	...	...	...	...	...	...	...	...
98	-0.241445	0.385098	-0.513547	-0.033553	-0.238244	0.260184	-0.007451	0.081141	-0.263986
99	2.114916	0.808533	0.843816	0.889159	-1.427511	0.484198	-0.262005	0.040045	0.080696
100	0.840526	0.346498	-0.323806	0.284237	-0.635425	-0.763662	-0.114685	-0.146151	0.179782
101	1.520676	0.732823	0.334888	0.503652	-1.105738	-0.044156	0.150588	-0.276369	-0.021229
102	0.872541	0.562881	2.373019	0.125821	-0.148791	-0.402497	0.531462	0.101004	-0.160779

KMeans Clustering:

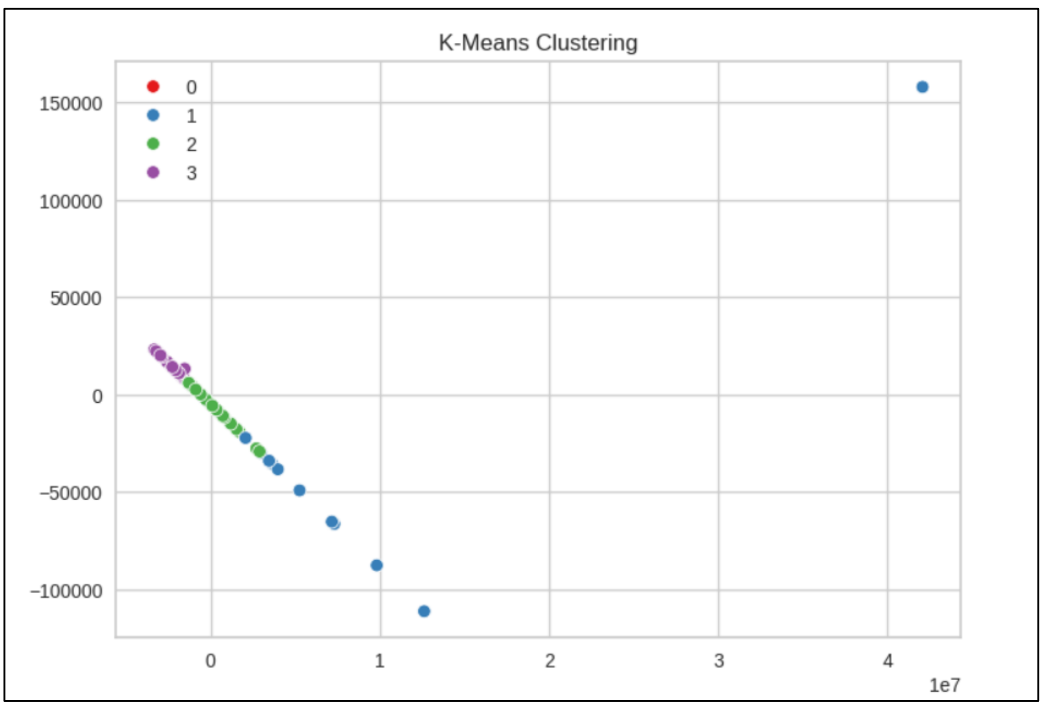
KMeans clustering is a popular unsupervised learning algorithm that partitions a dataset into k clusters. It works by iteratively assigning data points to the nearest cluster centroid and then updating the centroids based on the assigned points.

```
sns.scatterplot(data=data, x="PC1", y="PC9", hue=kmeans.labels_)
plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_centers_[:,1],
            marker="X", c="r", s=80, label="centroids")
plt.legend()
plt.show()
```

The above code is used to portray the formed clusters in the given dataset.



It can be explained that the nearest clusters formed for corresponding 0 and 0.5 to be close and 1.0 to be quite far from the actual data.



Prediction of Prices most used cars

Linear regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models targets prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Here we use a linear regression model to predict the prices of different Electric cars in different companies. X contains the independent variables and y is the dependent Prices that is to be predicted. We train our model with a splitting of data into a 4:6 ratio, i.e. 40% of the data is used to train the model.

```
X = data[['PC1', 'PC2','PC3','PC4','Pc5','PC6', 'PC7','PC8','PC9']]
y = ev['INR']
```

```
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.5, random_state = 1)
lin = lr.fit(X_train, y_train)
lin
```

```
LinearRegression
LinearRegression()
```

```
lin.intercept_
```

```
4983453.390582525
```

```
lin.coef_
```

```
array([-1.59573156e-09,  2.79396772e-09, -3.05590220e-09,  0.00000000e+00,
        7.27595761e-10,  3.49245965e-10,  2.52293830e-09,  5.00779098e+06,
        1.33877620e-09])
```

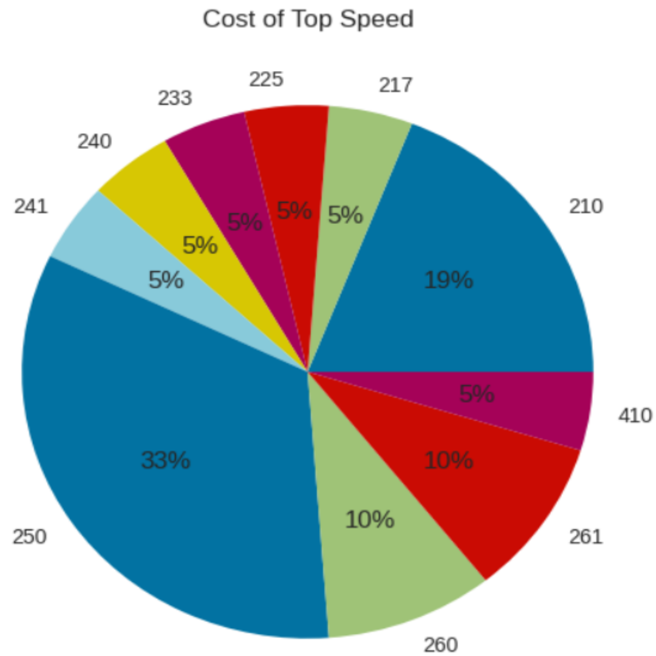
Profiling and Describing the Segments

Sorting the Top Speeds and Maximum Range in accordance to the Price with head () we can view the Pie Chart.

Pie Chart:

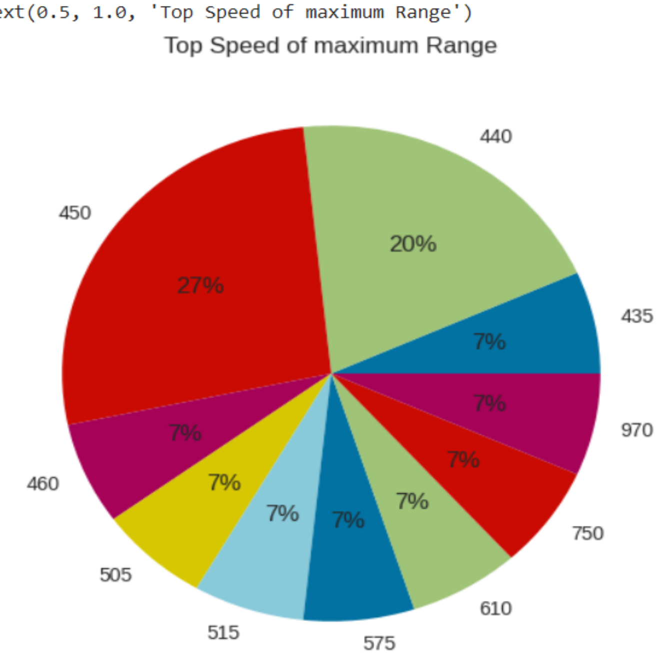
```
plt.figure(figsize = (10,6))
plt.pie(x = ev1['INR'].tail(10), labels = ev1.tail(10).index, autopct='%1.0f%%')
plt.title('Cost of Top Speed')
```

```
Text(0.5, 1.0, 'Cost of Top Speed')
```



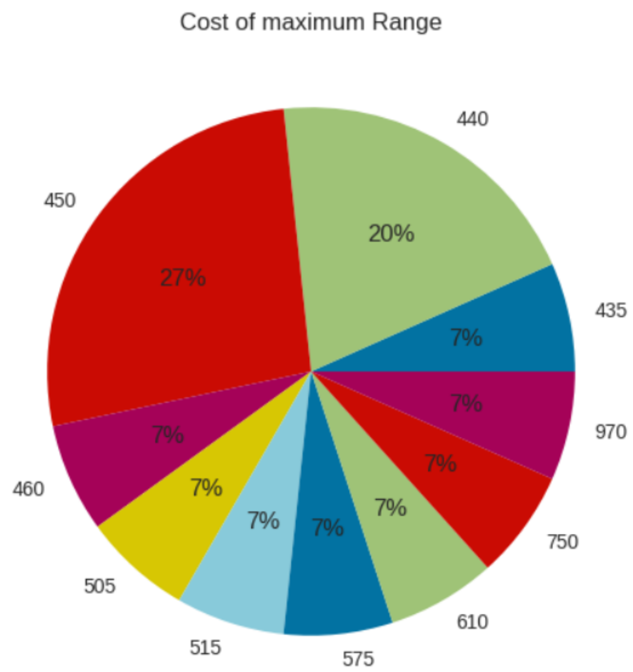

```
plt.figure(figsize = (10,6))
plt.pie(x = ev2['TopSpeed_KmH'].tail(10), labels = ev2.tail(10).index, autopct='%1.0f%%')
plt.title('Top Speed of maximum Range')
```

```
Text(0.5, 1.0, 'Top Speed of maximum Range')
```



```
[ ] plt.figure(figsize = (10,6))
    plt.pie(x = ev3['INR'].tail(10), labels = ev3.tail(10).index, autopct='%1.0f%%')
    plt.title('Cost of maximum Range')
```

```
Text(0.5, 1.0, 'Cost of maximum Range')
```



Conclusion:

Government Incentives: Government policies and incentives have played a crucial role in driving EV adoption in India.

Technological Advancements: Improvements in battery technology, charging infrastructure, and vehicle performance have made EVs more appealing to consumers.

The competition: The EV market in India is becoming increasingly competitive. You will need to find way to differentiate your startup from the competition.

Infrastructure Development: Continued investment in charging infrastructure is essential for expanding EV adoption, especially in rural areas.

Consumer Education: Efforts should be made to educate consumers about the benefits of EVs and address related to range, charging and cost.

In short, by understanding the diverse consumer segments and addressing the challenges and opportunities, stakeholders can contribute to the successful development of the Indian EV Market.

Link to Project:

Dataset 1:

https://github.com/payalchauhan9/Electric_Vehicle/blob/main/ev.csv

Dataset2:

https://github.com/payalchauhan9/Electric_Vehicle/blob/main/customer_dataset.csv

Github Link:

https://github.com/payalchauhan9/Electric_Vehicle