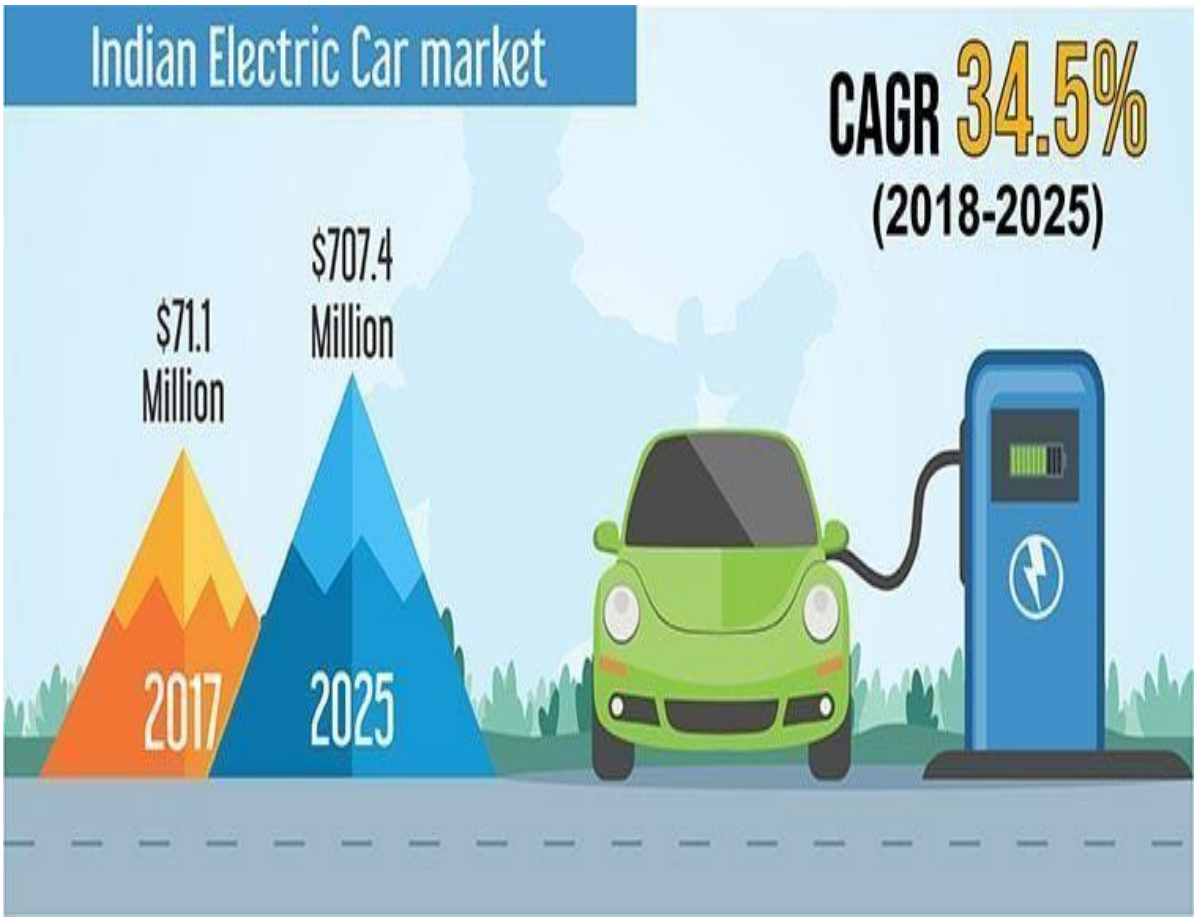
ELECTRIC VEHICLES MARKET

Market segmentation

BATTALA RADHIKA



Abstract:

Market segmentation becomes a crucial tool for evolving transportation technology such as electric vehicles (EVs) in emerging markets to explore and implement for extensive adoption. EVs adoption is expected to grow phenomenally in near future as low emission and low operating cost vehicle, and thus, it drives a considerable amount of forthcoming academic research curiosity. The main aim of this study is to explore and identify distinct sets of potential buyer segments for EVs based on *psychographic, behavioral, and socio-economic* characterization by employing an integrated research framework of *'perceived benefits-attitude-intention'*. The study applied robust analytical procedures including cluster analysis, multiple discriminant analysis and Chi-square test to operationalize and validate segments from the data collected of 563 respondents using a cross-sectional online survey. The findings posit that the three distinct sets of young consumer groups have been identified and labelled as *'Conservatives', 'Indifferents', and 'Enthusiasts'* which are deemed to be buddying EV buyers The implications are recommended, which may offer some pertinent guidance for scholars and policy makers to encourage EVs adoption in the backdrop of emerging sustainable transport market. In this report we are going to analyse the data and solve the problem using **Fermi Estimation** by breaking down the problem.

KeyWords : *Electric vehicles, Market segmentation, Cluster analysis, Attitude to wards electric vehicles, Subjective norms, Adoption intention, Sustainable transporta*ction.

What is Electric Vehicle?

An EV is a shortened acronym for an electric vehicle. EVs are vehicles that are either partially or fully powered on electric power. Electric vehicles have low running costs as they have less moving parts for maintaining and also very environmentally friendly as they use little or no fossil fuels (petrol or diesel).

While some EVs used lead acid or nickel metal hydride batteries, the standard for modern battery electric vehicles is now considered to be lithium ion batteries as they have a greater longevity and are excellent at retaining energy, with a selfdischarge rate of just 5% per month. Despite this improved efficiency, there are still challenges with these batteries as they can experience thermal runaway, which have, for example, caused fires or explosions in the Tesla model S, although efforts have been made to improve the safety of these batteries.

Working principle

An electric vehicle works on a basic principle of science: **conversion of energy**. Electrical energy is converted into mechanical energy. There is a motor used in the electrical system to carry on this duty of conversion. Motors can be of various types.

Data Collection

The data has been collected manually, and the sources used for this process are listed

below :

• https://www.kaggle.com/datasets

• https://data.gov.in/

• https://www.data.gov/

Market Segmentation

Target Market:

The target market of Electric Vehicle Market Segmentation can be categorized into Geographic, SocioDemographic, Behavioral, and Psychographic Segmentation.

Behavioral Segmentation: searches directly for similarities in behavior or reported behavior.

Example: prior experience with the product, amount spent on the purchase, etc.

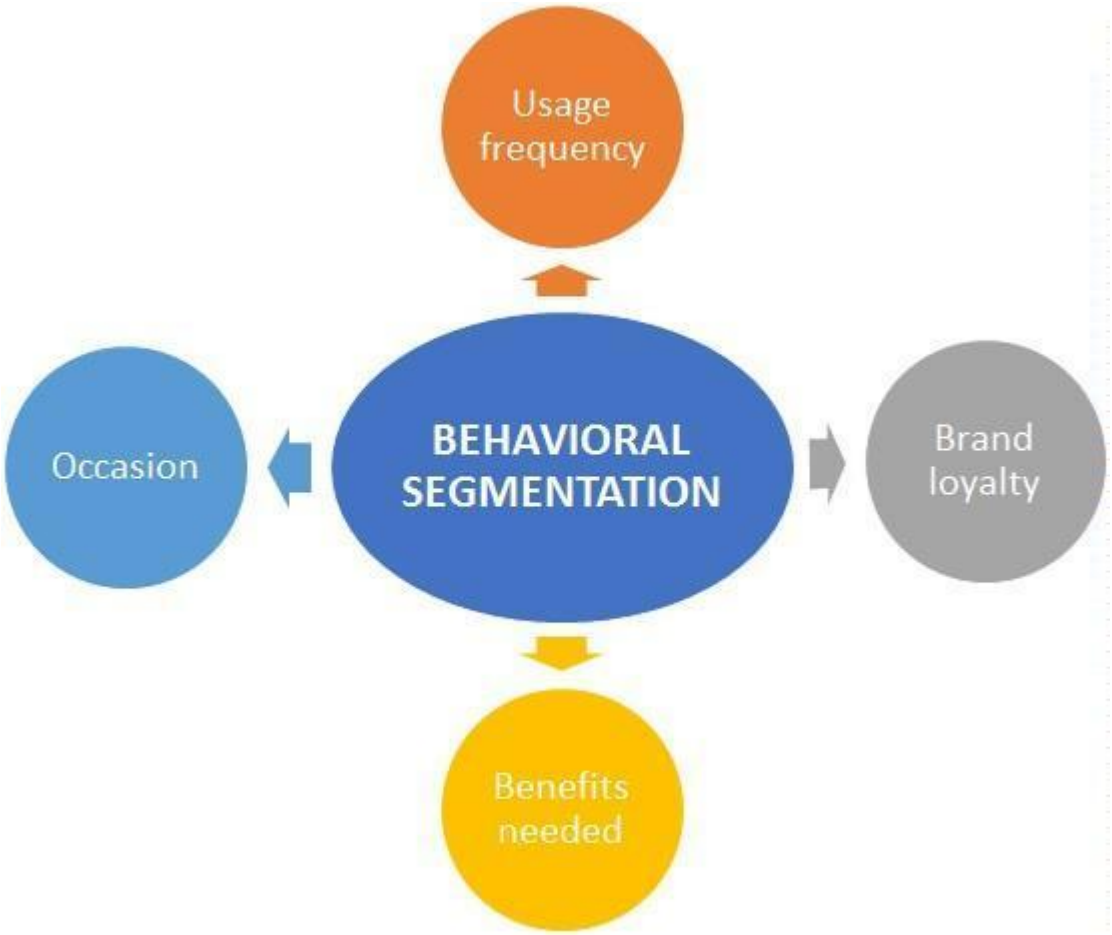


Figure 1: *Behavioral Segmentation*

Advantage: uses the very behavior of interest is used as the basis of segment extraction.

Disadvantage: not always readily available.

Psychographic Segmentation: grouped based on beliefs, interests, preferences, aspi rations, or benefits sought when purchasing a product. Suitable for lifestyle segmentation.involve many segmentation variables.

Advantage: generally more reflective of the underlying reasons for differences in consumer behavior.

Disadvantage: increased complexity of determining segment memberships for consumers.

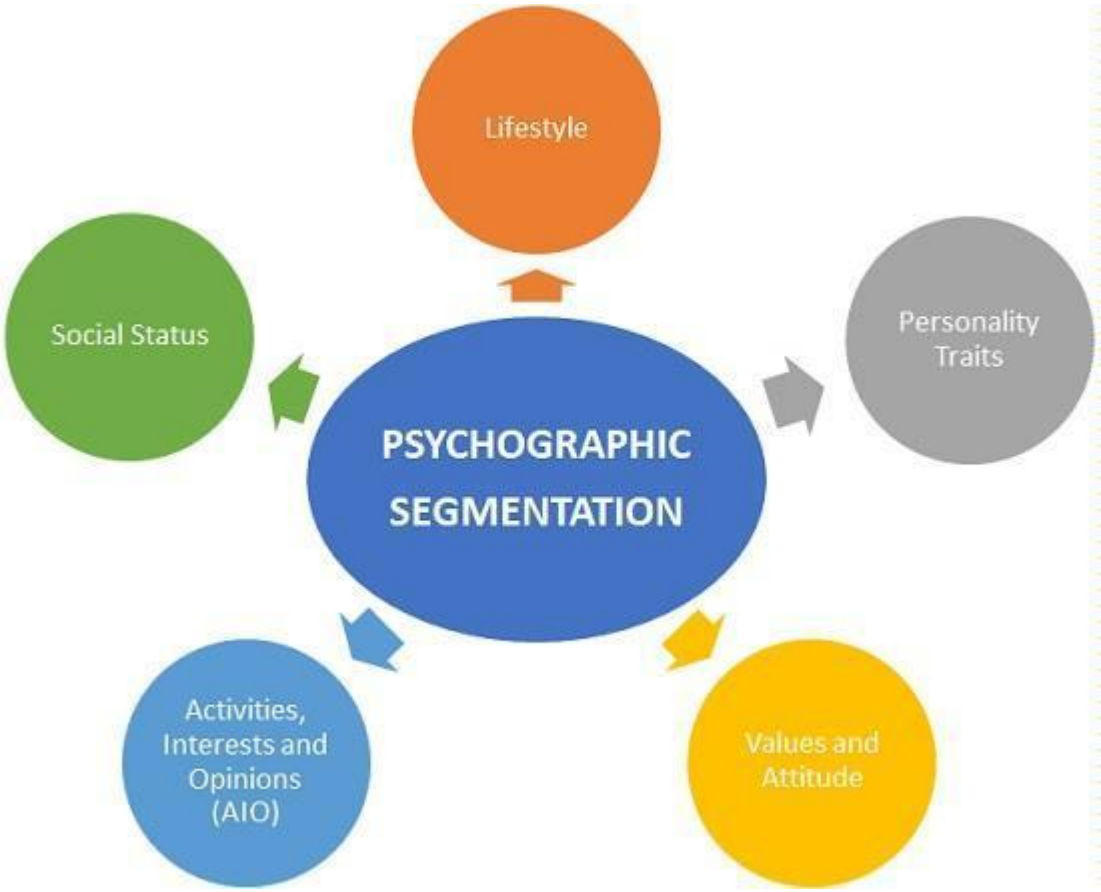


Figure 2: *Psychographic Segmentation*

Socio-Demographic Segmentation: includes age, gender, income and education.

Useful in industries

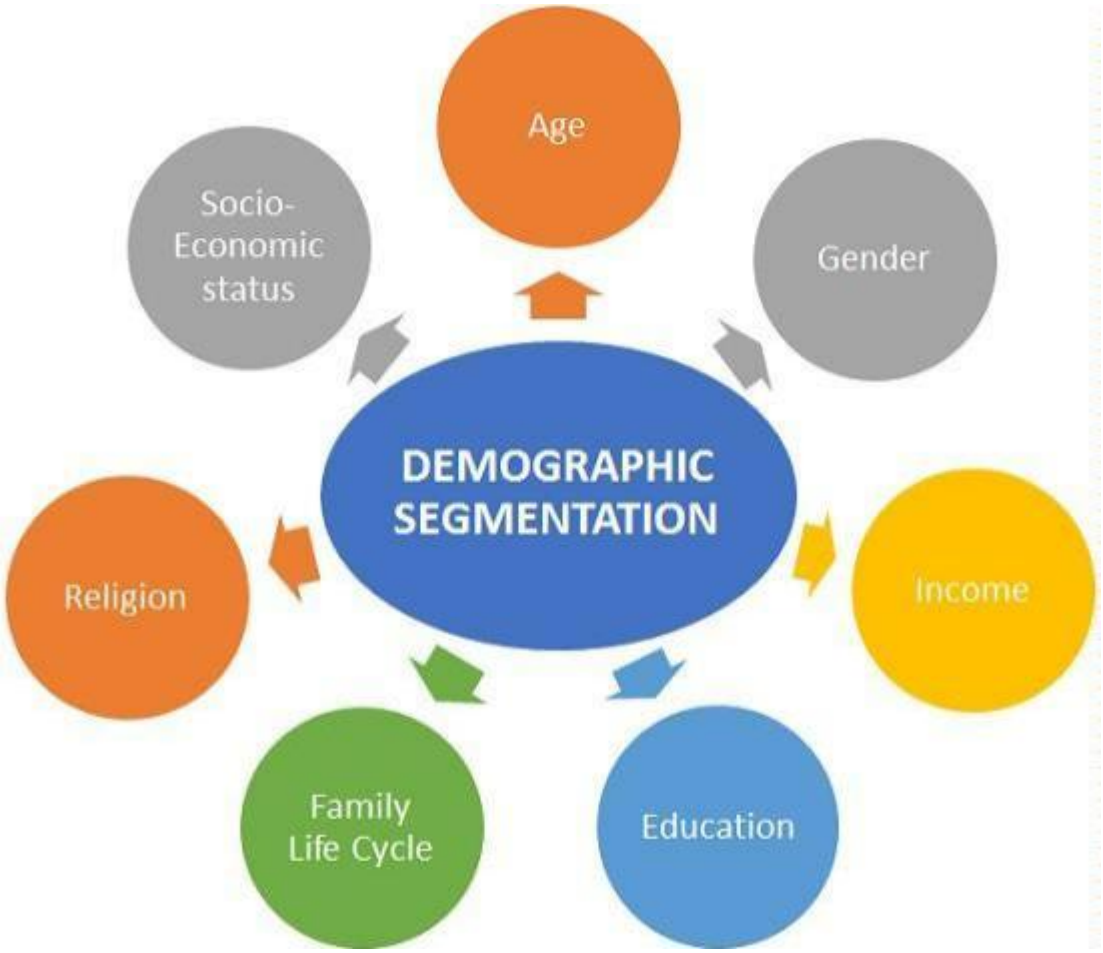


Figure 3: *Behavioral Segmentation*

Advantage: segment membership can easily be determined for every customer.

Disadvantage: if this criteria is not the cause for customers product preferences then

it does not provide sufficient market insight for optimal segmentation decisions.

Segmenting for Electric Vehicle Market

The market segmentation approach aims at defining actionable, manageable, homogenous subgroups of individual customers to whom the marketers can target with a similar set of marketing strategies. In practice, there are two ways of segmenting the market-a-priori and post-hoc. An a-priori approach utilizes predefined characteristics such as age, gender, income, education, etc. to predefine the segments followed by pro filing based on a host of measured

variables *(behavioral, psychographic or benefit)*. In the post-hoc approach to segmentation on other hand, the segments are identified based on the relationship among the multiple measured variables. The commonality between both approaches lies in the fact that the measured variables determine the *'segmentation theme'*. The present study utilizes an a-priori approach to segmentation o as to divide the potential EV customers into sub-groups.

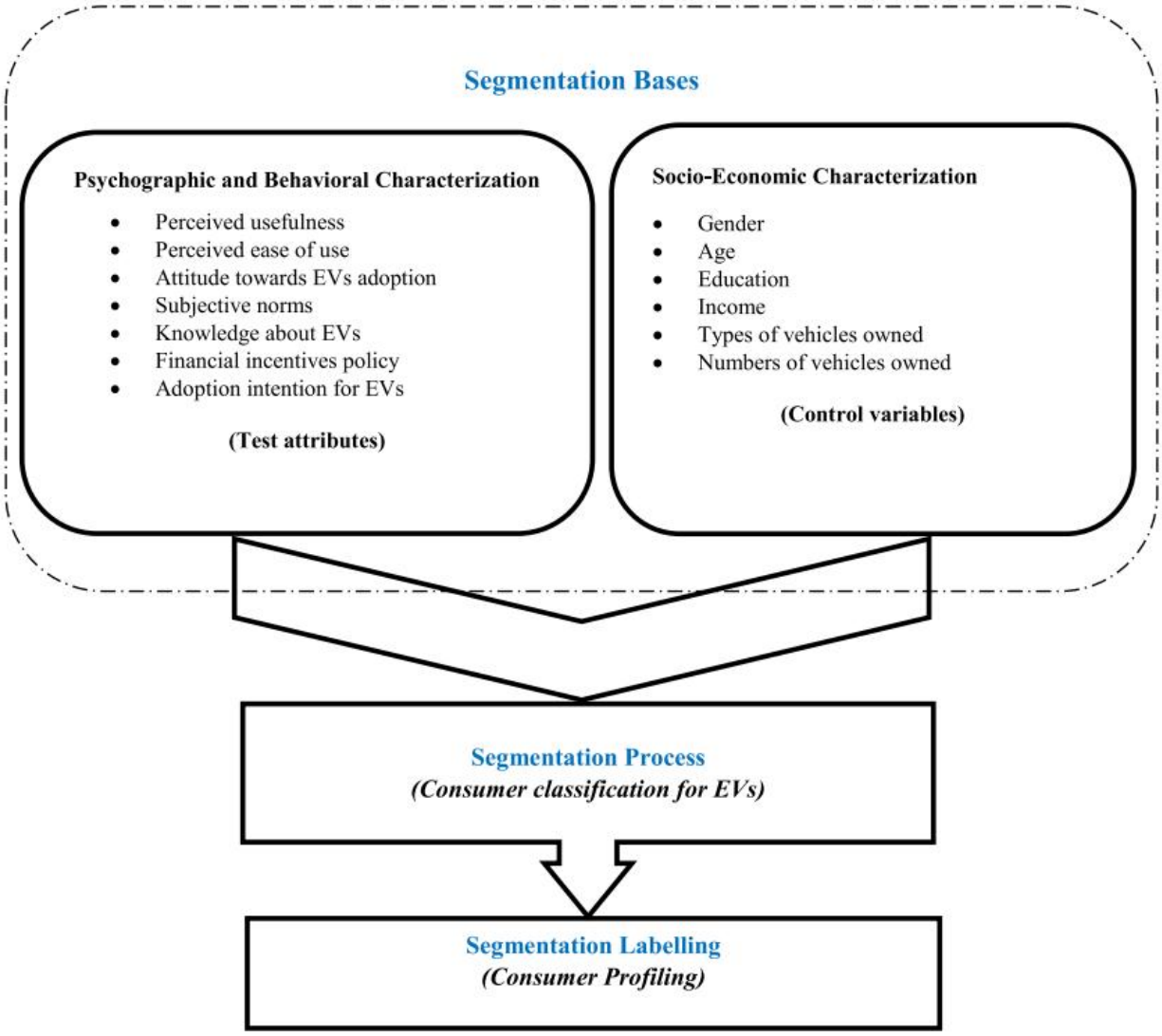


Figure 4: *Market Segmentation Electric Vehicles*

It is argued that the blended approach of *psychographic* and *socioeconomic attributes* for market segmentation enables the formulation of sub-market

strategies which in turn satisfy the specific tastes and preferences of the consumer groups. Straughan and Roberts presented a comparison between the usefulness of *psychographic, demographic, and economic* characteristics based on consumer evalua tion for eco-friendly products.

They pinpointed the perceived superiority of the psychographic characteristics in profiling the consumer segments in the market for eco-friendly products. The present study adds perceived-benefit characteristics guided by blended psychographic and socio-economic aspects for segmenting the consumer market.

Code 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.

Importing Necessary Libraries

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sbn
        import os
        import warnings
```

Fig 1: Importing Libraries for Code Implementation

1. NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.
2. Pandas is a library written for the Python programming language for data manipulation and analysis
3. Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays.
4. Seaborn is an open-source Python library built on top of matplotlib. It is used for data visualization and exploratory data analysis.

5. Warnings are provided to warn the developer of situations that aren't necessarily exceptions

Reading Data

```
data = pd.read_csv(file_path)
print(data)
```

```
          Brand                              Model  AccelSec  TopSpeed_KmH  \
0         Tesla    Model 3 Long Range Dual Motor       4.6           233
1    Volkswagen                      ID.3 Pure       10.0           160
2      Polestar                              2       4.7           210
3           BMW                             iX3       6.8           180
4         Honda                               e       9.5           145
..          ...                             ...       ...           ...
97       Nissan                 Ariya 63kWh          7.5           160
98         Audi    e-tron S Sportback 55 quattro       4.5           210
99       Nissan             Ariya e-4ORCE 63kWh       5.9           200
100      Nissan  Ariya e-4ORCE 87kWh Performance       5.1           200
101       Byton               M-Byte 95 kWh 2WD       7.5           190

     Range_Km  Battery_Pack Kwh  Efficiency_WhKm  FastCharge_KmH RapidCharge  \
0         460              70.0              161             940         Yes
1         270              45.0              167             250         Yes
2         400              75.0              181             620         Yes
3         360              74.0              206             560         Yes
4         170              28.5              168             190         Yes
..        ...               ...              ...             ...         ...
97        330              63.0              191             440         Yes
98        335              86.5              258             540         Yes
99        325              63.0              194             440         Yes
100       375              87.0              232             450         Yes
101       400              95.0              238             480         Yes

    PowerTrain    PlugType  BodyStyle Segment  Seats  PriceEuro          INR
0          AWD  Type 2 CCS      Sedan       D      5      55480  4540988.068
1          RWD  Type 2 CCS  Hatchback       C      5      30000  2455473.000
2          AWD  Type 2 CCS   Liftback       D      5      56440  4619563.204
3          RWD  Type 2 CCS        SUV       D      5      68040  5569012.764
4          RWD  Type 2 CCS  Hatchback       B      4      32997  2700774.753
..         ...          ...       ...     ...    ...        ...          ...
97         FWD  Type 2 CCS  Hatchback       C      5      45000  3683209.500
```

Fig 2: Dataset used for Code Implementation

Analysing the Dataset

```
In [14]: data.columns

Out[14]: Index(['Brand', 'Model', 'AccelSec', 'TopSpeed_KmH', 'Range_Km',
                'Battery_Pack Kwh', 'Efficiency_WhKm', 'FastCharge_KmH', 'RapidCharge',
                'PowerTrain', 'PlugType', 'BodyStyle', 'Segment', 'Seats', 'PriceEuro',
                'INR'],
               dtype='object')
```

Fig 3: columns of the Data set

```
In [5]: data.describe()
```

	AccelSec	TopSpeed_KmH	Range_Km	Battery_Pack Kwh	Efficiency_WhKm	FastCharge_KmH	Seats	PriceEuro	INR
count	102.000000	102.000000	102.000000	102.000000	102.000000	102.000000	102.000000	102.000000	1.020000e+02
mean	7.391176	179.313725	338.627451	65.415686	189.303922	435.686275	4.882353	55997.588235	4.583352e+06
std	3.031913	43.771228	126.700623	29.955782	29.679072	220.447384	0.799680	34250.724403	2.803391e+06
min	2.100000	123.000000	95.000000	16.700000	104.000000	0.000000	2.000000	20129.000000	1.647541e+06
25%	5.100000	150.000000	250.000000	43.125000	168.000000	260.000000	5.000000	34414.750000	2.816816e+06
50%	7.300000	160.000000	340.000000	64.350000	180.500000	440.000000	5.000000	45000.000000	3.683210e+06
75%	9.000000	200.000000	400.000000	83.700000	204.500000	557.500000	5.000000	65000.000000	5.320192e+06
max	22.400000	410.000000	970.000000	200.000000	273.000000	940.000000	7.000000	215000.000000	1.759756e+07

Fig 4: Information

Checking for Null values in the dataset

```
In [13]: data.isnull().sum()

Out[13]: Brand                0
         Model                0
         AccelSec             0
         TopSpeed_KmH         0
         Range_Km             0
         Battery_Pack Kwh     0
         Efficiency_WhKm      0
         FastCharge_KmH       0
         RapidCharge          0
         PowerTrain           0
         PlugType             0
         BodyStyle            0
         Segment              0
         Seats                0
         PriceEuro            0
         INR                  0
         dtype: int64
```

Fig 5: Checking for the null values in the Data set

Visualization

```
In [10]: def train(dataframe):
             sbn.countplot(x=dataframe['PowerTrain'])
             plt.title('Count Plot of a Powertrain')
             plt.xlabel('PowerTrain')
             plt.ylabel('Count')

         train(data)
```

Out[10]:

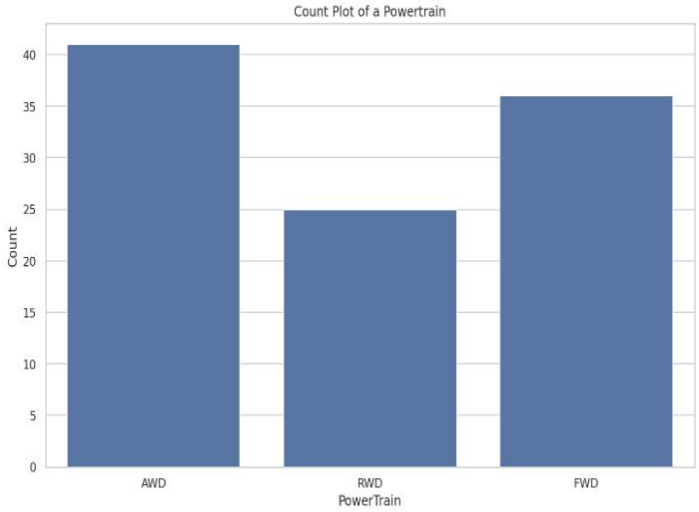


Fig 6: Count Plot of a Powertrain

```
In [13]: def bodystyle(dataframe):
             plt.figure(figsize=(10,5))
             sbn.countplot(x='BodyStyle', data=dataframe, hue='PowerTrain')
             plt.title('Count plot of Body Style')
             plt.xlabel('Body Style')
             plt.ylabel('Count')
             plt.show()

         bodystyle(data)
```

Out[13]:

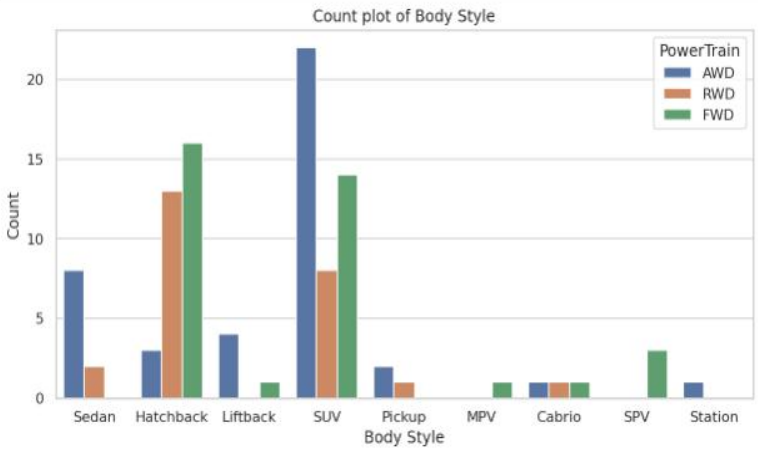


Fig 7: Count plot of body Style of the cars

```
In [18]: import matplotlib.pyplot as plt
import seaborn as sbn
import pandas as pd
data = pd.DataFrame({
    'Brand': ['A', 'B', 'C', 'D'],
    'Range_Km': [100, 200, 300, 150]
})

ax = plt.figure(figsize=(20, 5))
sbn.barplot(x='Brand', y='Range_Km', data=data, palette='tab10')
plt.grid(axis='y')
plt.title('Maximum Range achieved by a brand')
plt.xlabel('Brand')
plt.ylabel('Range')
plt.xticks(rotation=45)
plt.show()
```

Out[18]:

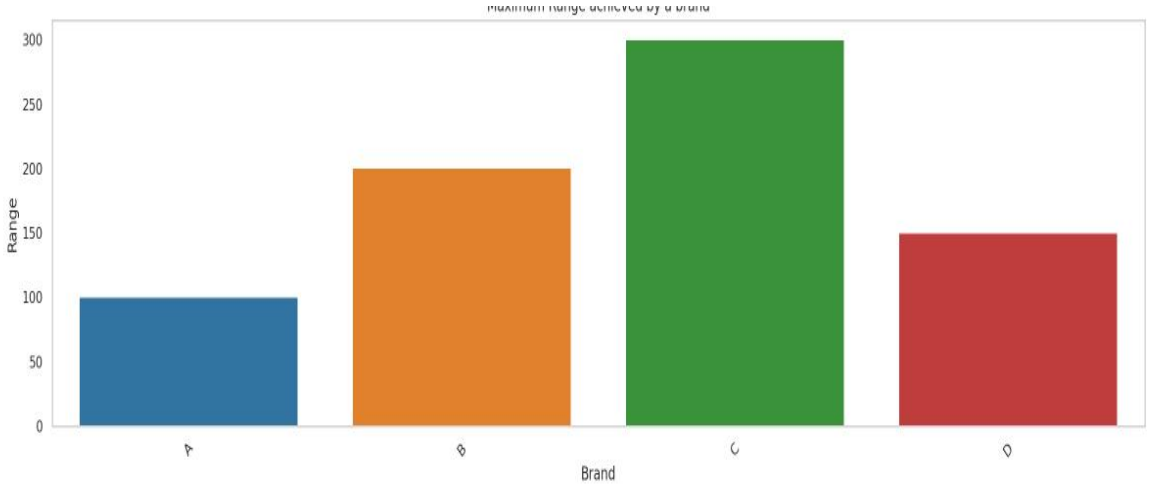


Fig 8: Bar graph of Range of EV's

```
In [13]: file_path = 'ElectricCarData_Clean_Me.csv'
data = pd.read_csv(file_path)
data['CarName'] = data['Brand'] + '-' + data['Model']
data1= data.loc[data['INR'] <=4000000]
data2 = data.loc[data['INR'] >4000000]
t1 = ['Less than INR 4000000']
t2 = ['More thanINR 4000000']

def range(dataframe, price):
    plt.figure(figsize=(20,5))
    sbn.set_theme(style="whitegrid")
    sbn.barplot(x='Model', y='Range_Km', data=data, hue=dataframe['PowerTrain'])
    plt.title('''Range(Km) of EV's costing{}'''.format(price))
    plt.ylabel('Range(Km)')
    plt.xlabel('Model')
    plt.xticks(rotation=90)
    plt.show()

range(data1, t1)
range(data2, t2)
```

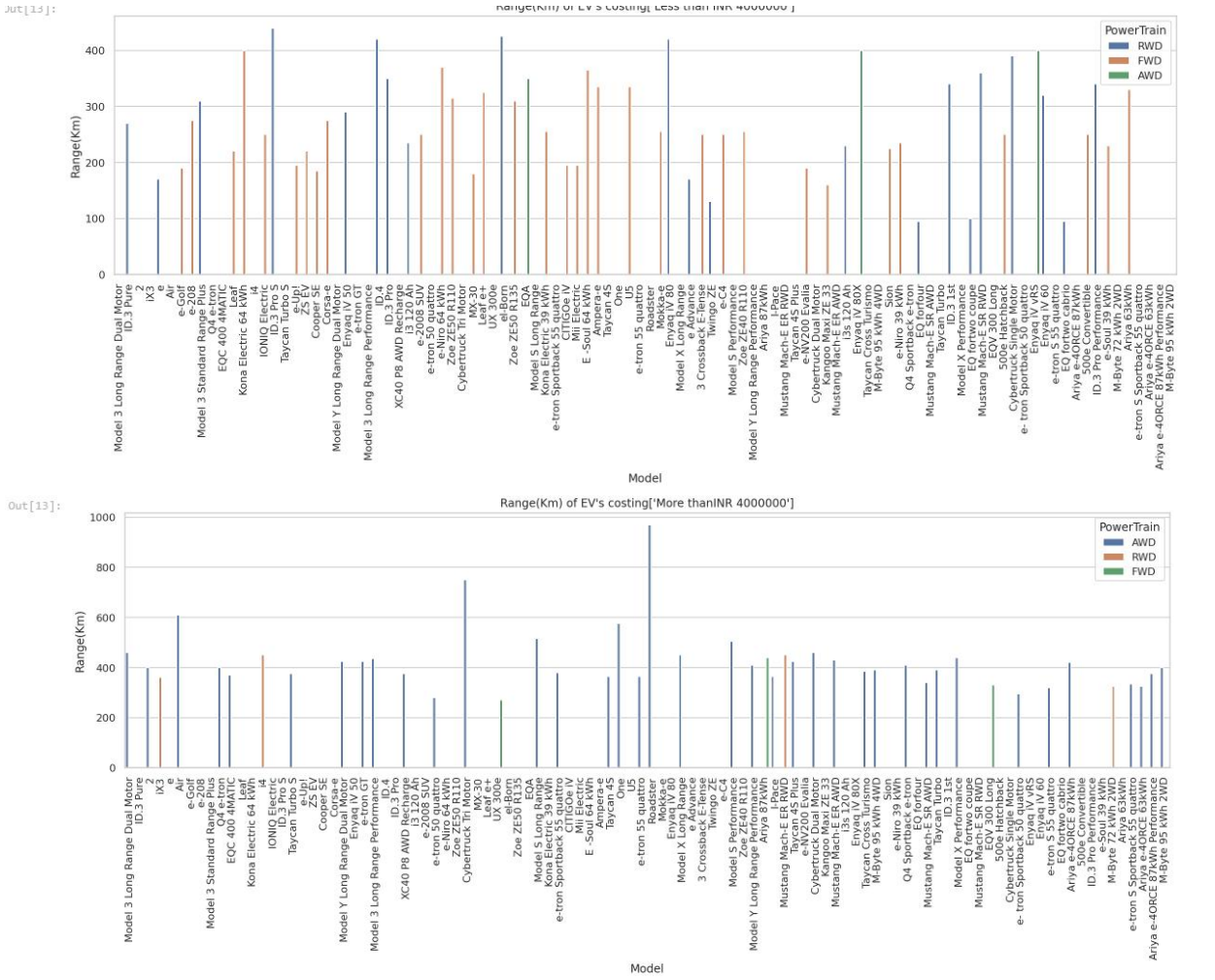


Fig 9: Bar graph of Range of EV'S

In [26]:

```
file_path = 'ElectricCarData_Clean_Me.csv'
data = pd.read_csv(file_path)
data['CarName'] = data['Brand'] + '-' + data['Model']
data1= data.loc[data['INR'] <=4000000]
data2 = data.loc[data['INR'] >7000000]
t1 = ['Less than INR 4000000']
t2 = ['More thanINR 4000000']
def range_price(dataframe, text):
    fig = plt.figure(figsize=(20, 6))
    a1 = plt.subplot()
    a1.bar(data['CarName'], data['Range_Km'], label='Range (Km)', color='blue')
    plt.legend(loc='upper left', bbox_to_anchor = (0, 1.1))
    a2= a1.twinx()
    a2.scatter(data['CarName'], data['INR'], label = 'Price', color = 'black')
    plt.title('''RANGE (Km) vs PRICE(INR) OF EV's COSTING {}'''.format(text), fontsize=16)
    a1.set_xlabel('Models', size=16)
    a1.set_ylabel('Range (Km)', color = 'red')
    a2.set_ylabel('Price(INR)', color= 'black')
    plt.legend(loc = 'upper left', bbox_to_anchor = (0,1.1))
    a1.set_xticklabels(data['CarName'], rotation = 'vertical')
    plt.show()

range_price(data1, t1)
range_price(data2 ,t2)
```

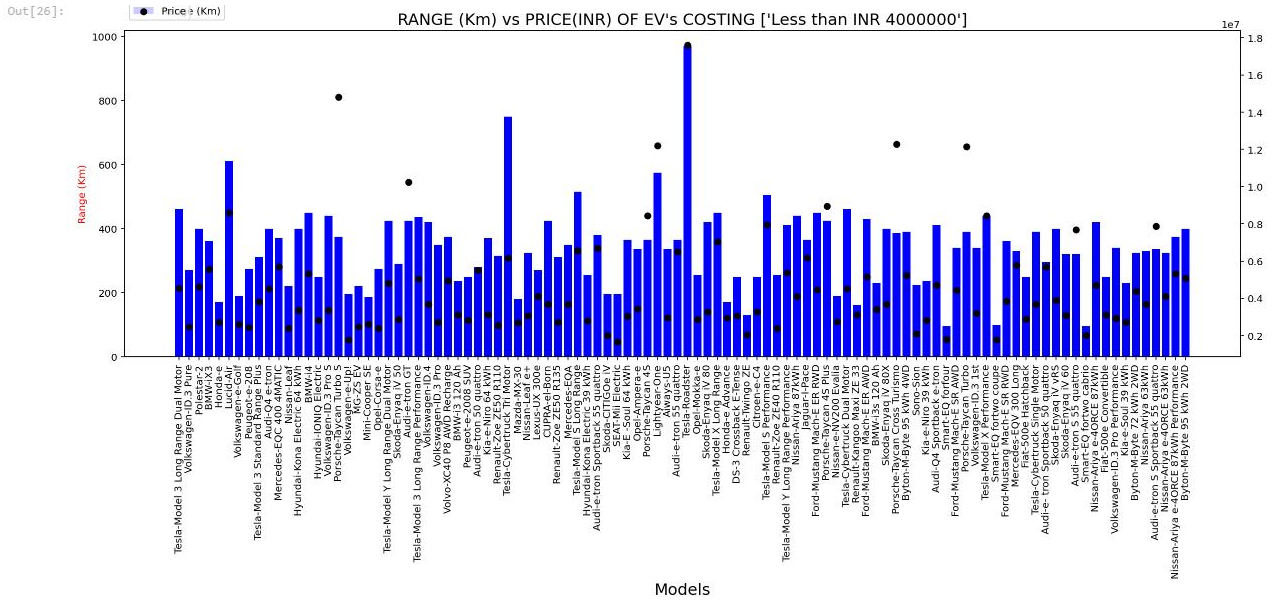


Fig 10: Bar graph of Range vs Price of EV's

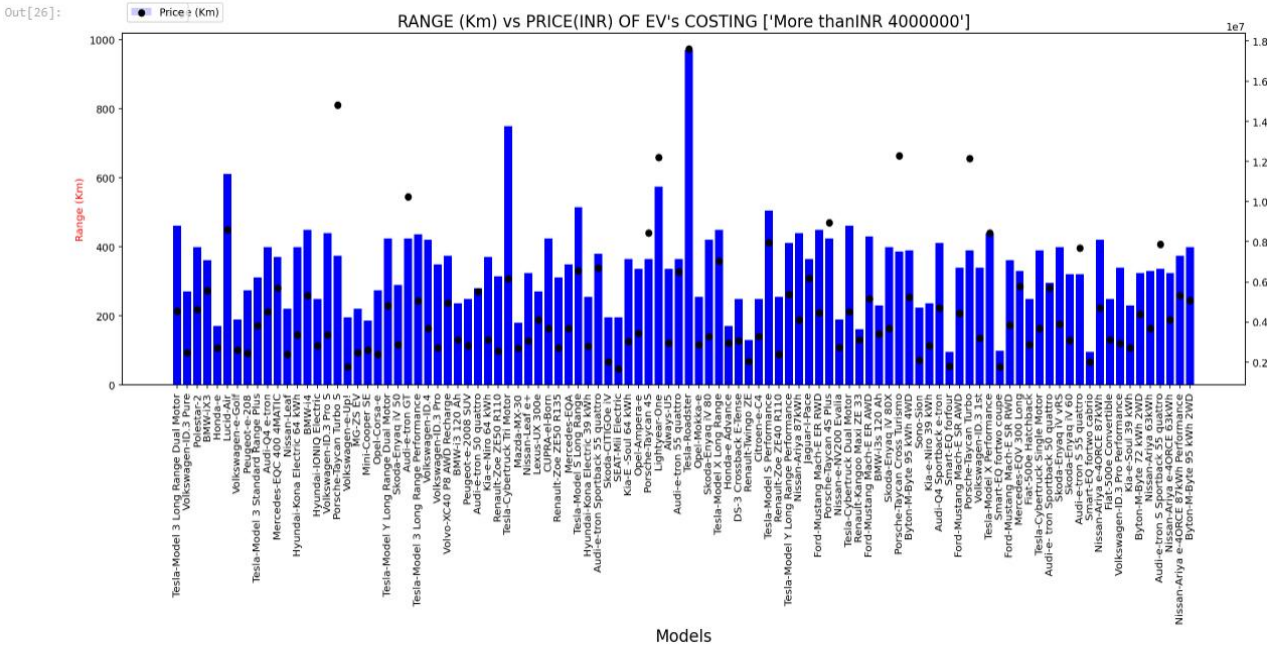


Fig 11: Bar graph of Range vs Price of EV's

In [8]:
```
#Accelaration(0-100km/hr)
import seaborn as sbn
file_path = 'ElectricCarData_Clean_Me.csv'
data = pd.read_csv(file_path)
data['CarName'] = data['Brand'] + '-' + data['Model']
data1= data.loc[data['INR'] <=4000000]
data2 = data.loc[data['INR'] >7000000]
t1 = ['Less than INR 4000000']
t2 = ['More thanINR 4000000']
def acc(dataframe, text):
    plt.figure(figsize=(20,5))
    sbn.set_theme(style="darkgrid")
    sbn.barplot(x='CarName', y='AccelSec', hue='PowerTrain', data=dataframe)
    plt.title('''Acceleration 0-100 Km of EV's costing {}'''.format(text), fontsize=16)
    plt.ylabel('Acceleration (Seconds)')
    plt.xlabel('Model')
    plt.xticks(rotation=90)
    plt.show()

acc(data1, t1)
acc(data2, t2)
```

Out[8]:

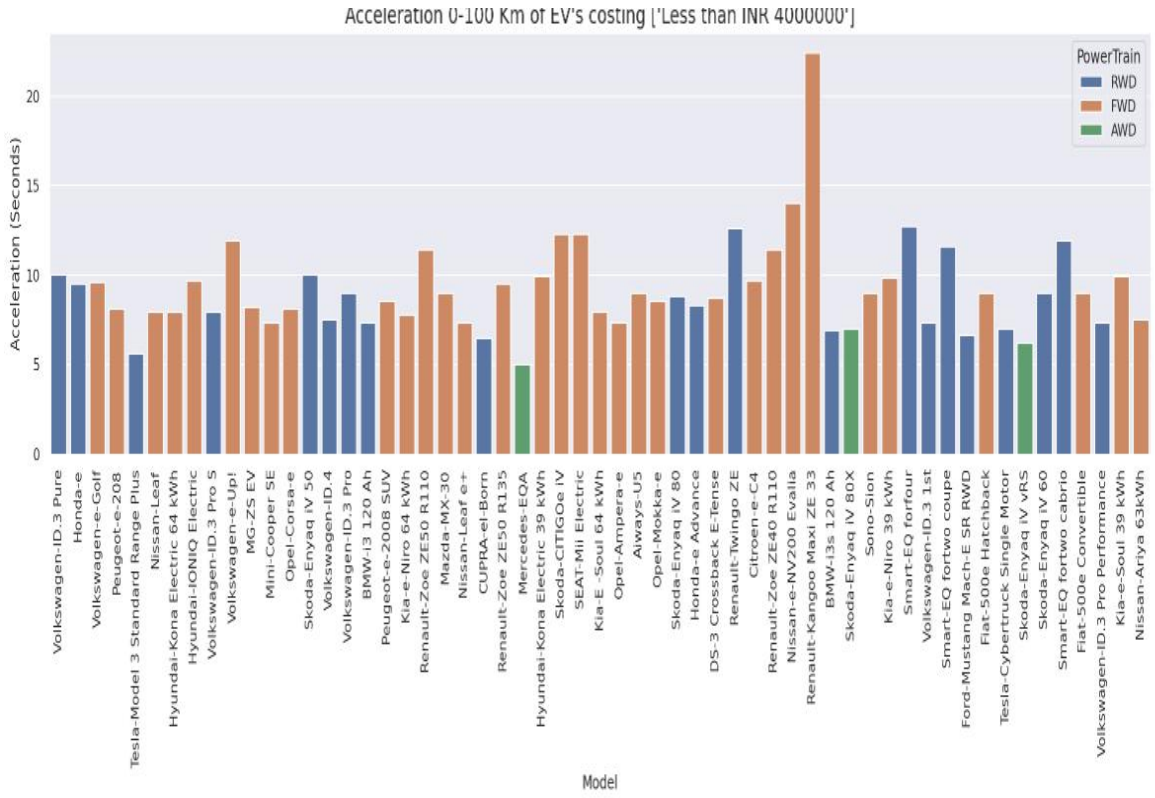


Fig 12: Bar graph of Acceleration vs Price of EV's

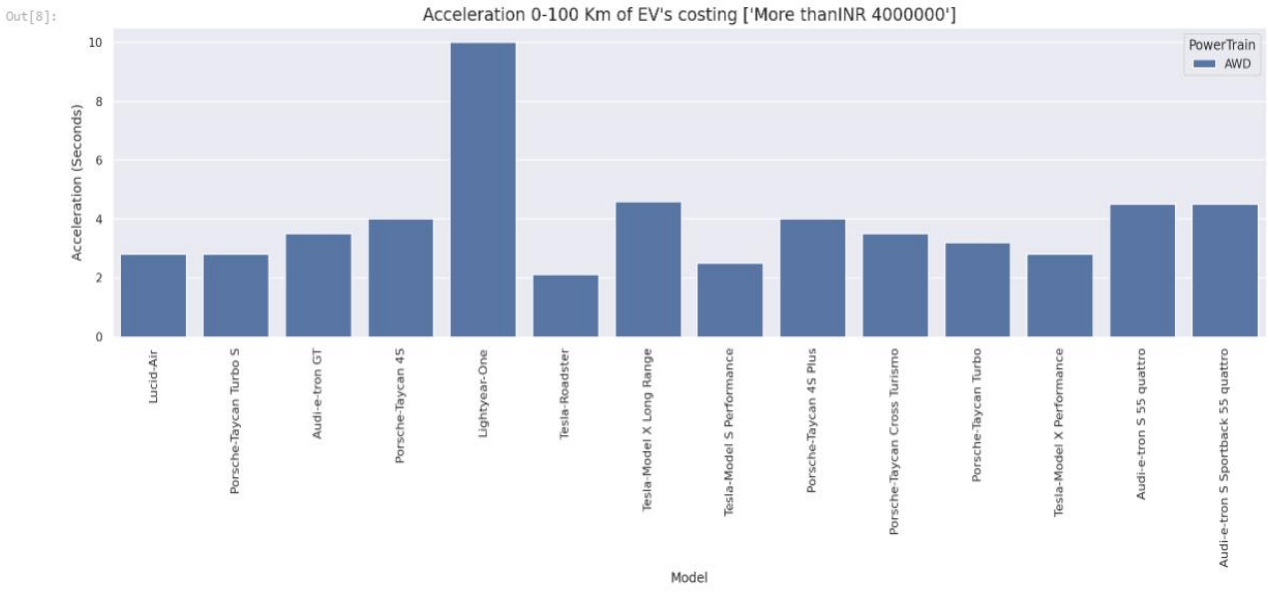


Fig 13: Bar graph of Acceleration vs Price of EV's

```
In [9]: # Fast Charging data
def fastcharge(dataframe, price):
    plt.figure(figsize=(20, 5))
    sbn.set_theme(style="whitegrid")
    sbn.barplot(data=dataframe,x='CarName', y='FastCharge_KmH', color = 'lightslategrey')
    plt.title('''Fast Charging of EV's costing {}'''.format(price), fontsize = 16)
    plt.ylabel('Charging Capacity (KmH)')
    plt.xlabel('Model')
    plt.xticks(rotation=90)
    plt.show()

fastcharge(data1, t1)
fastcharge(data2, t2)
```

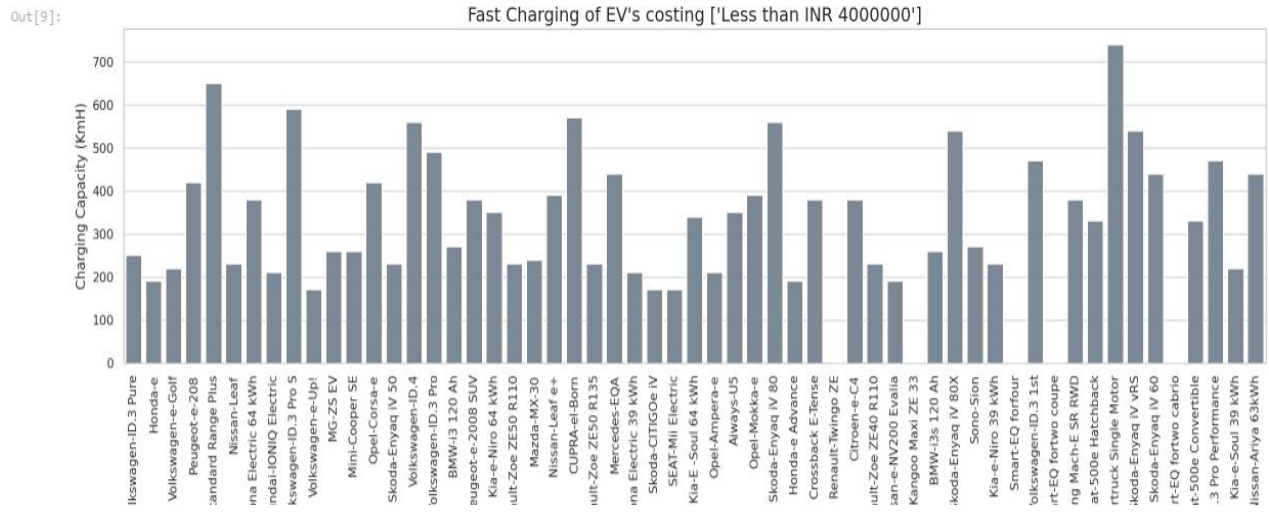


Fig 14: Bar graph of Fast Charging ability of EV'

```
In [39]: pd.set_option('display.max_columns', None)
         top_range_1 = data.sort_values(by= 'Range_Km', ascending= False)
         print(top_range_1[['CarName', 'Range_Km', 'Battery_Pack Kwh', 'INR', 'RapidCharge']])
```

```
                        CarName  Range_Km  Battery_Pack Kwh          INR  \
51                Tesla-Roadster       970             200.0  1.759756e+07
33   Tesla-Cybertruck Tri Motor       750             200.0  6.138682e+06
5                   Lucid-Air         610             110.0  8.594156e+06
48                Lightyear-One       575              60.0  1.219552e+07
40     Tesla-Model S Long Range       515              90.0  6.547110e+06
..                          ...       ...               ...           ...
68   Renault-Kangoo Maxi ZE 33        160              31.0  3.110266e+06
57           Renault-Twingo ZE        130              21.3  2.029039e+06
82       Smart-EQ fortwo coupe        100              16.7  1.750507e+06
91      Smart-EQ fortwo cabrio         95              16.7  2.010623e+06
77             Smart-EQ forfour        95              16.7  1.803136e+06

   RapidCharge
51         Yes
33         Yes
5          Yes
48         Yes
40         Yes
..         ...
68          No
57          No
82          No
91          No
77          No

[102 rows x 5 columns]
```

Fig 15: Vehicles to buy under INR 40,00000

```
In [48]: data.plot(figsize = (12,8),title='Car Price',xlabel = 'No. of Samples',ylabel = 'Car Price',color = 'red')
```

```
Out[48]: <Axes: title={'center': 'Car Price'}, xlabel='No. of Samples', ylabel='Car Price'>
```

Out[48]:

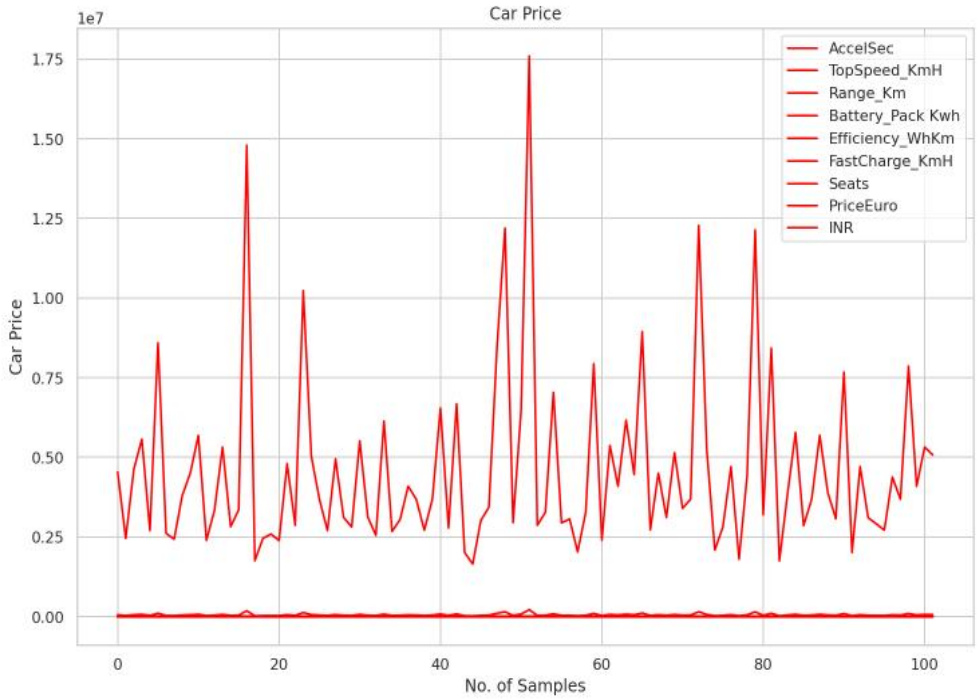


Figure 16: Car price

```
In [21]: from scipy.cluster.hierarchy import linkage, dendrogram
         data = pd.get_dummies(data, drop_first=True).values  # drop_first to avoid dummy variable trap

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

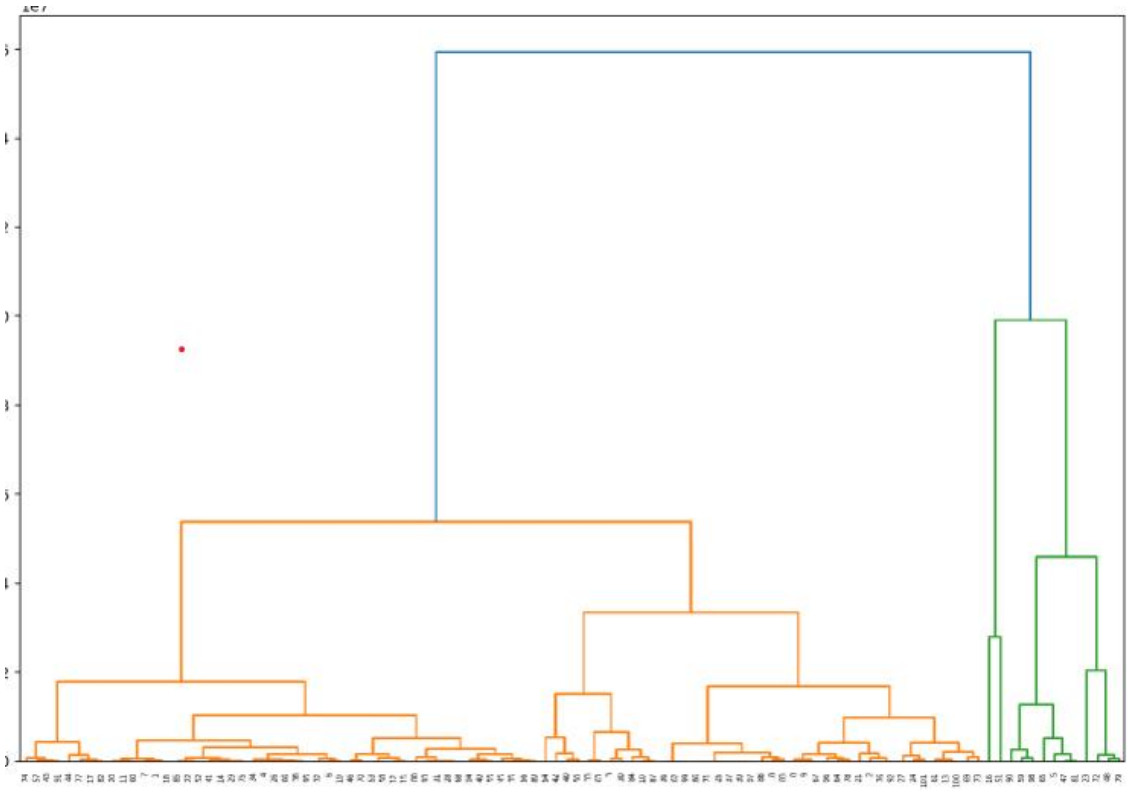


Figure 17: *Dendrogram Plot for our Dataset*

State wise Pollution Data Analysis

Importing Necessary Libraries

```
In [5]: import pandas as pd
        import numpy as np
        import seaborn as sns
        import matplotlib.pyplot as plt
        import os
```

Fig 1: importing libraries

Reading the data

```
In [7]: data=pd.read_csv("Pollution.csv")
        data
```

Out[7]:

	state	status	AQI-US	PM2.5	PM10	Temp	Humid
0	Andhra Pradesh	MODERATE	56	16	31	28	74
1	Arunachal Pradesh	GOOD	39	11	17	21	100
2	Assam	GOOD	46	13	20	23	98
3	Bihar	MODERATE	87	28	53	31	58
4	Chandigarh	POOR	107	38	49	25	53
5	Chhattisgarh	MODERATE	67	20	46	27	72
6	Dadra And Nagar Havel	MODARATE	62	16	35	27	82
7	Daman And Diu	MODERATE	61	16	33	28	79
8	Delhi	POOR	108	37	113	29	58
9	Goa	GOOD	30	8	20	27	81
10	Gujarat	MODERATE	68	20	42	30	68
11	Haryana	ODERATE	100	35	73	28	64
12	Himachal Pradesh	MODERATE	76	21	46	14	73
13	Jammu And Kashmir	MODERATE	64	15	38	13	86
14	Jharkhand	MODERATE	78	22	52	27	71
15	Karnataka	GOOD	40	10	29	23	82
16	Kerala	MODERATE	60	19	39	25	87
17	Madhya Pradesh	MODERATE	57	14	53	27	69
18	Maharashtra	MODERATE	62	16	51	20	76
19	Manipur	GOOD	28	7	12	20	98
20	Meghalaya	MODERATE	55	16	23	19	98
21	Mizoram	GOOD	14	2	5	17	100
22	Nagaland	GOOD	25	6	10	26	97
23	Odisha	MODERATE	79	25	42	28	84
24	Puducherry	MODERATE	54	15	31	27	80
25	Punjab	MODERATE	73	22	51	29	57
26	Rajasthan	POOR	107	38	67	18	61
27	Sikkim	MODERATE	70	21	54	27	96
28	Tamil Nadu	MODERATE	61	17	36	29	77
29	Telangana	MODERATE	69	21	31	24	62
30	Tripura	GOOD	14	2	5	29	96
31	Uttar Pradesh	MODERATE	96	34	85	29	61
32	Uttarakhand	POOR	108	38	59	20	67
33	West Bengal	MODERATE	68	20	45	26	84

Fig2: Data set used for code implementation

Checking for null values in the data set

```
In [8]: import pandas as pd
        data=pd.read_csv("Pollution.csv")
        data.isnull().sum()

Out[8]: state     0
        status    0
        AQI-US    0
        PM2.5     0
        PM10      0
        Temp      0
        Humid     0
        dtype: int64
```

Fig3: Checking for null values in the dataset

Analysing the data:

```
In [16]: sns.pairplot(data)

Out[16]: <seaborn.axisgrid.PairGrid at 0x7f0cb4791180>
```

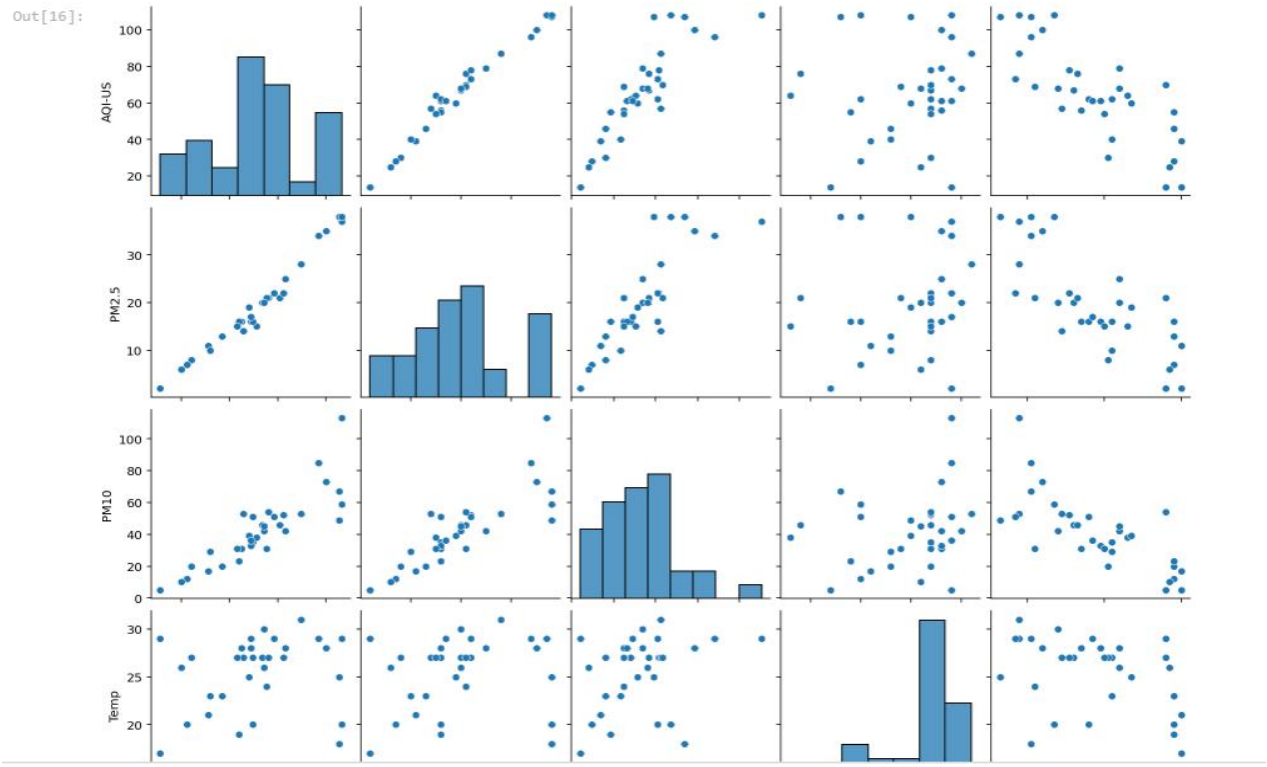


Fig4:pairplot of data present in the data

```
In [17]: sns.displot(x=data["Temp"])
Out[17]: <seaborn.axisgrid.FacetGrid at 0x7f0cb14e8460>
```

Out[17]:

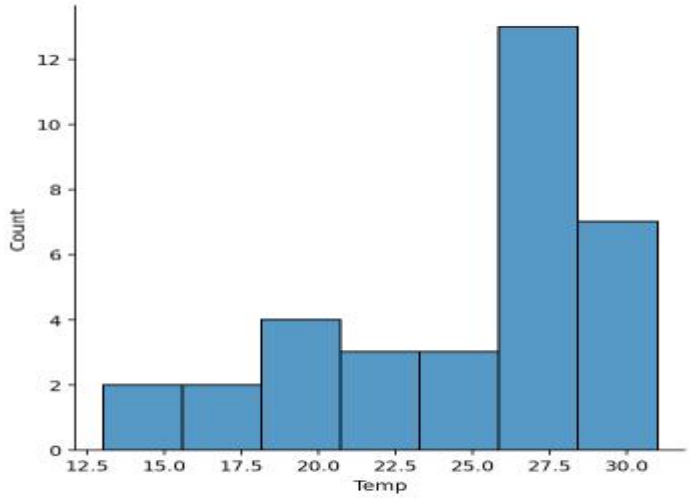


Fig5 :Displot of state wise Temparature data

```
In [20]: import seaborn as sns
         import pandas as pd

         # Select only numeric columns
         numeric_data = data.select_dtypes(include=[np.number])

         # Calculate the correlation and plot the heatmap
         sns.heatmap(numeric_data.corr(), annot=True)
```

Out[20]: <Axes: >

Out[20]:

Fig6:Heat map

```
In [22]: plt.figure(figsize=(10,5))
         sns.barplot(x="Temp",y="state",data=data)

Out[22]: <Axes: xlabel='Temp', ylabel='state'>
```

Out[22]:

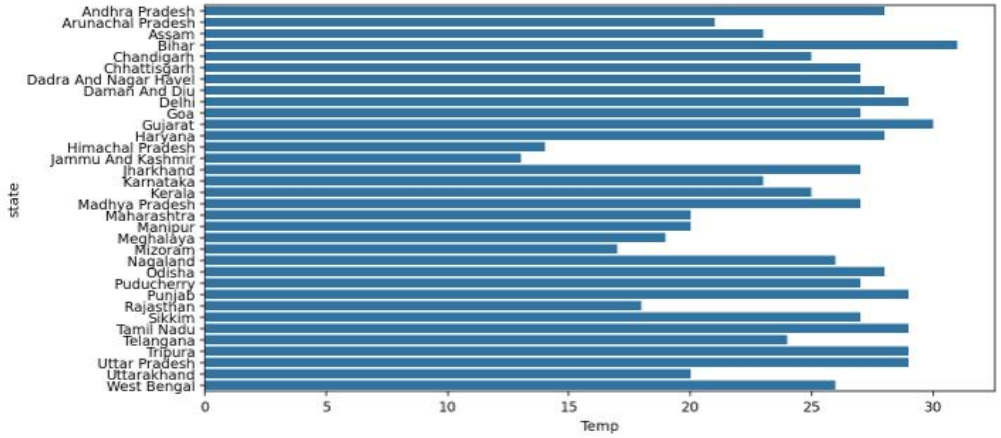


Fig7 : Barplot

Conclusion

Based on the above analysis and visualizations, it would be really helpful for any company which is looking to open up an EV start up in India. In this report, 4 wheeler EV's are more concentrated, the customer space has been visualized in a detailed manner to understand the trends and move accordingly.

Git hub link: [battalaradhika/EV-Market-India.]