Market Segmentation Analysis of ElectricVehicles Market in India

Date: 22nd sep 2023

~ chandan VK

Fynn Labs: Project 2

Link: https://github.com/ChandanVK6/MARKET-SEGMENT-ANALYSIS-MCDONALDS

Problem Statement:

Task is to analyse the Electric Vehicles Market in India using *Segmentation* analysis and come up witha feasibstrategy to enter the market, targeting the segments most likely to use their product in terms of Geographic, Demographic, Psychographic, and Behavioural. In this report we analyse the Electric Vehicles Market in India using segments such a region, price, charging facility, type of vehicles (e.g., 2 wheelers, 3 wheelers, 4 wheelers etc.), retail outlets, manufacturers, body type (e.g., Hatchback, Sedan, SUV, Autorickshaw etc.),safety, plug types and much more.

Fermi Estimation

Wild Guess: Around 8-10% people will have electric vehicles by the end of 2023 in India.Educated Guess:

Employment rate = it is the ratio of number of available labor force to the population of People inthe working age.

We think there are about 1.5 billion Indians in the world. Let's assume the only people over18and under 60 works, assuming that they account for around 60% of the population then that would make 0.9 billion Indians in the working class. Out of the 0.9 billion people not all are
employed, assuming only 2023 had 45% employment rate that would bring the numberaround 405 million.
Since, not everyone can afford an electric vehicle, let's assume only people above middle
class can afford an electric vehicle, that would be 40 million. Not everyone buys an electricvehicle. Let's assume out of these 40 million only 10 million are willing to buy an electric vehicle.

Variables and Formulas:

Let E(x) be the employment rate of the year x (in %). Let P(x) be the population of the year x.Let A(x) be

the number of available Labor in the year x.

Let r be the ratio of Indians between the age of 18 and 60 to the total population of India.E(x) =

(A(x)*100)/(P(x)*r)

This formula will formulate the Employment ratio for the year x.

Gathering More Information:

Estimation for the population of the year 2022 can be obtained by the increase in populationeach year
P (2019) = 1.3676 billion

P (2020) = 1.3786 billion
P (2021) = 1.39199 billion

P (2020)-P (2019) =
11million P (2021)-P (2020) = 13.39
million themean would be
12.195 million

thus P (2022) = 1.44185 billion assuming A(x) is constantevery year=
471,688,990r=0.6 C=0.75
E (2022) = (471,688,990/(1,441,850,000*0.6))*0.75 E (2022) = 42%

Conclusion: By this analysis, we conclude that by the end of the year 2024 there would a Employment rate of 42%. That would make 42% of 405 million i.e., 170 million. Out of these
170 million only 10% afford EV'S. So around 17 million people will have EV's by the end of 2024"

Columns explanations:

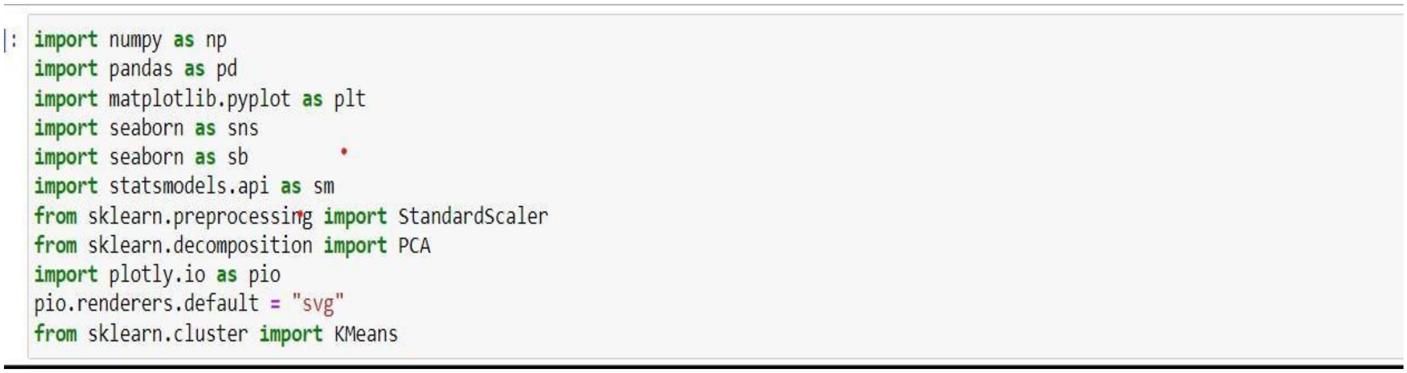
1. ‘Brand’ and tells the manufacturers of electric vehicles.
2. ‘model’ tells the various of electric vehicles.
3. ‘Accuse’, ‘Top Speed’, ’Power Train’ tells specification about the vehicles.
4. ‘Range’, ‘Fast Charge’, ‘Plug_type’ and ‘ Bodystyle’ tells us about range of vehicle per fullcharge,fast charging is provided or not, type of charging plug and body style of vehicle respectively.
5. ‘Seats’ and ‘Price’ tells about the number of seats available on vehicle and their price.
6. ‘Region’ and ‘State/UT’ tells about the states of India.
7. ‘EV Charging Facility’ and ‘Chargers’ tells about the facility of charging in the respectivestates.

Data pre-processing

Required libraries

In order to perform EDA and clustering on the collected data, the following Python libraries are used:

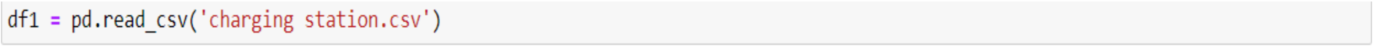
1. Pandas: for data handling/manipulation
2. Matplotlib and Seaborn: for data visualization
3. Scikit-learn: for the k-means clustering algorithm and some other algorithms



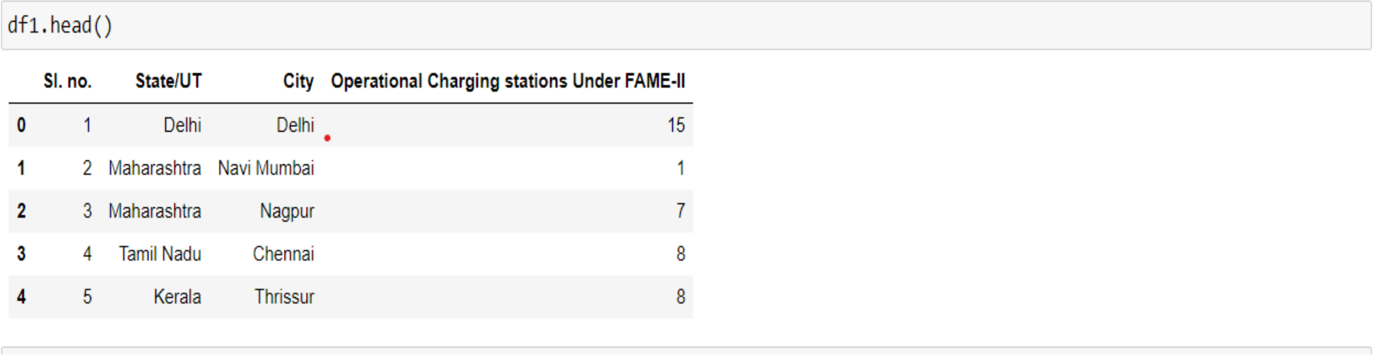
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import seaborn as sb
import statsmodels.api as sm
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
import plotly.io as pio
pio.renderers.default = "svg"
from sklearn.cluster import KMeans
```

Pulling the datasets

Dataset 1



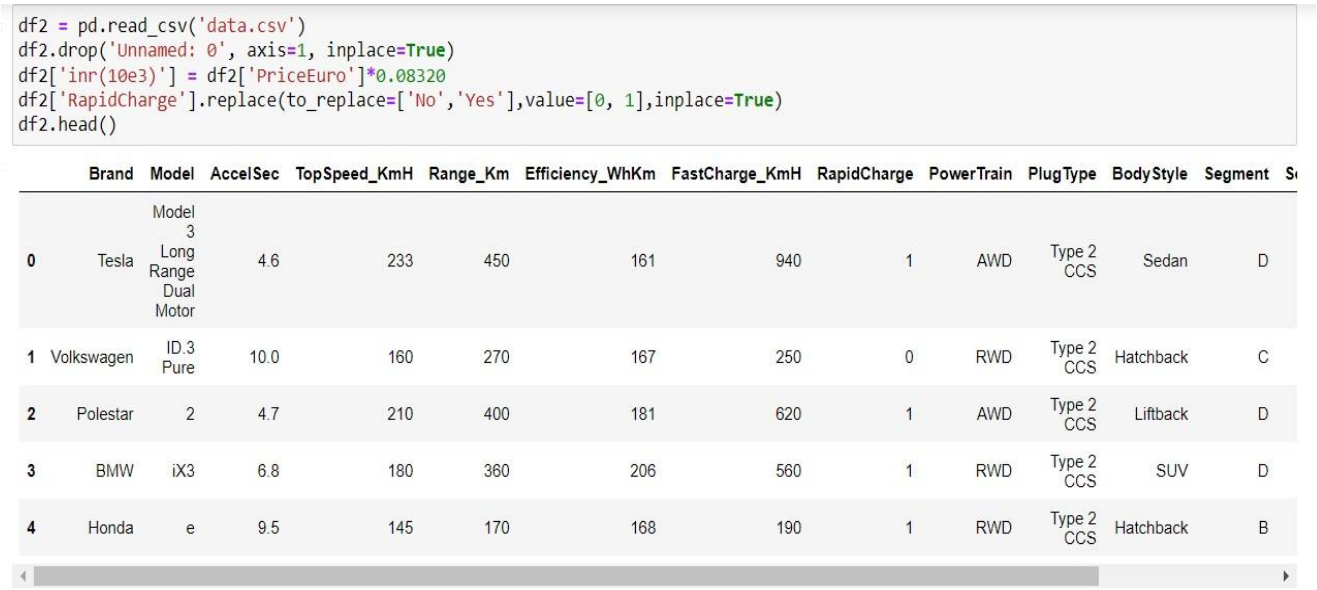
```
df1 = pd.read_csv('charging station.csv')
```



```
df1.head()
```

	Sl. no.	State/UT	City	Operational Charging stations Under FAME-II
0	1	Delhi	Delhi	15
1	2	Maharashtra	Navi Mumbai	1
2	3	Maharashtra	Nagpur	7
3	4	Tamil Nadu	Chennai	8
4	5	Kerala	Thrissur	8

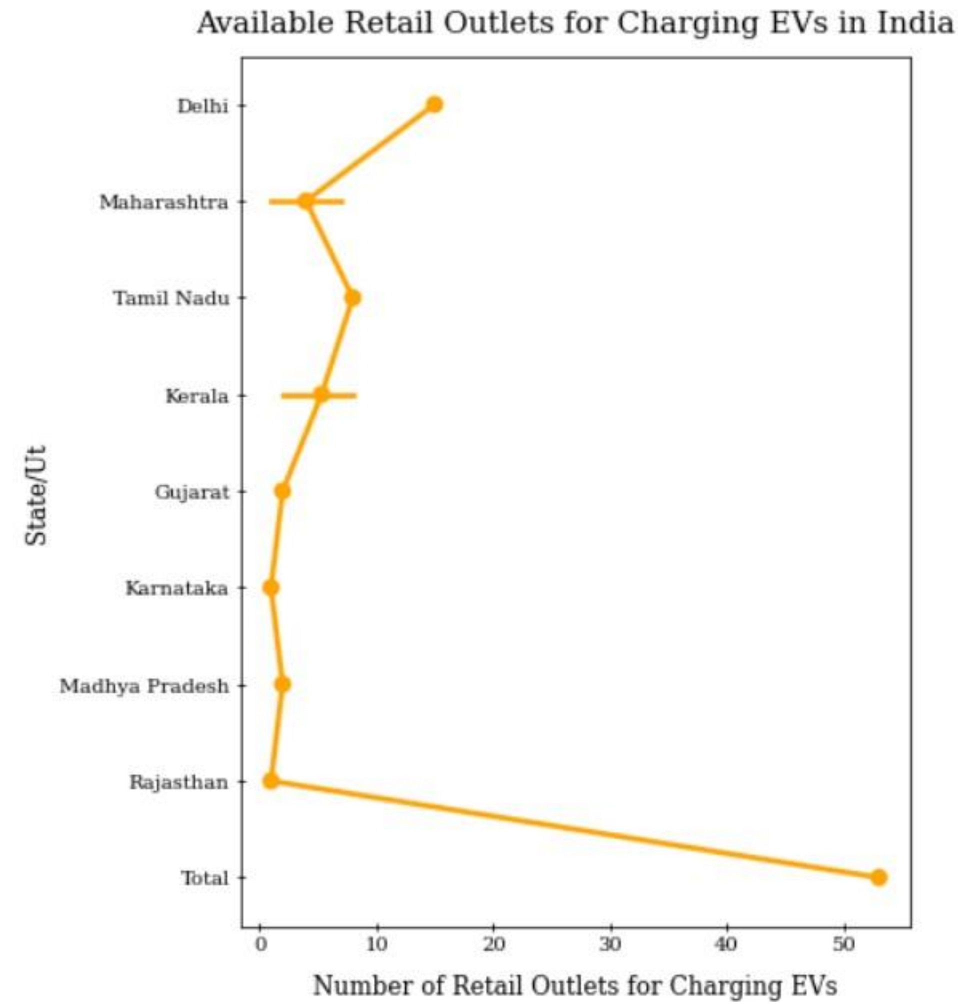
Dataset 2



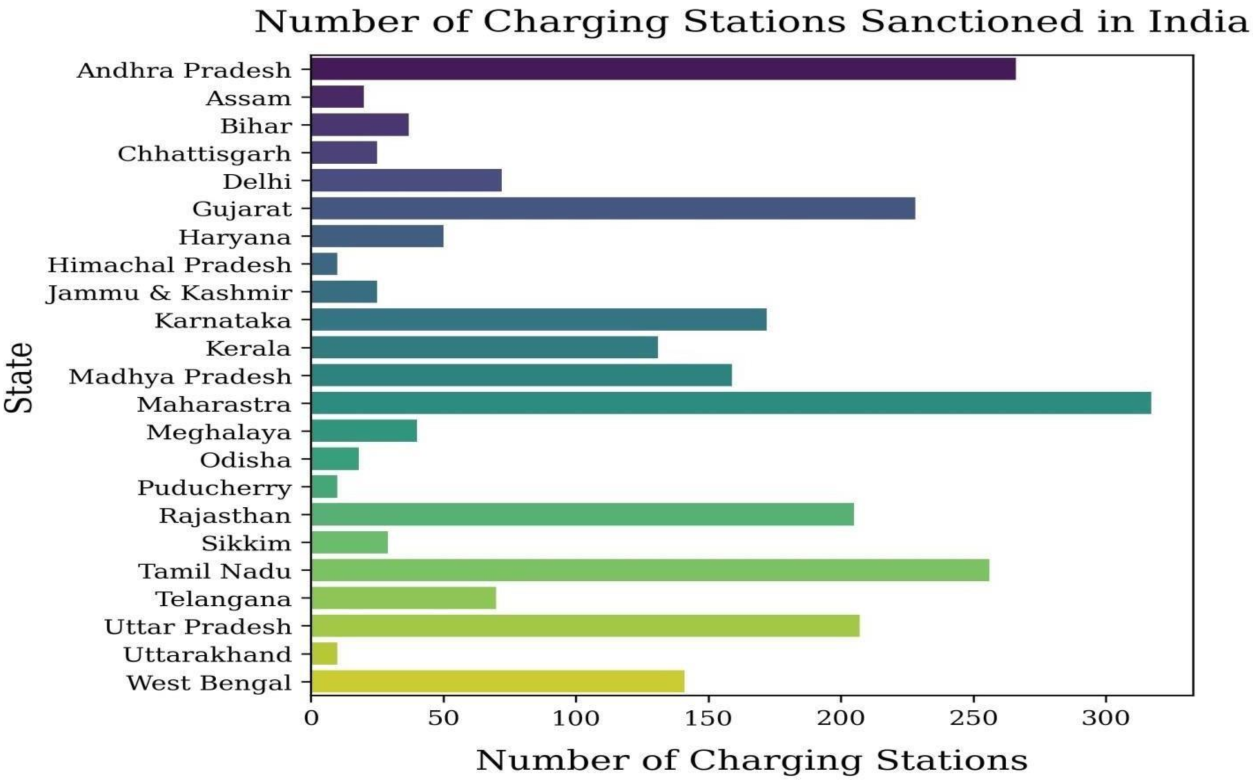
```
df2 = pd.read_csv('data.csv')
df2.drop('Unnamed: 0', axis=1, inplace=True)
df2['inr(10e3)'] = df2['PriceEuro']*0.08320
df2['RapidCharge'].replace(to_replace=['No','Yes'],value=[0, 1],inplace=True)
df2.head()
```

	Brand	Model	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	PowerTrain	PlugType	BodyStyle	Segment	S
0	Tesla	Model 3 Long Range Dual Motor	4.6	233	450	161	940	1	AWD	Type 2 CCS	Sedan	D	
1	Volkswagen	ID.3 Pure	10.0	160	270	167	250	0	RWD	Type 2 CCS	Hatchback	C	
2	Polestar	2	4.7	210	400	181	620	1	AWD	Type 2 CCS	Liftback	D	
3	BMW	iX3	6.8	180	360	206	560	1	RWD	Type 2 CCS	SUV	D	
4	Honda	e	9.5	145	170	168	190	1	RWD	Type 2 CCS	Hatchback	B	

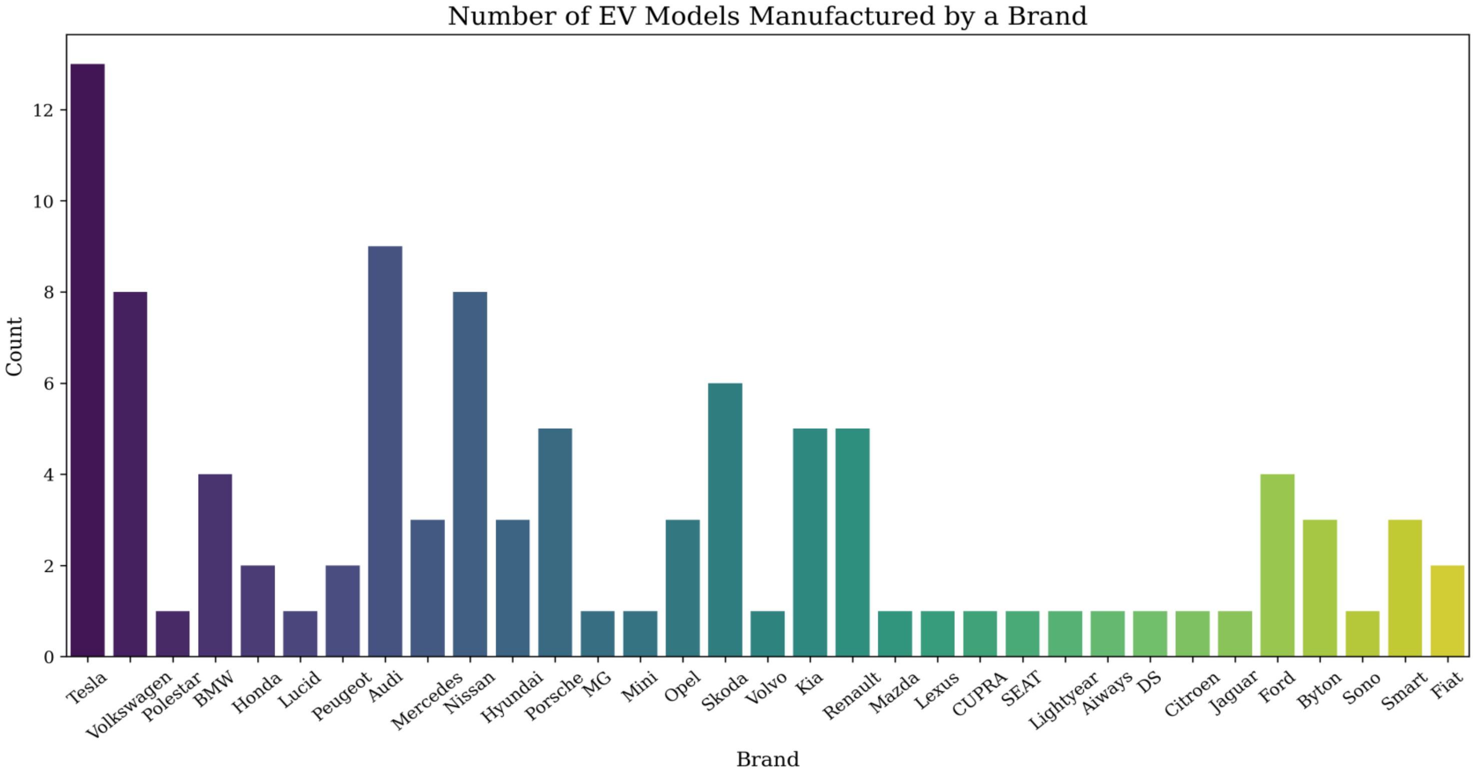
Retail outlets in India for charging EVs



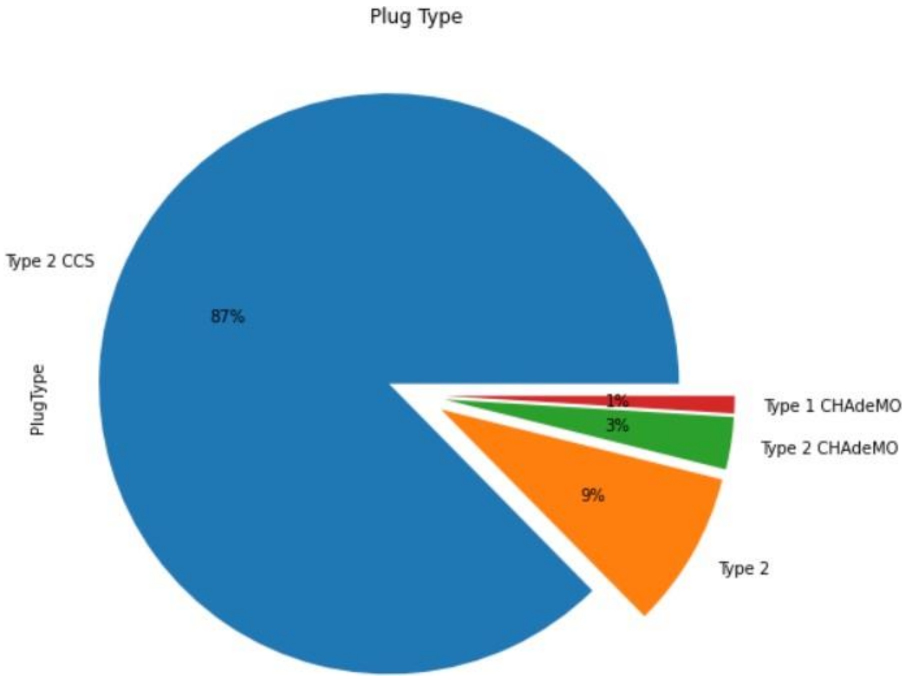
Number of charging stations sanctioned by Government of India



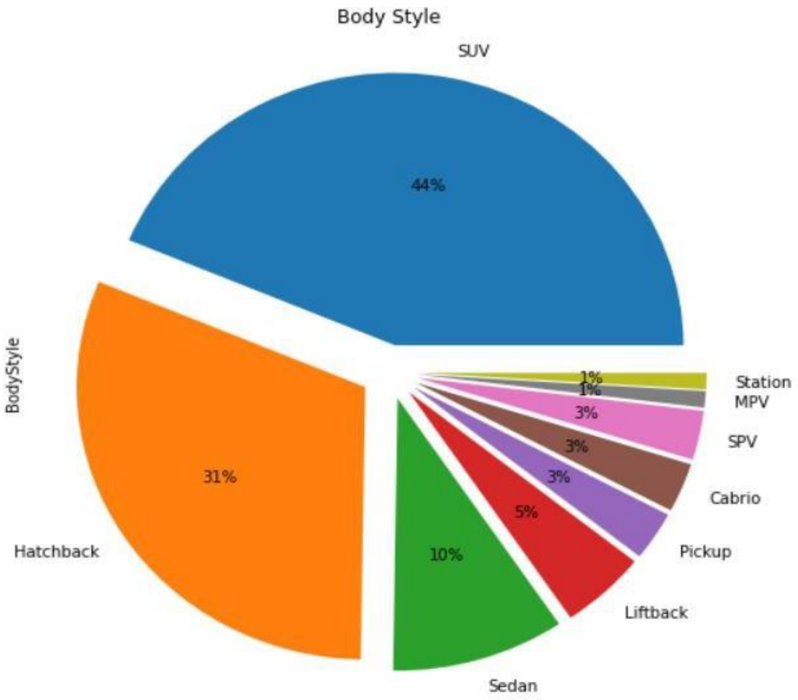
Top EV manufacturing brands in India



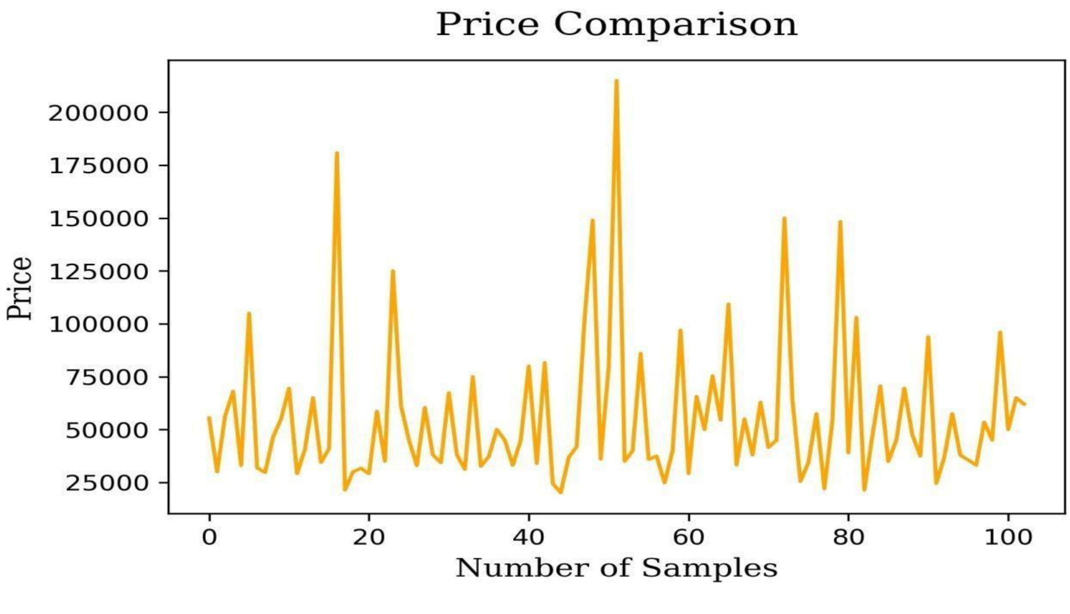
Types of EV plugs available in India



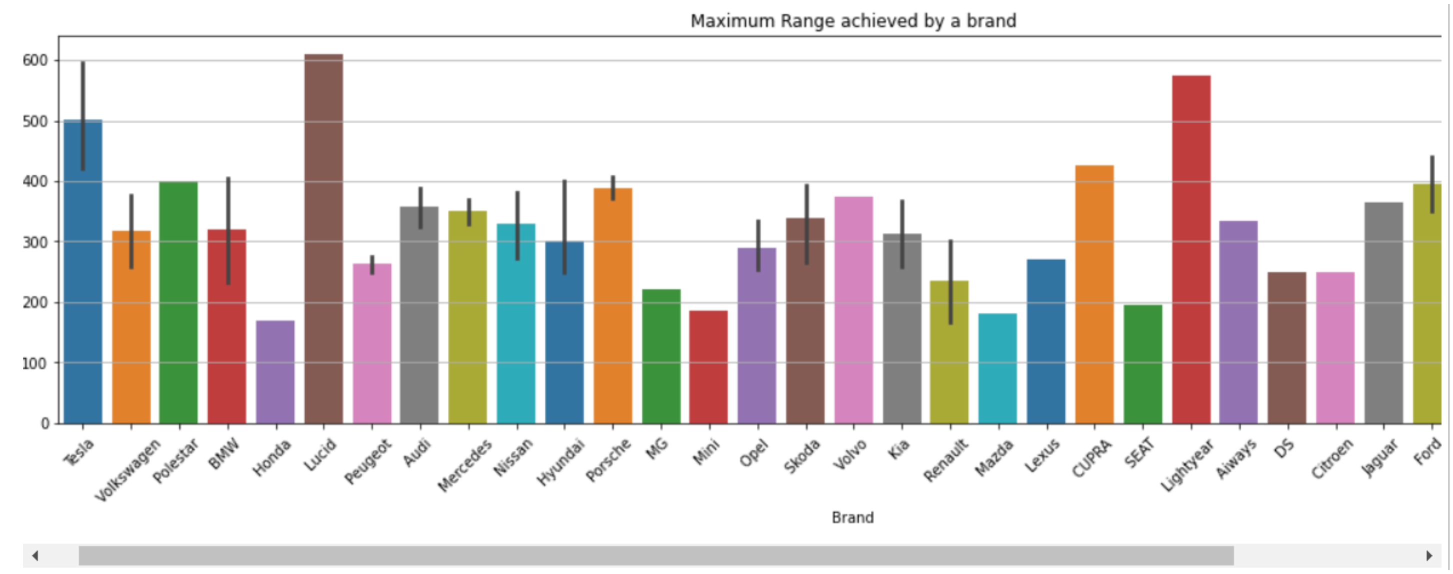
Body types of EVs in India



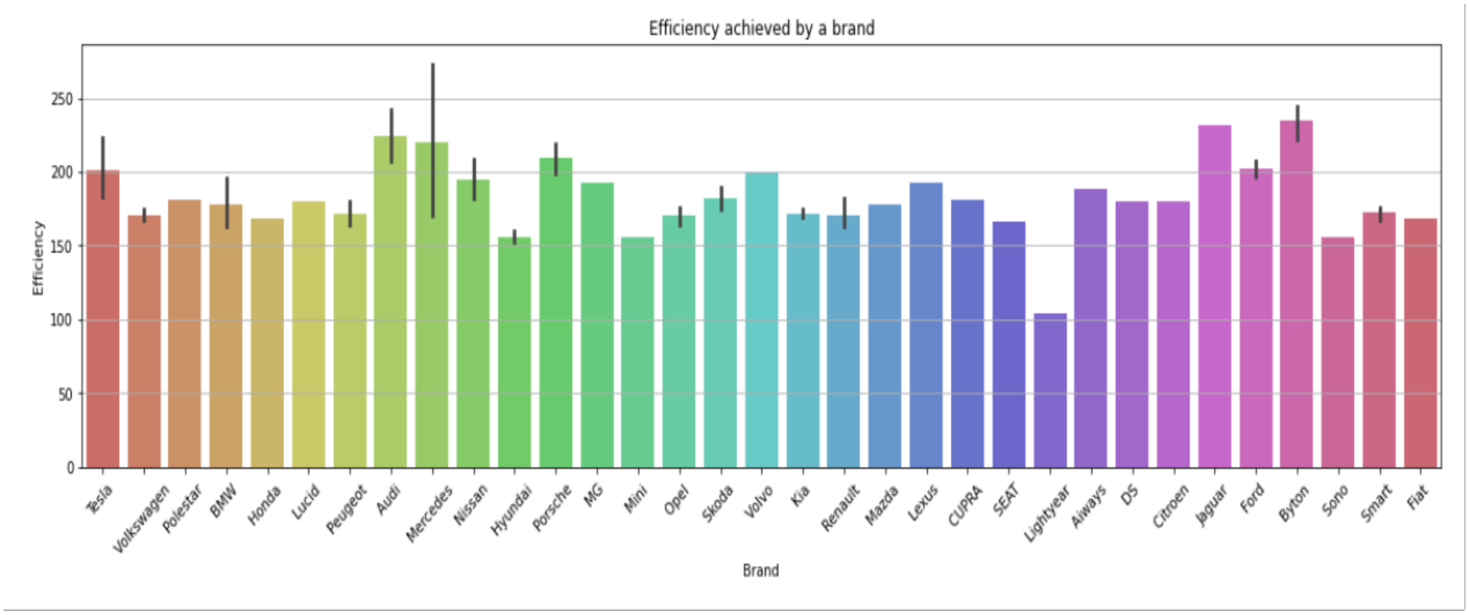
Price comparison of different brands of EVs in India



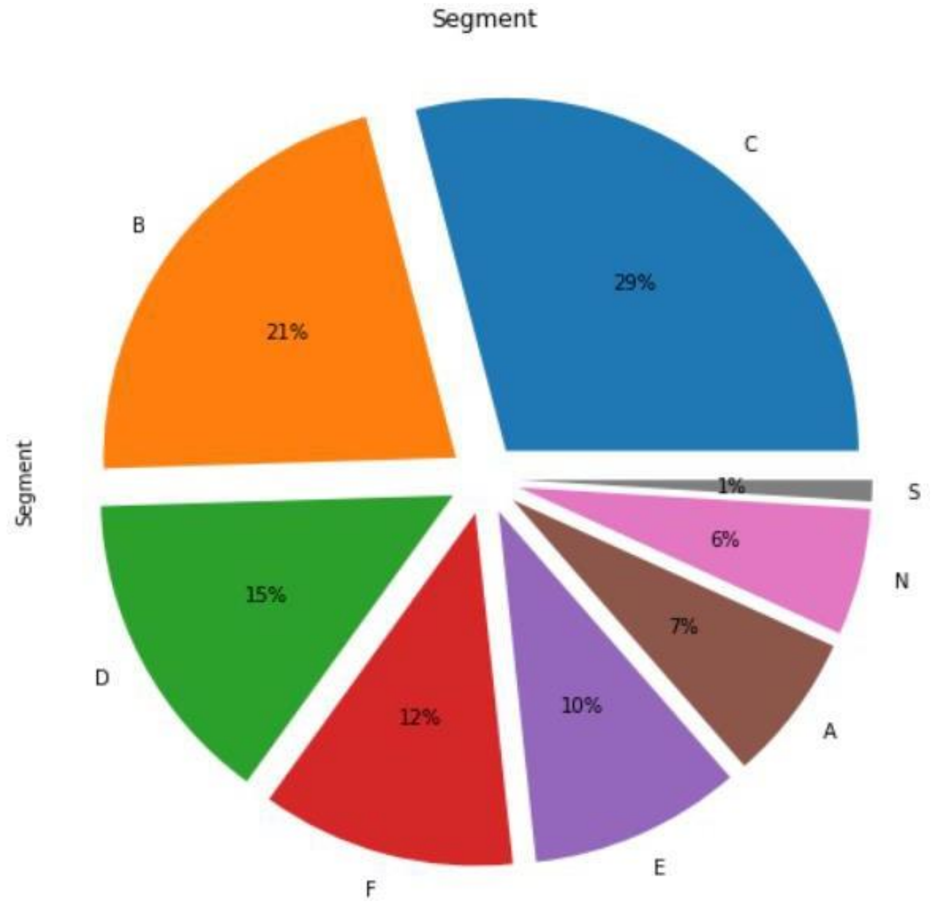
Efficiency achieved by a brand



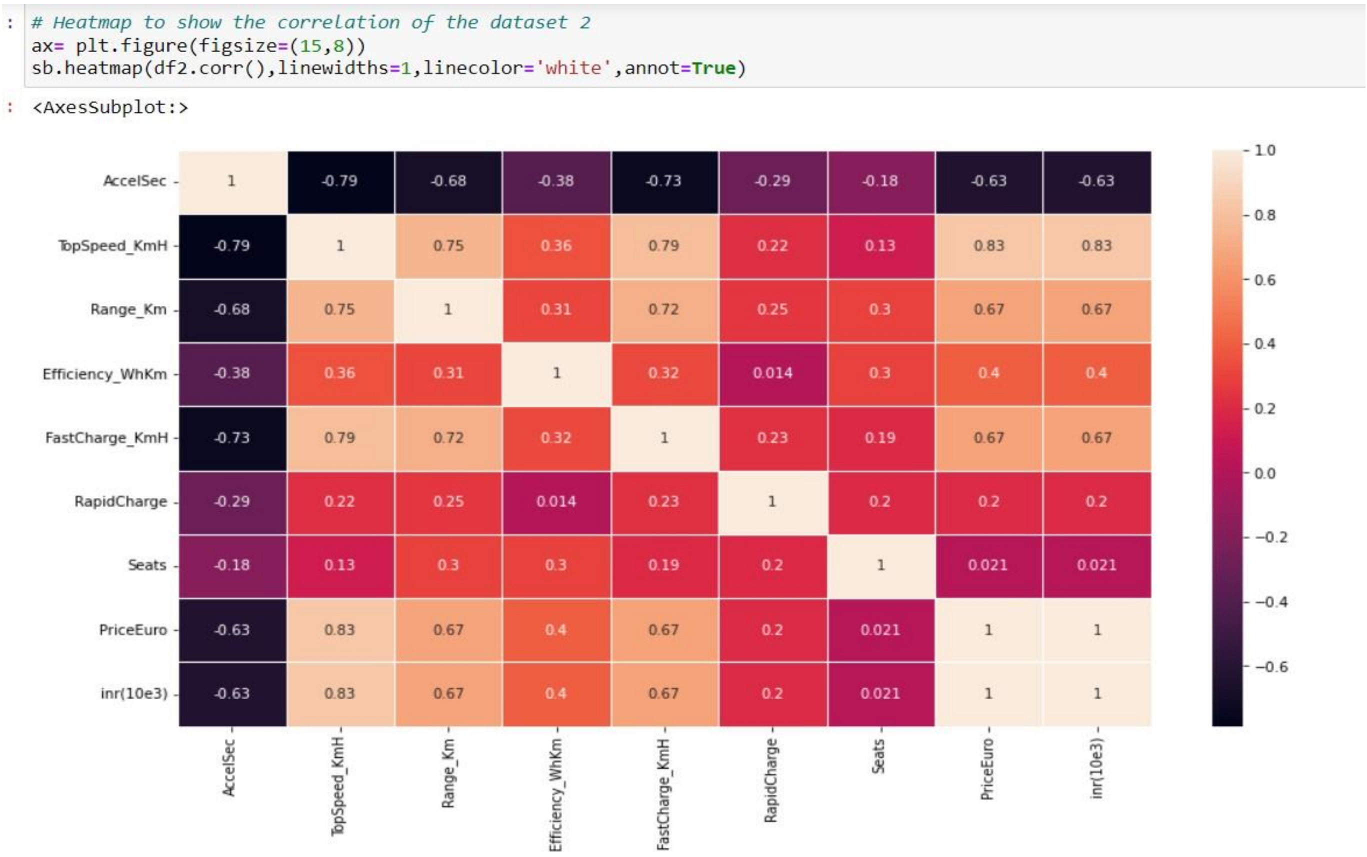
Maximum Range achieved by a brand



EV Segments in India



Correlation Matrix



```
# Heatmap to show the correlation of the dataset 2
ax= plt.figure(figsize=(15,8))
sb.heatmap(df2.corr(),linewidths=1,linecolor='white',annot=True)
```

```
<AxesSubplot:>
```

	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	Seats	PriceEuro	inr(10e3)
AccelSec	1	-0.79	-0.68	-0.38	-0.73	-0.29	-0.18	-0.63	-0.63
TopSpeed_KmH	-0.79	1	0.75	0.36	0.79	0.22	0.13	0.83	0.83
Range_Km	-0.68	0.75	1	0.31	0.72	0.25	0.3	0.67	0.67
Efficiency_WhKm	-0.38	0.36	0.31	1	0.32	0.014	0.3	0.4	0.4
FastCharge_KmH	-0.73	0.79	0.72	0.32	1	0.23	0.19	0.67	0.67
RapidCharge	-0.29	0.22	0.25	0.014	0.23	1	0.2	0.2	0.2
Seats	-0.18	0.13	0.3	0.3	0.19	0.2	1	0.021	0.021
PriceEuro	-0.63	0.83	0.67	0.4	0.67	0.2	0.021	1	1
inr(10e3)	-0.63	0.83	0.67	0.4	0.67	0.2	0.021	1	1

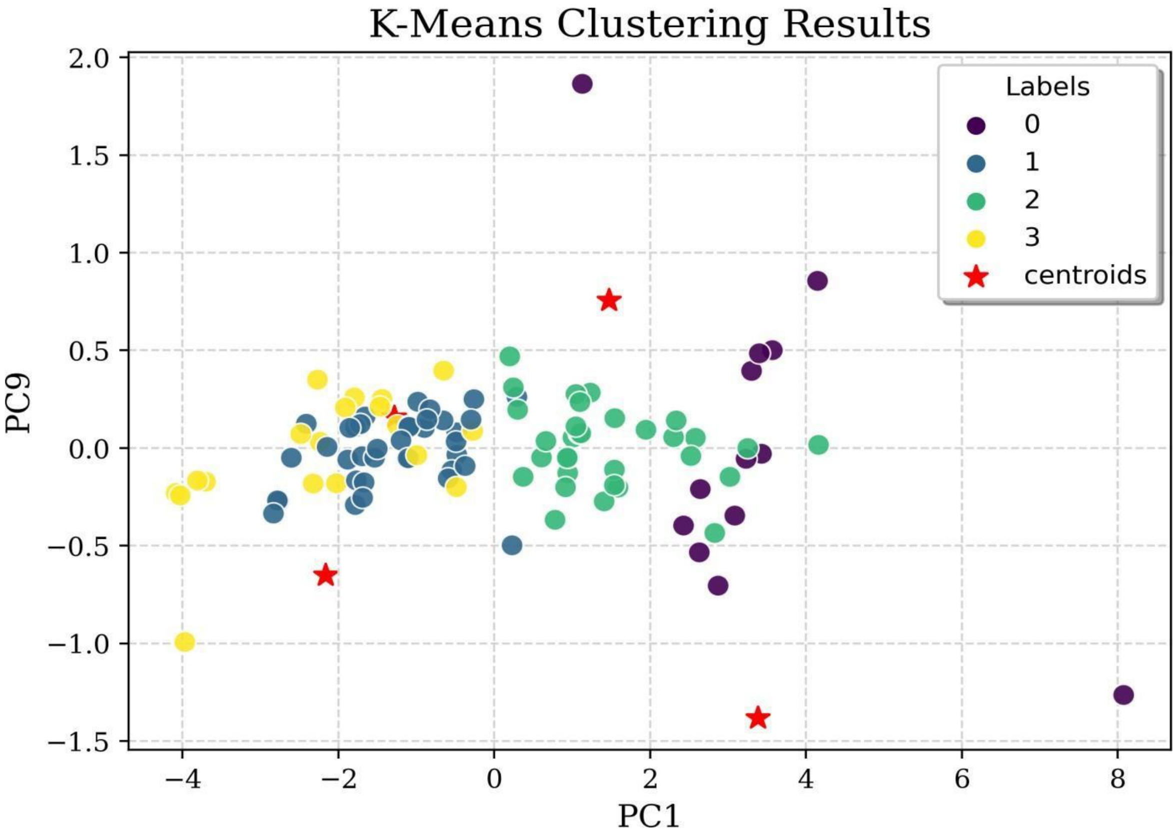
Segmentation Approaches

Clustering

Clustering is an unsupervised machine learning technique of grouping similar data points into clusters. The sole objective of this technique is to segregate datapoints with similar traits and place them into different clusters. There are several algorithms to perform clustering on data such as k-means clustering, hierarchical clustering, density-based clustering etc.

K-Means Clustering

K-Means Clustering is an unsupervised learning algorithm whose job is to group the unlabelled dataset into different clusters where each datapoint belongs to only one cluster. Here, K is the number of clusters that need to be created in the process. The algorithm finds its applicability into a variety of use cases including market segmentation, image segmentation, image compression, document clustering etc. The below image is the results of clustering on one of our datasets.



The K-Means Algorithm works the following way:

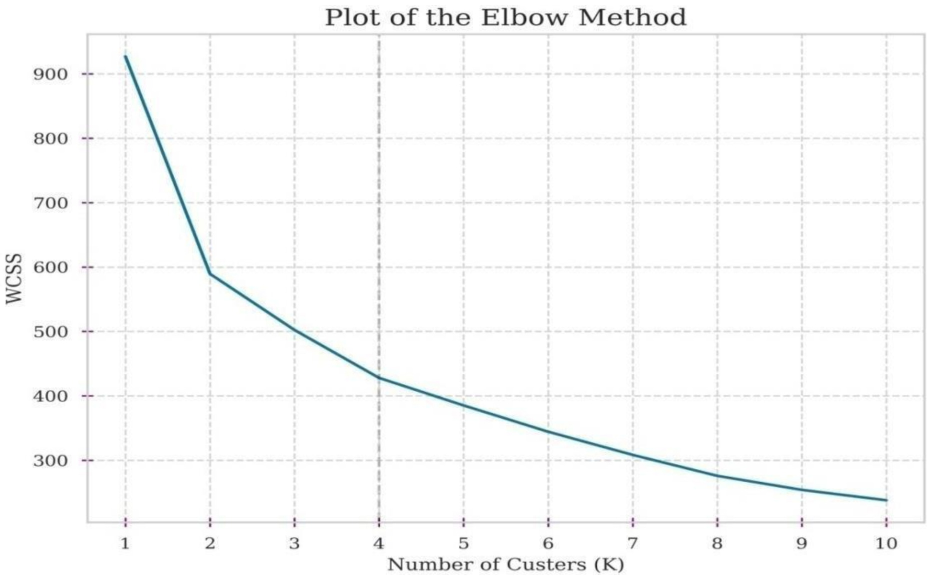
1. Specify the number of clusters, i.e. K
2. Select K random points in the dataset. These points will be the centroids (centres) of each of the K clusters.
3. Assign each data point in the dataset to one of the K centroids, based on its distance from each of the centroids.
4. Consider this clustering to be correct and reassign the Centroids to the mean of these clusters.
5. Repeat Step 3. If any of the points change clusters, Go to step 4. Else Go to step 6.
6. Calculate the variance of each of the clusters.
7. Repeat this clustering ‘n’ number of times until the sum of variance of each cluster is minimum.

Principle Component Analysis

Principal component analysis (PCA) is a linear dimensionality-reduction technique that is used to reduce the dimensionality of large data sets by transforming a large set of variables into a smaller one while preserving most of the information present in the large set.

Elbow Method

The Elbow method is a way of determining the optimal number of clusters (k) in K-Means Clustering. It is based on calculating the Within Cluster Sum of Squared Errors (WCSS) for a different number of clusters (k) and selecting the k for which change in WCSS first starts to diminish. When you plot its graph, at one point the line starts to run parallel to the X-axis and that point, known as the Elbow Point, is considered as the best value for the k (as 4 in the below figure).


Plot of the Elbow Method
WCSS
Number of Custers (K)