Comprehensive Analysis Report: EV Market in India

Market Segmentation

Name Amrendra Kumar

Date – 8/1/2025

Electric Vehicle Market analysis in India

"The electric vehicle is not a single invention; it's a revolution in how we think about transportation and sustainability."
– Elon Musk

Abstract

Market segmentation is a vital strategic tool for advancing transportation technologies, particularly electric vehicles (EVs), in emerging markets. This approach facilitates the identification and implementation of strategies to drive widespread adoption. EV adoption is anticipated to surge in the near future, driven by their low emissions and reduced operating costs, which have piqued significant academic and industrial interest.

The primary objective of this study is to explore and classify potential consumer segments for EVs, based on psychographic, behavioral, and socio-economic attributes. Utilizing an integrated research framework of "perceived benefits–attitude–intention," the study applies advanced analytical techniques, including cluster analysis, multiple discriminant analysis, and Chi-square tests. Data from a cross-sectional online survey of 563 respondents were analysed to validate the identified segments.

The findings reveal three distinct consumer groups—Conservatives, Indifferents, and Enthusiasts—each representing emerging buyer personas within the EV market. These insights are instrumental for academics and policymakers in crafting targeted strategies to promote EV adoption, aligning with the broader vision of sustainable transportation.

Additionally, this report extends the analysis through Fermi estimation, deconstructing complex adoption scenarios into manageable and quantifiable elements to provide actionable insights.

Keywords: Electric vehicles, Market segmentation, Cluster analysis, Consumer behavior, Attitude towards EVs, Adoption intention, Sustainable transportation, Emerging markets

Introduction

Electric vehicles represent the future of sustainable transportation, providing a solution to rising fuel costs and environmental pollution. This study aims to analyze key factors influencing the Indian EV market, including brand performance, price trends, and consumer behavior. The insights derived from this analysis can guide policymakers, manufacturers, and marketers in strategizing for widespread EV adoption.

Data Collection:

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

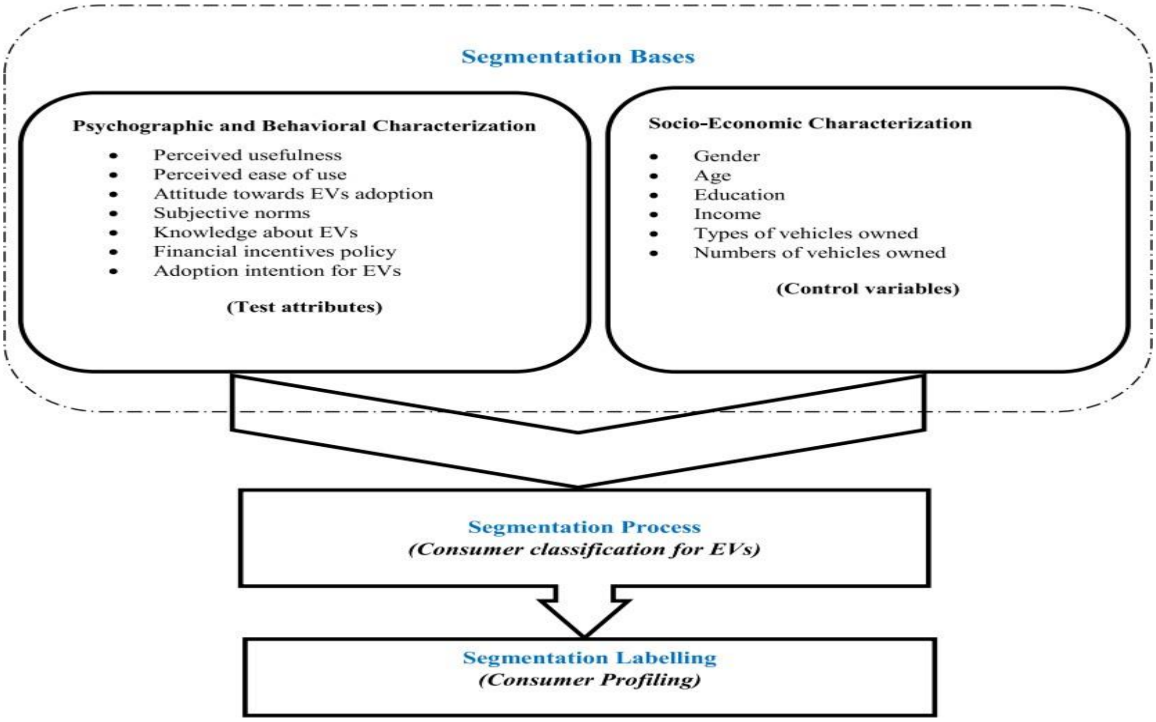
- https://www.kaggle.com/datasets

Segmenting the Electric Vehicle Market

Market segmentation involves categorizing customers into distinct, manageable, and homogeneous subgroups that marketers can target with tailored strategies. There are two primary methods of segmentation: a-priori and post-hoc.

- A-priori segmentation begins by defining customer groups based on predefined criteria such as age, gender, income, or education. After establishing these segments, they are further analyzed using additional variables, such as behavior, lifestyle, or specific benefits sought.
- Post-hoc segmentation, on the other hand, identifies segments by analyzing patterns and relationships across a set of measured variables without predefined criteria.

While the methods differ in approach, both rely on measured variables to determine the segmentation theme. This study employs an a-priori segmentation method to classify potential electric vehicle (EV) customers into meaningful subgroups, enabling targeted marketing strategies.



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 evaluation for eco-friendly products.

They pinpointed the perceived superiority of the psychographic characteristics over the socio-demographic and economic ones in explaining the environmentally-conscious consumer behavior and thus, the study recommended the use of 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.

Implementation

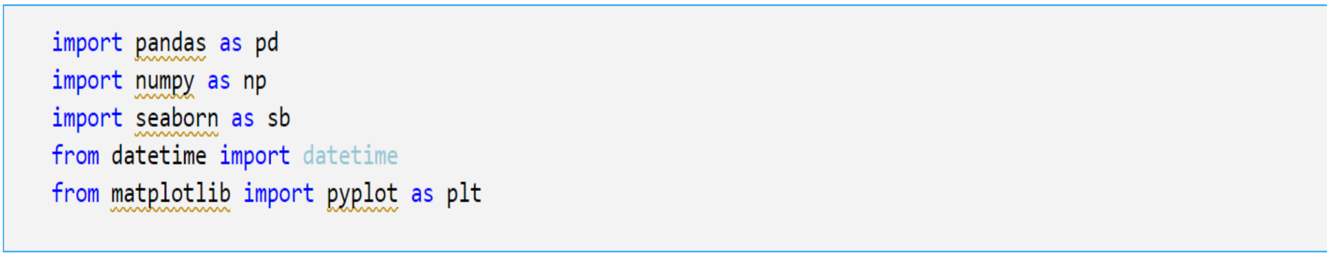
Packages/Tools Used:

1. **Numpy**: Utilized for performing mathematical operations and calculations on arrays and multi-dimensional data.
2. **Pandas**: Used to read, manipulate, and analyze datasets efficiently.
3. **Seaborn**: A visualization library employed for creating aesthetically pleasing statistical plots.
4. **Matplotlib**: Used for creating various types of plots and visualizations to analyze data trends.
5. **Datetime**: Provides functionality for working with dates and times in Python.

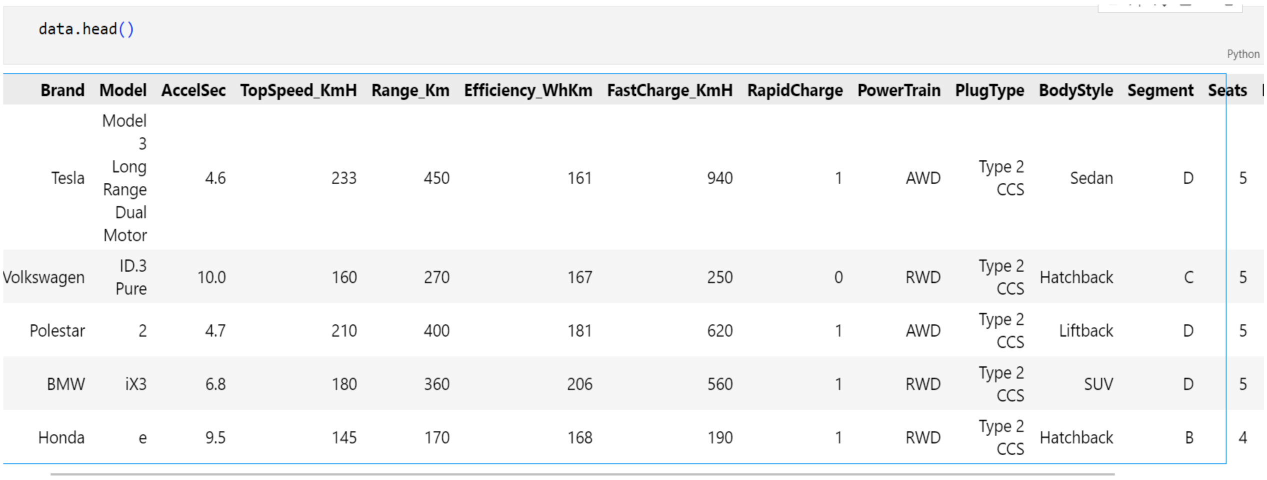
Data-Preprocessing

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.



```
import pandas as pd
import numpy as np
import seaborn as sb
from datetime import datetime
from matplotlib import pyplot as plt
```



```
data.head()
```

Brand	Model	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	PowerTrain	PlugType	BodyStyle	Segment	Seats
Tesla	Model 3 Long Range Dual Motor	4.6	233	450	161	940	1	AWD	Type 2 CCS	Sedan	D	5
Volkswagen	ID.3 Pure	10.0	160	270	167	250	0	RWD	Type 2 CCS	Hatchback	C	5
Polestar	2	4.7	210	400	181	620	1	AWD	Type 2 CCS	Liftback	D	5
BMW	iX3	6.8	180	360	206	560	1	RWD	Type 2 CCS	SUV	D	5
Honda	e	9.5	145	170	168	190	1	RWD	Type 2 CCS	Hatchback	B	4

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) begins with examining and understanding the data to uncover patterns, identify anomalies, and draw initial insights. In this context, we analyze the data both before and after applying Principal Component Analysis (PCA).

What is PCA?

Principal Component Analysis is a statistical technique used to simplify complex datasets by reducing their dimensions while preserving as much variance as possible. It works by transforming a set of correlated features into a new set of uncorrelated features, called Principal Components, through an orthogonal transformation.

- Correlated Features: In many datasets, certain variables tend to have a strong correlation with each other, leading to redundancy. For example, in a dataset with height and weight, these variables might have overlapping information.
- Transformation: PCA identifies the directions (principal components) along which the data varies the most. The first principal component captures the maximum variance, the second captures the next highest variance orthogonal to the first, and so on.
- Dimensionality Reduction: By focusing on the most significant principal components, PCA reduces the number of features while retaining the most critical information. This is particularly useful for making machine learning models more efficient by reducing computational costs and avoiding overfitting.

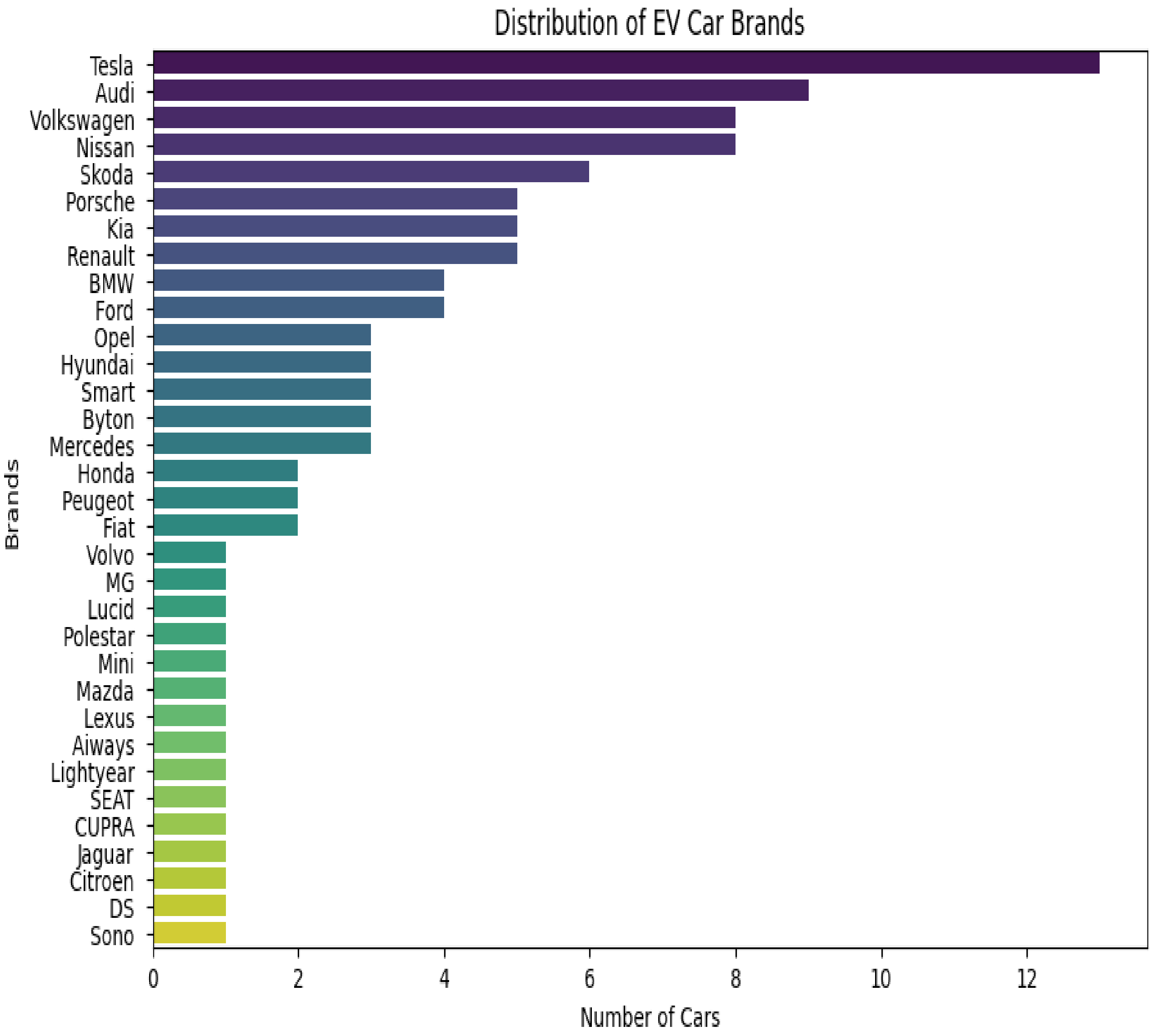
EDA with and without PCA

- Without PCA: In the initial analysis, the dataset is examined as-is, using its full set of features. This helps in understanding the raw data, identifying patterns, and evaluating feature relationships.
- With PCA: After applying PCA, the data is transformed into a reduced set of principal components. This streamlined dataset is then analyzed to compare results with the original data. PCA often reveals underlying trends and structures that may not be apparent in the raw data.

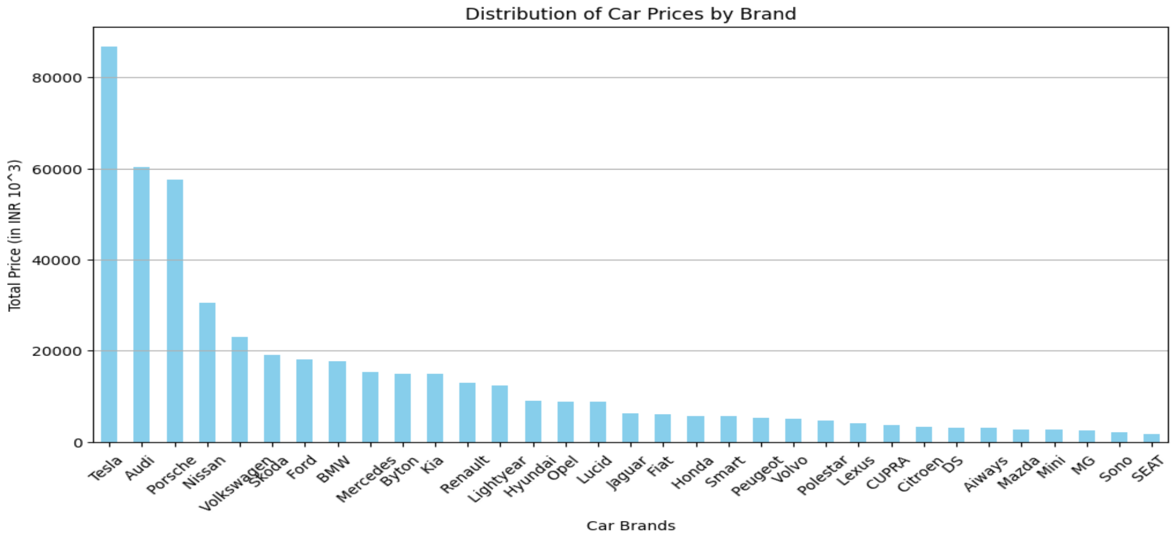
The combined approach of EDA with and without PCA allows us to gain a deeper understanding of the dataset while optimizing its representation for machine learning tasks such as classification, regression, or clustering. This makes the overall process more cost-effective, efficient, and interpretable.

Comparison of cars in our data

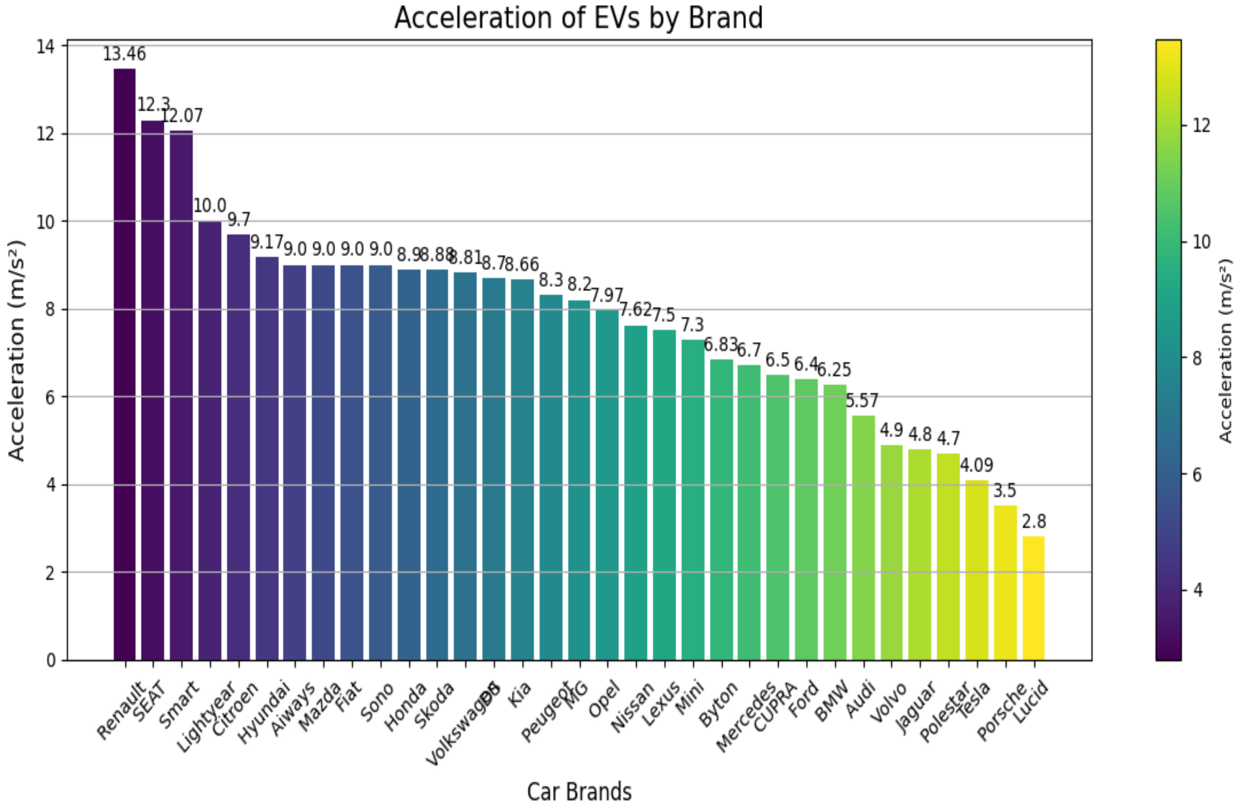
Distribution of EV Car Brands



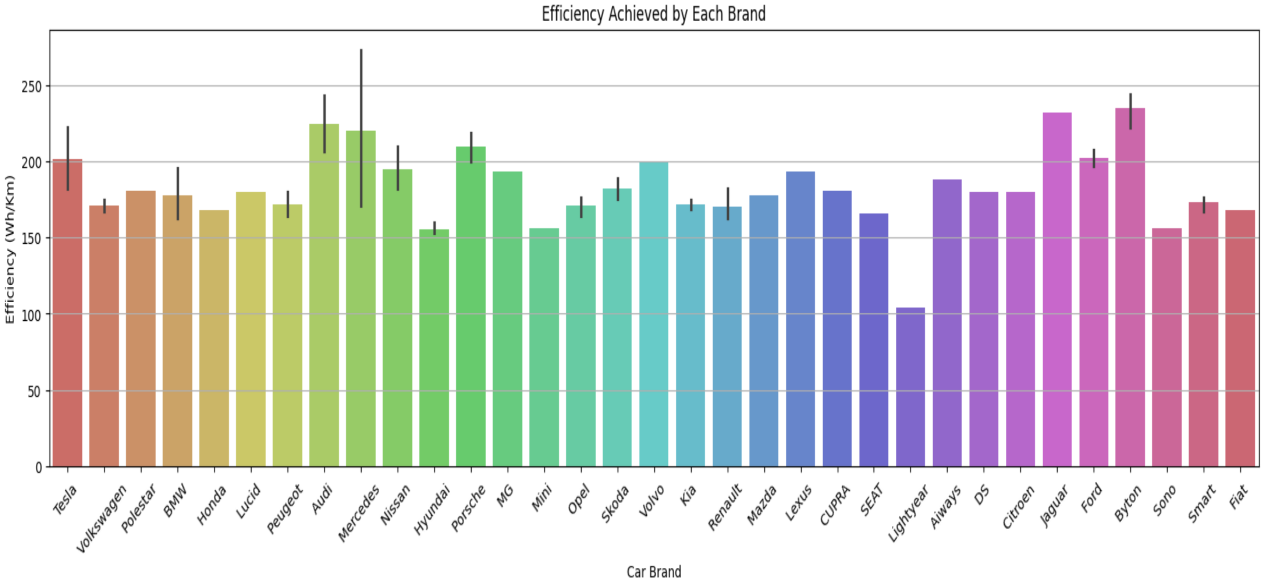
Distribution of Car Prices by Brand



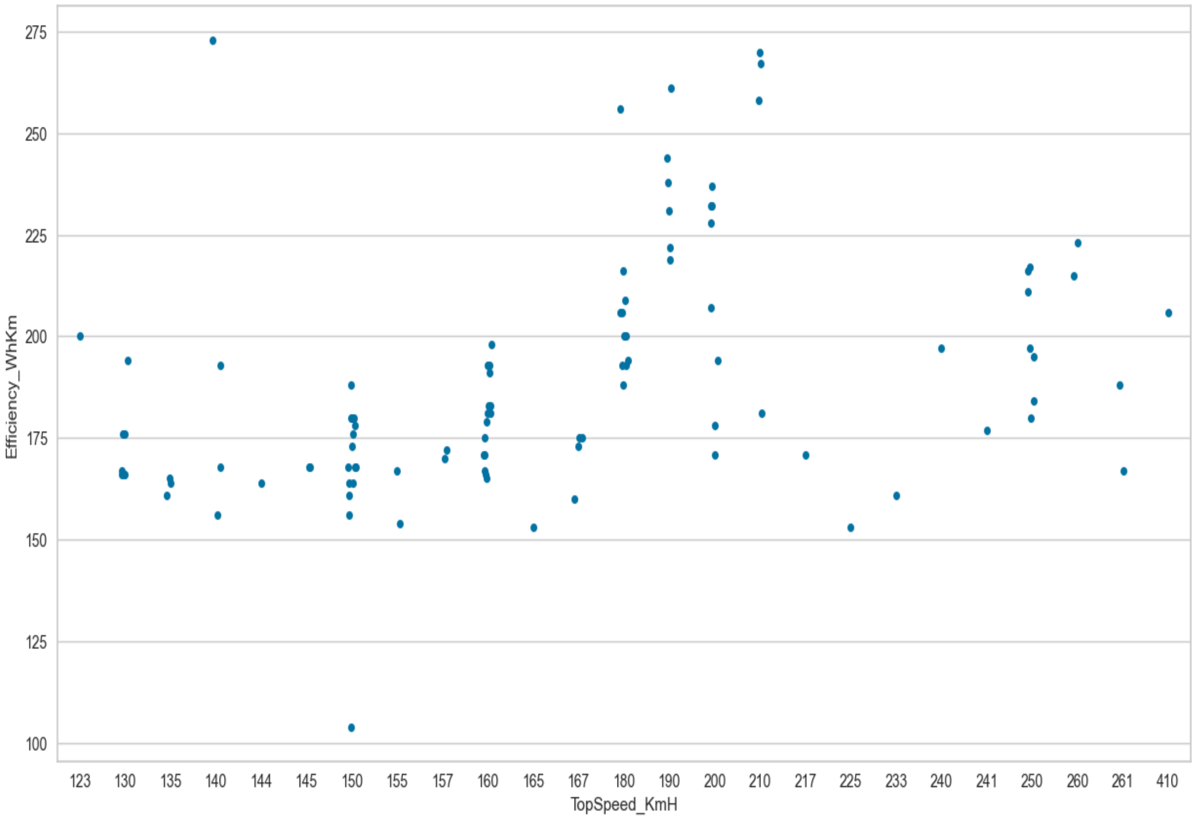
Acceleration of EVs by Brand



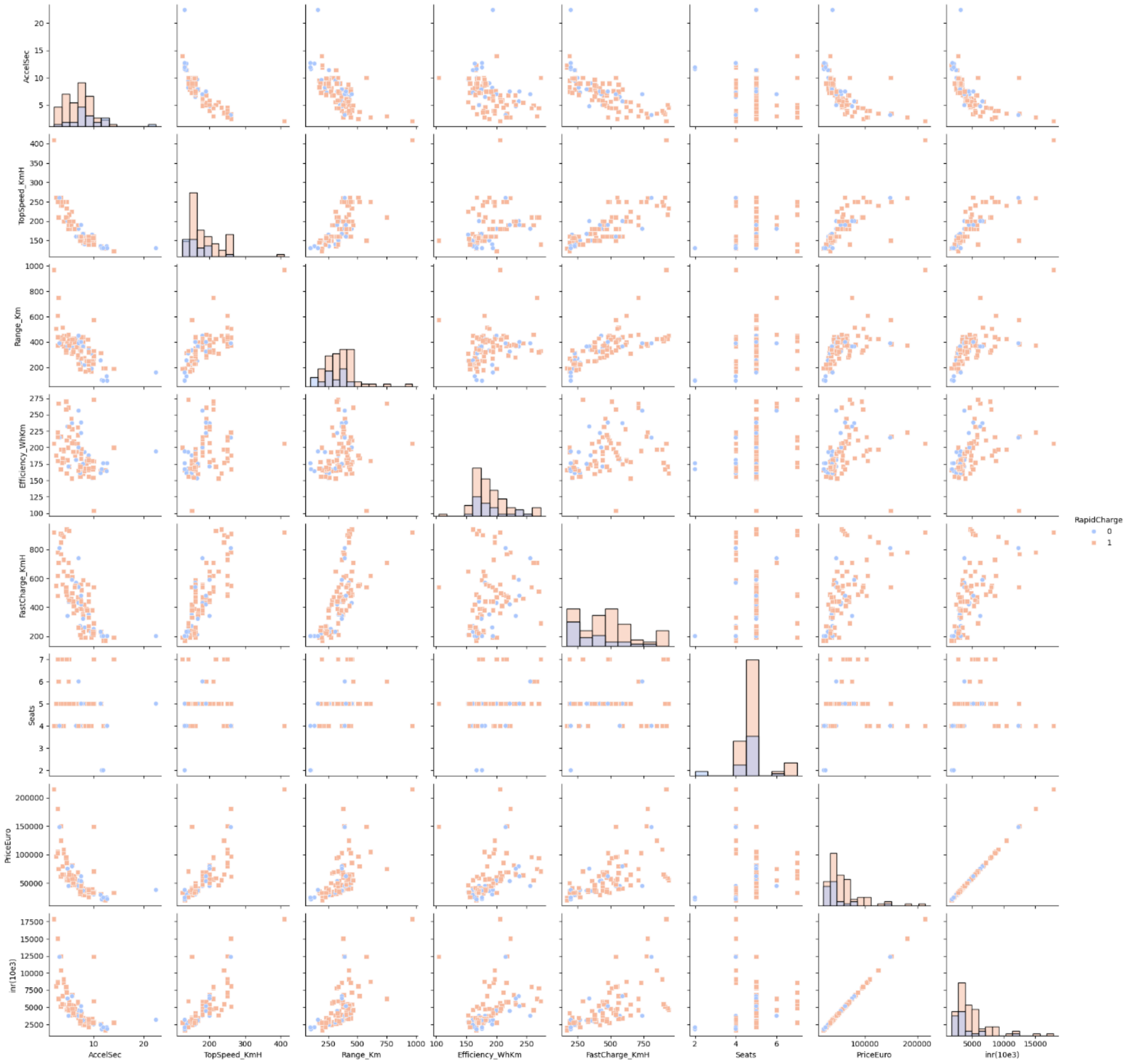
Efficiency Achieved by Each Brand



Efficiency - WhKm Vs TopSpeed – Kmh



Pair Plot based on Rapid charger Presence



Correlation Matrix

A **correlation matrix** is a table that shows the degree of correlation between different variables in a dataset. It is particularly useful for identifying linear relationships between variables. Each cell in the matrix represents the correlation coefficient between a pair of variables, indicating how closely they are related.

Understanding the Correlation Coefficient:

- The **correlation coefficient** ranges from -1 to 1:
  - A value close to **1** indicates a strong positive relationship (as one variable increases, the other also increases).

- A value close to **-1** indicates a strong negative relationship (as one variable increases, the other decreases).
- A value close to **0** indicates little to no linear relationship between the variables.

Key Points:

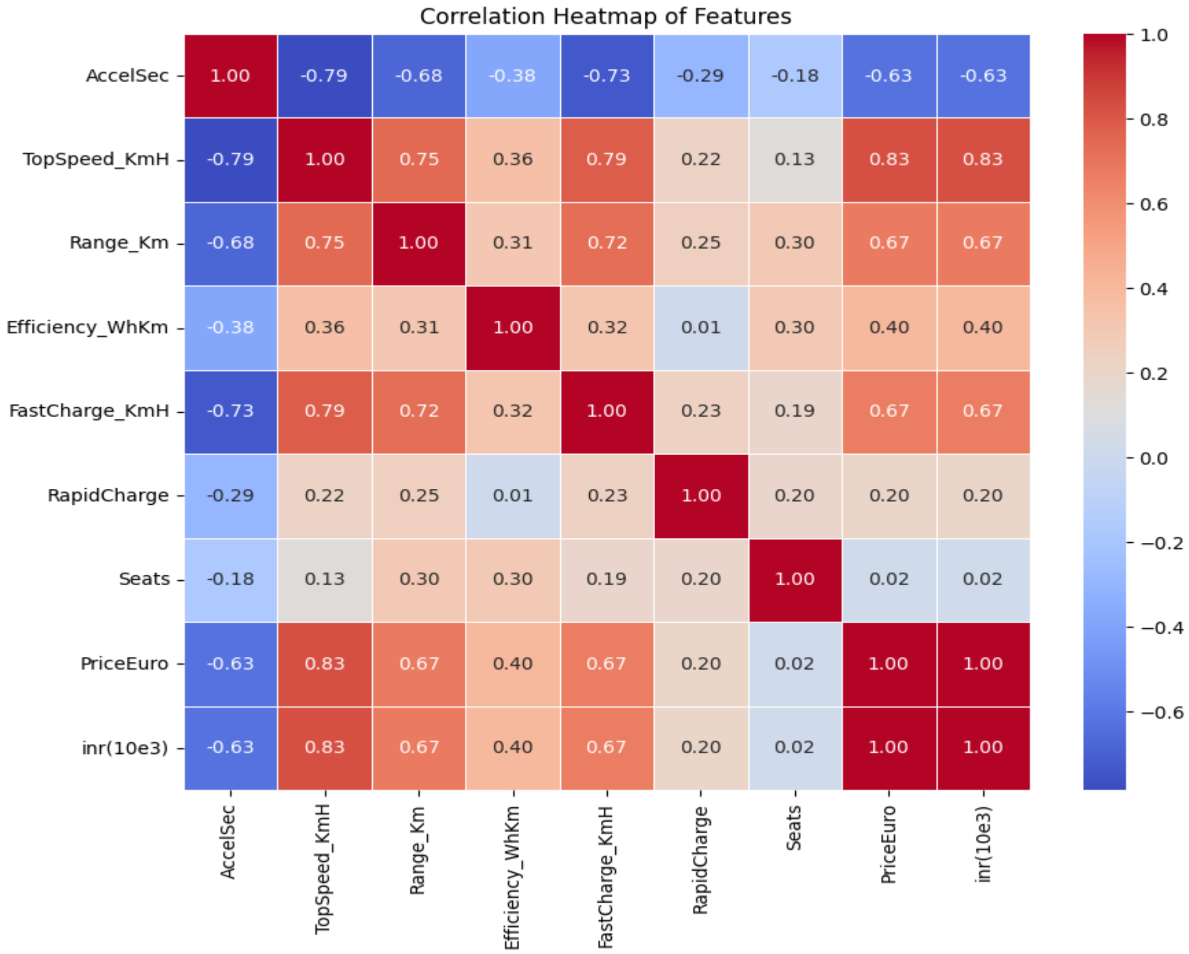
- A **correlation coefficient greater than 0.7** typically suggests a strong positive relationship.
- Conversely, coefficients less than -0.7 indicate a strong negative relationship.

Visualizing the Correlation Matrix:

A **heatmap** is commonly used to visualize the correlation matrix. In the heatmap:

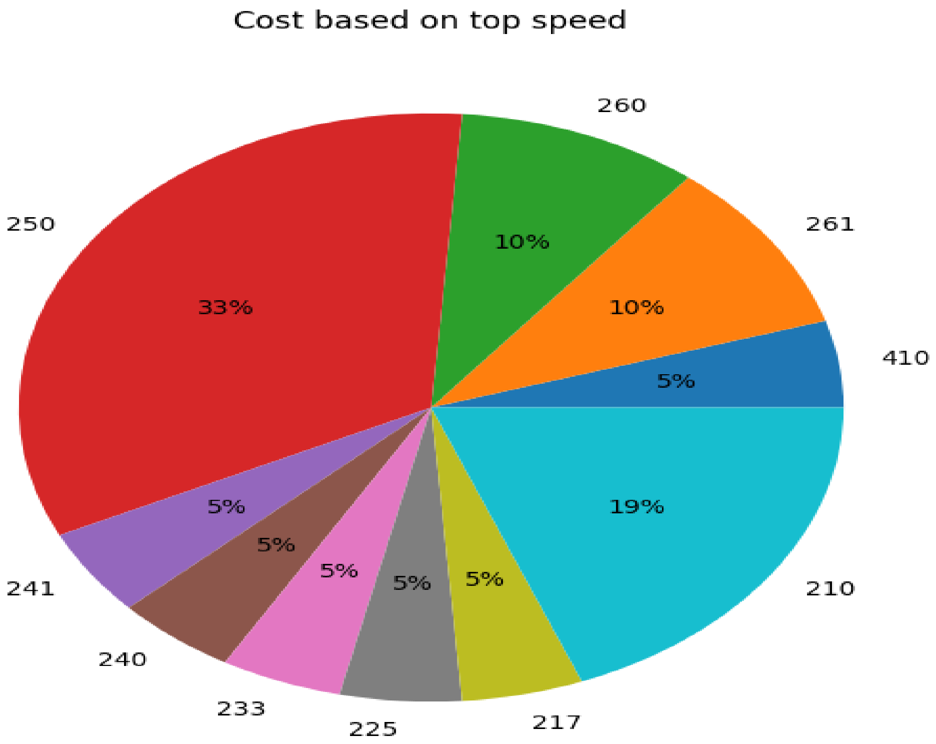
- Each cell is color-coded based on the correlation value, making it easy to identify relationships at a glance.
- Darker or more saturated colors often represent stronger correlations, whether positive or negative.

This visualization helps in quickly spotting variables with strong relationships, which can be critical for feature selection in machine learning models or for understanding data patterns.

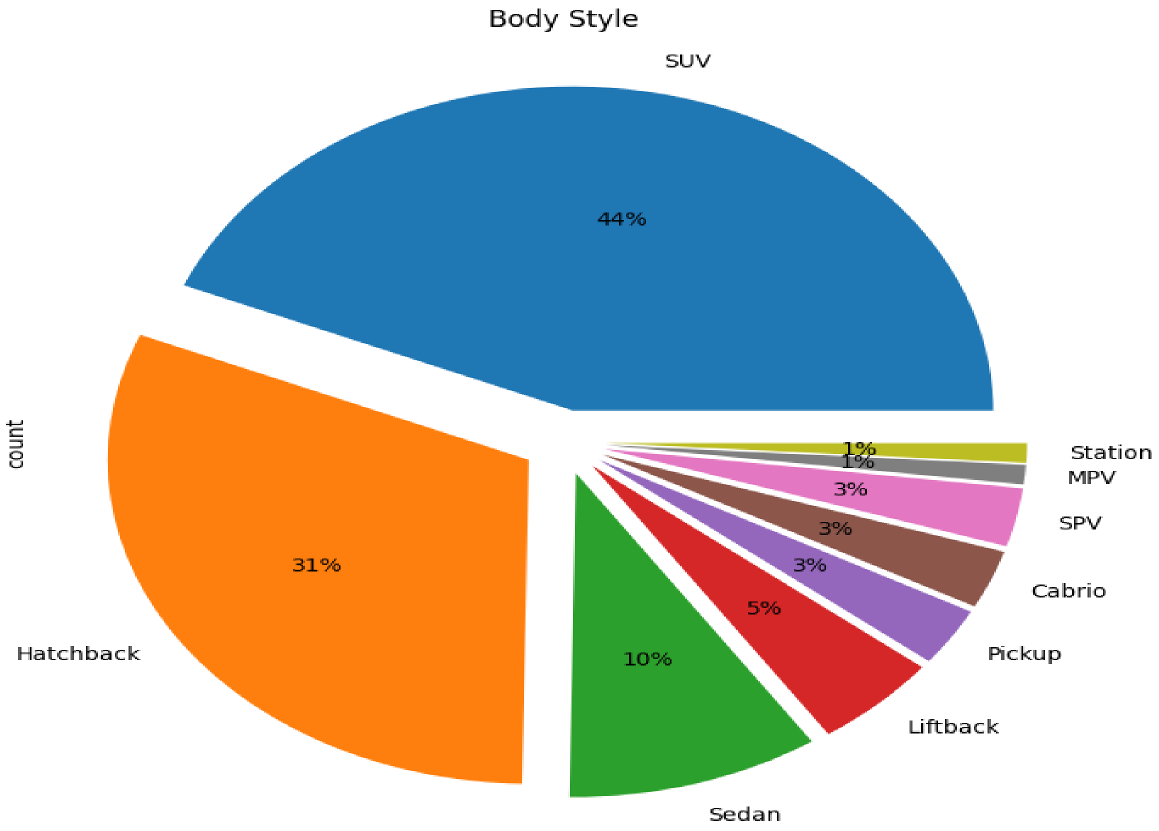


	AccelSec	TopSpeed_KmH	Range_Km	Efficiency_WhKm	FastCharge_KmH	RapidCharge	Seats	PriceEuro	inr(10e3)
AccelSec	1.00	-0.79	-0.68	-0.38	-0.73	-0.29	-0.18	-0.63	-0.63
TopSpeed_KmH	-0.79	1.00	0.75	0.36	0.79	0.22	0.13	0.83	0.83
Range_Km	-0.68	0.75	1.00	0.31	0.72	0.25	0.30	0.67	0.67
Efficiency_WhKm	-0.38	0.36	0.31	1.00	0.32	0.01	0.30	0.40	0.40
FastCharge_KmH	-0.73	0.79	0.72	0.32	1.00	0.23	0.19	0.67	0.67
RapidCharge	-0.29	0.22	0.25	0.01	0.23	1.00	0.20	0.20	0.20
Seats	-0.18	0.13	0.30	0.30	0.19	0.20	1.00	0.02	0.02
PriceEuro	-0.63	0.83	0.67	0.40	0.67	0.20	0.02	1.00	1.00
inr(10e3)	-0.63	0.83	0.67	0.40	0.67	0.20	0.02	1.00	1.00

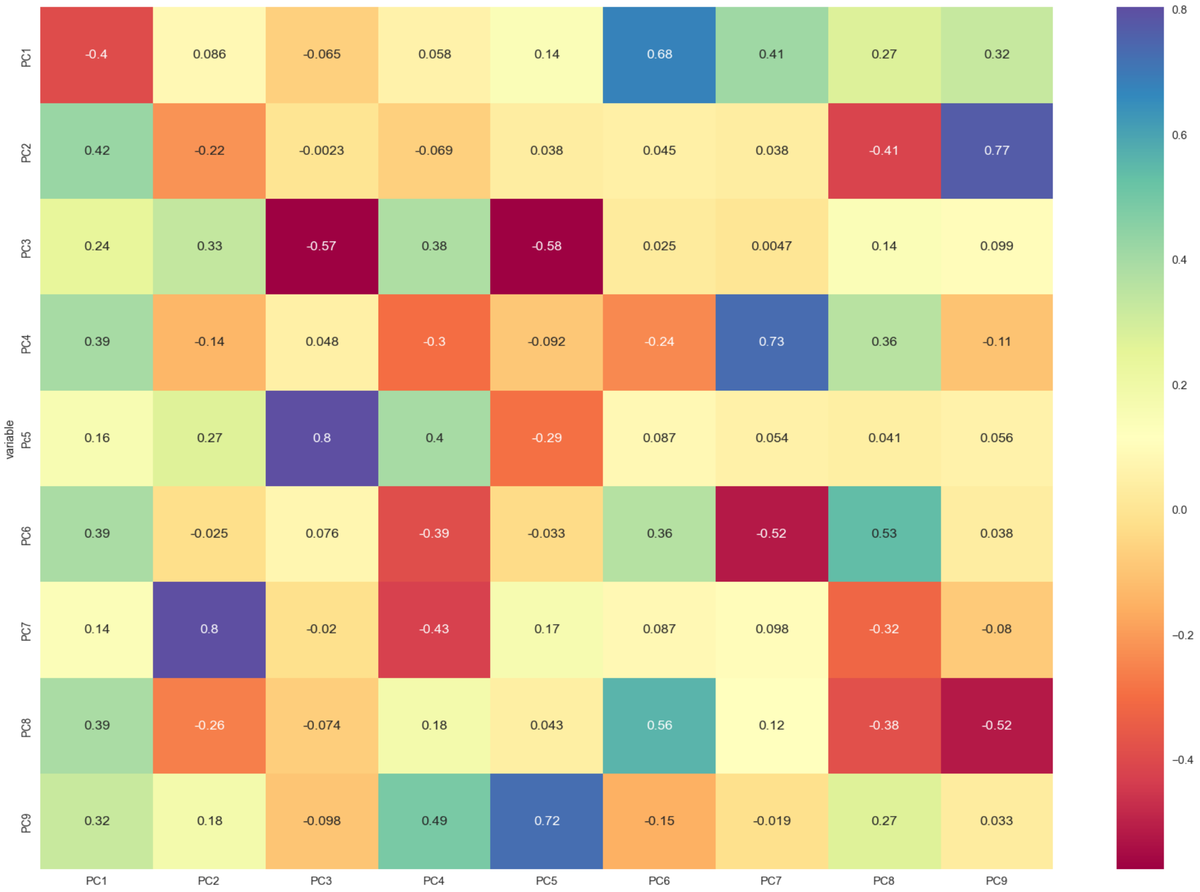
Cost based on top speed



Car and their body style



Correlation matrix plot for loadings



variable	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
PC1	-0.4	0.086	-0.065	0.058	0.14	0.68	0.41	0.27	0.32
PC2	0.42	-0.22	-0.0023	-0.069	0.038	0.045	0.038	-0.41	0.77
PC3	0.24	0.33	-0.57	0.38	-0.58	0.025	0.0047	0.14	0.099
PC4	0.39	-0.14	0.048	-0.3	-0.092	-0.24	0.73	0.36	-0.11
Pc5	0.16	0.27	0.8	0.4	-0.29	0.087	0.054	0.041	0.056
PC6	0.39	-0.025	0.076	-0.39	-0.033	0.36	-0.52	0.53	0.038
PC7	0.14	0.8	-0.02	-0.43	0.17	0.087	0.098	-0.32	-0.08
PC8	0.39	-0.26	-0.074	0.18	0.043	0.56	0.12	-0.38	-0.52
PC9	0.32	0.18	-0.098	0.49	0.72	-0.15	-0.019	0.27	0.033

Scree Plot

A **scree plot** is a graphical tool used in Principal Component Analysis (PCA) to determine the number of principal components (PCs) to retain for analysis. It provides a visual representation of how much variance each principal component explains in the data.

Structure of the Scree Plot:

- The **x-axis** represents the number of principal components (or factors).
- The **y-axis** represents the eigenvalues, which indicate the amount of variance explained by each principal component.

The plot typically forms a downward-sloping curve because:

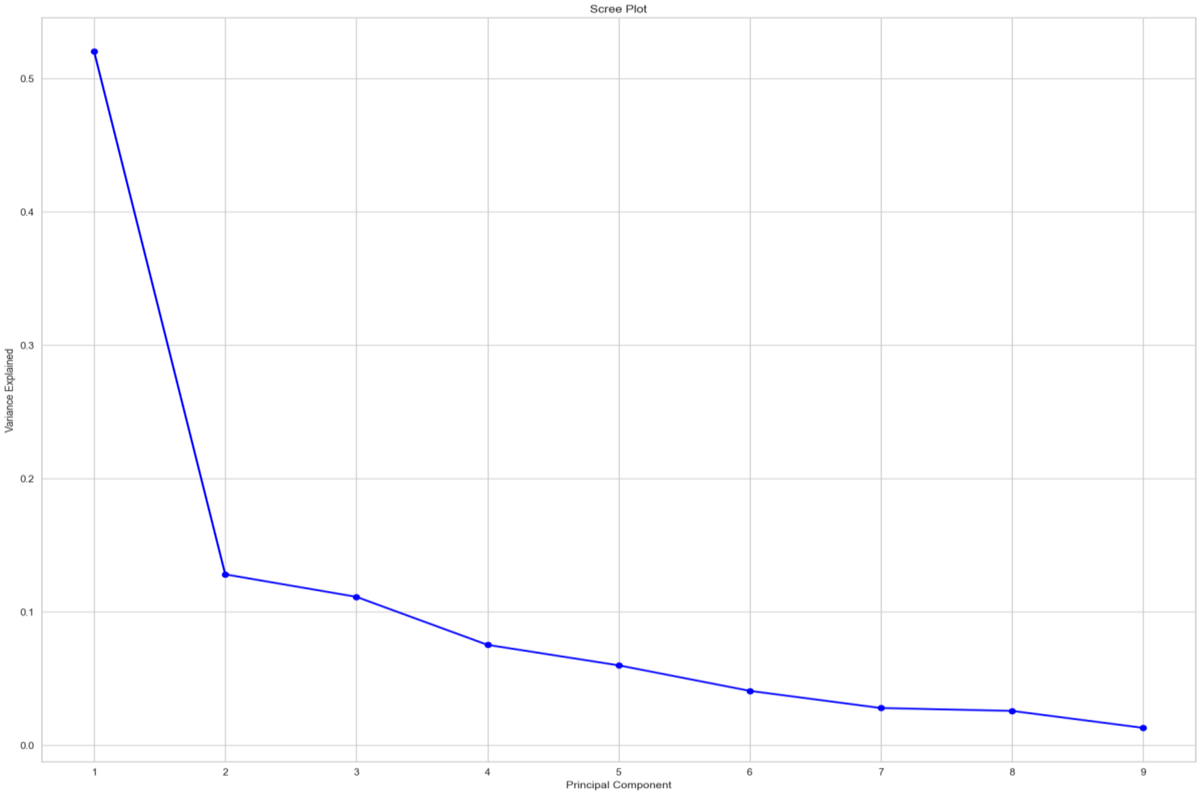
1. The first principal component explains the largest portion of the variance.
2. Subsequent components explain progressively smaller portions of the variance.
3. Toward the end, the components contribute very little and are mostly noise.

Key Features:

- **Elbow Point**: The scree plot often displays a sharp drop followed by a flattening of the curve. The "elbow" is the point where the drop slows significantly, marking the optimal number of components to retain.
- **Proportion of Variance**: A general rule is to retain enough components to explain at least **80% of the variance** in the data. This ensures the retained components capture the most critical information while eliminating redundancy.

Why Use a Scree Plot?

By identifying the elbow point and ensuring sufficient variance is explained, the scree plot helps simplify the dataset while retaining its most valuable features. This makes downstream tasks like classification, clustering, or regression more efficient and effective.



Extracting Segments Using a Dendrogram

A **dendrogram** is a visual tool used in hierarchical clustering, specifically the **agglomerative hierarchical method**, to determine the optimal number of clusters.

How Agglomerative Hierarchical Clustering Works:

1. Each data point is initially treated as an individual cluster.
2. Clusters are progressively merged based on their similarity (or distance), forming a hierarchical structure.
3. This process continues until all data points are grouped into a single cluster.

Role of the Dendrogram:

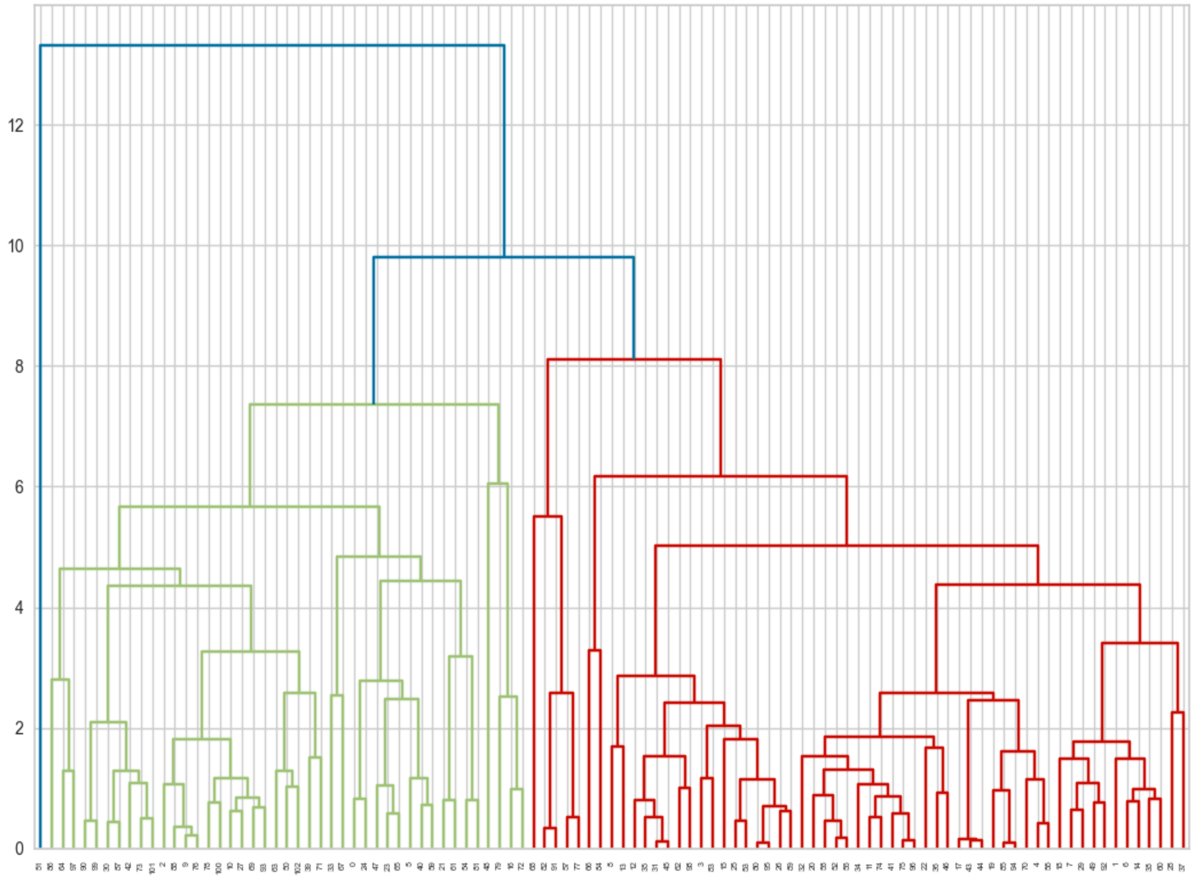
- A dendrogram is a tree-like chart that represents the sequence of cluster merges.
- When two clusters are merged, they are connected in the dendrogram, and the **height of the connection** represents the distance (or dissimilarity) between those clusters.

Identifying the Optimal Number of Clusters:

- To find the best number of clusters, we look for the **largest vertical distance** in the dendrogram that does not cross a horizontal merge line. This is often referred to as the **"cut-off point."**
- For example, if a dendrogram suggests a clear structure, we may choose 4–5 clusters based on where the hierarchy of merges indicates natural groupings.

Why Use a Dendrogram?

A dendrogram provides an intuitive way to visualize the clustering process and identify the most meaningful groups. Combined with other cluster validation metrics, it ensures that the selected number of clusters accurately represents the data's structure.

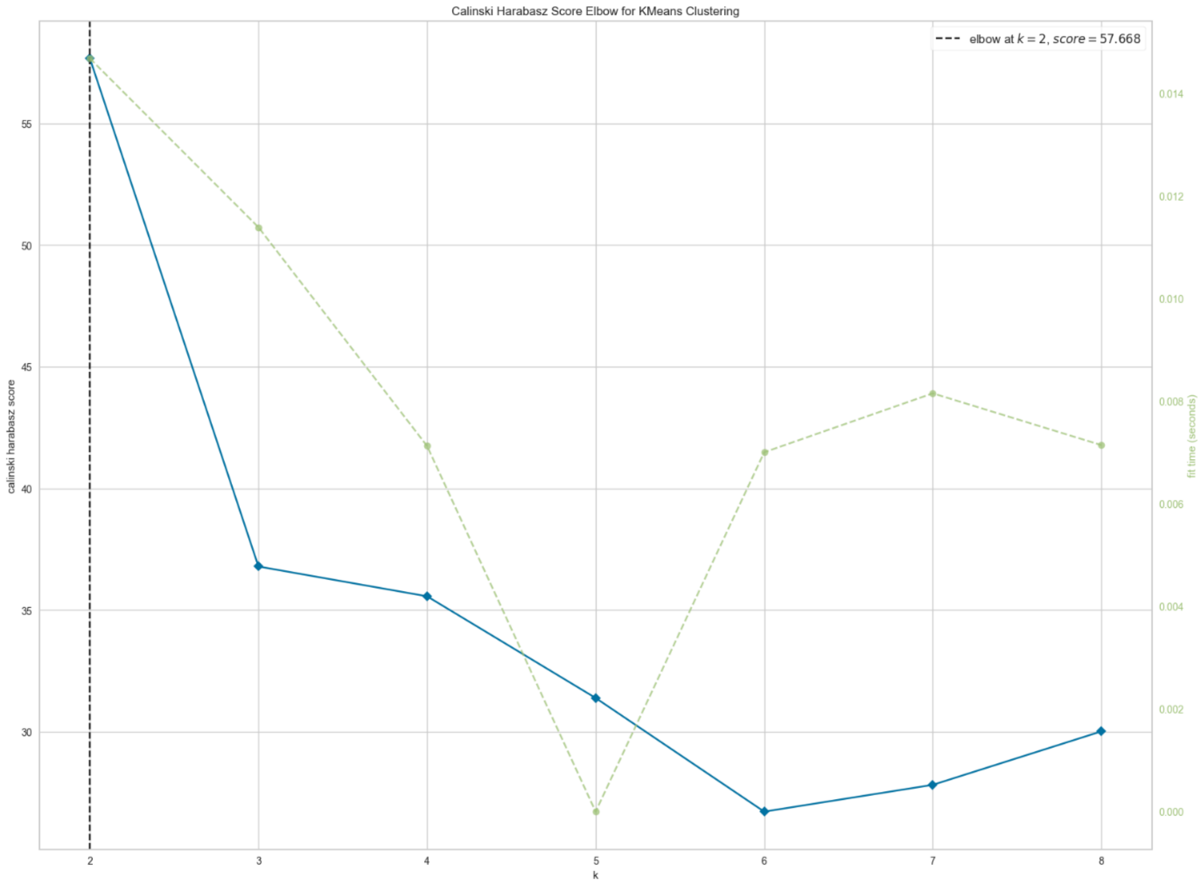


Elbow Method

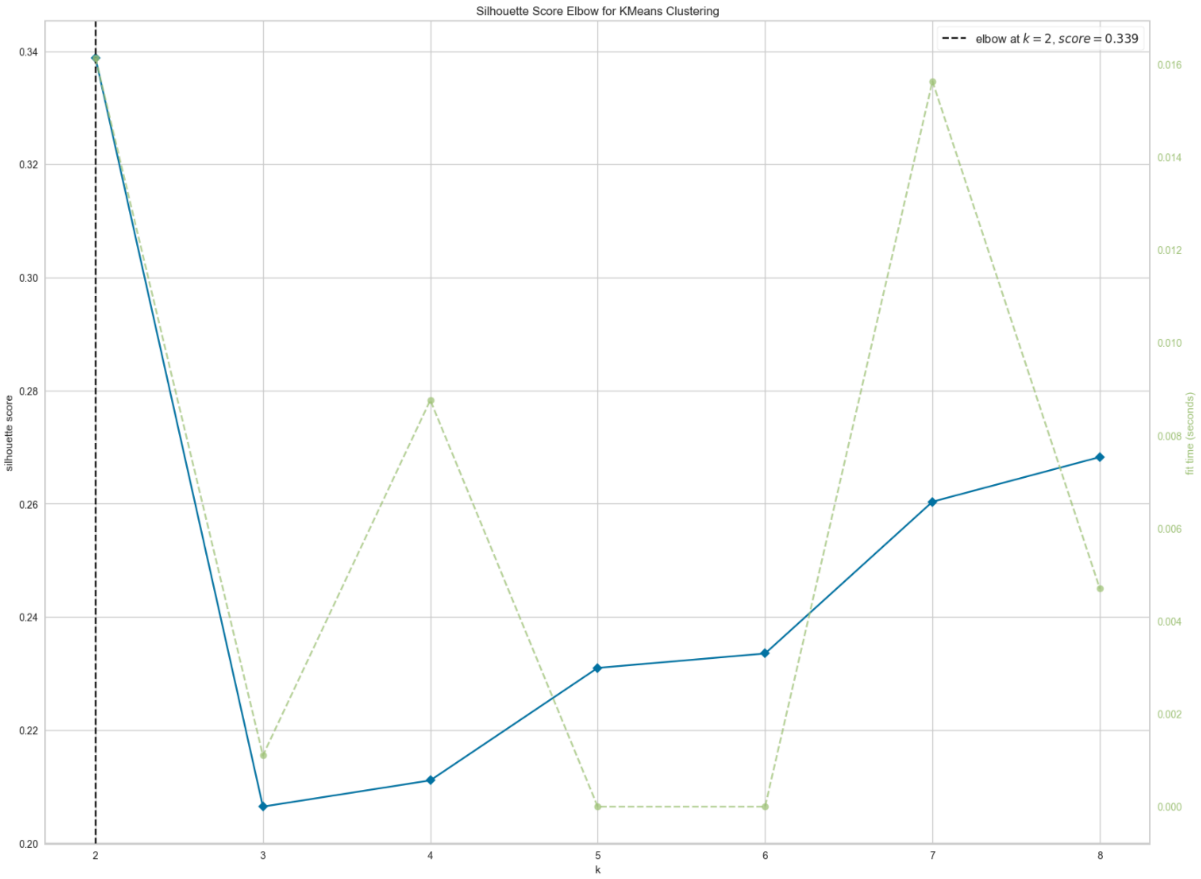
The **Elbow Method** is a simple and effective technique to determine the optimal number of clusters in a dataset. It works by calculating the **Within-Cluster Sum of Squared Errors (WSS)** for different values of clusters (k) and identifying the point where the WSS stops decreasing significantly, forming an "elbow" in the curve.

- **How it Works**:
  For small numbers of clusters, WSS decreases rapidly as adding clusters improves the grouping. However, after a certain point, the improvement slows down, creating the "elbow" in the plot. This elbow indicates the optimal number of clusters.

- **KElbowVisualizer**:
  This function automates the process by fitting a KMeans model for a range of clusters (e.g., 2 to 8) and highlights the elbow point. It also provides insights such as the time taken for each model, offering both accuracy and efficiency in choosing the cluster count.



Optimal number of clusters by elbow method



Analysis and Approaches for Segmentation: Clustering

Clustering is a widely used exploratory data analysis technique for uncovering the inherent structure within a dataset. The goal is to group data points into subgroups (or clusters) where points within the same cluster are highly similar, and points in different clusters are distinctly different. These subgroups are formed based on a chosen similarity measure, such as **Euclidean distance** or **correlation-based distance**, which is determined by the specific application.

Clustering can be performed:

1. **Based on Features**: Grouping samples by their attributes or characteristics.
2. **Based on Samples**: Grouping attributes by their shared characteristics across samples.

K-Means Algorithm

The **K-Means algorithm** is an iterative clustering method that divides the dataset into a predefined number of **K non-overlapping clusters**. Each data point is assigned to a single cluster based on its proximity to the cluster's centroid. The algorithm aims to minimize the **sum of squared distances (intra-cluster variance)** between data points and their cluster centroids, ensuring that clusters are as compact and distinct as possible.

Steps of K-Means Algorithm:

1. **Specify the Number of Clusters (K)**:
   Determine the desired number of clusters to segment the data.
2. **Initialize Cluster Centroids**:
   - Randomly shuffle the dataset.
   - Select K data points as the initial centroids without replacement.
3. **Assign Data Points to Clusters**:
   - Compute the distance of each data point to all centroids.
   - Assign each point to the cluster with the nearest centroid.
4. **Update Centroids**:
   - Recalculate the centroid of each cluster as the mean of all data points assigned to it.
5. **Repeat Until Convergence**:
   - Continue reassigning data points and updating centroids until there are no changes in cluster assignments or centroids.

Key Features of K-Means:

- **Minimizes Intra-Cluster Variance**: Ensures that data points within a cluster are as similar as possible.
- **Non-Overlapping Clusters**: Each data point belongs to only one cluster.
- **Application-Specific Distance Metrics**: While Euclidean distance is common, other metrics can be used depending on the dataset and goals.

K-Means is an efficient and widely used algorithm for applications like customer segmentation, pattern recognition, and anomaly detection, where simplicity and speed are critical.

Mathematical Approach of K-Means: Expectation-Maximization

The **K-Means algorithm** uses the **Expectation-Maximization (EM)** framework to solve the clustering problem iteratively. The process alternates between two main steps:

1. **E-Step (Expectation)**:
   Assign each data point to the nearest cluster based on the distance to the current cluster centroids.
2. **M-Step (Maximization)**:
   Recalculate the centroid of each cluster as the mean of all the data points assigned to it.

This iterative process minimizes the **objective function**, which measures the total squared distance between the data points and their respective cluster centroids.

Objective Function:

The goal of K-Means is to minimize the following function:

$$J = \sum_{k=1}^{K} \sum_{i=1}^{m} w_{ik} \|x_i - \mu_k\|^2$$

Where:

- **J**: Objective function (sum of squared distances).
- **K**: Total number of clusters.
- **m**: Total number of data points.
- **W_{ik}:** Indicator variable (1 if data point x_i belongs to cluster k, 0 otherwise).
- X_i: Data point.
- M_k: Centroid of cluster k.

M-Step (Centroid Update):

To update the centroids mathematically, we take the partial derivative of **J** with respect to µ_k and set it to zero:

$$\frac{\partial J}{\partial \mu_k} = \sum_{i=1}^{m} w_{ik}(x_i - \mu_k) = 0$$

Rearranging to solve for µ_k:

$$\mu_k = \frac{\sum_{i=1}^{m} w_{ik} x_i}{\sum_{i=1}^{m} w_{ik}}$$

This formula calculates the new centroid µ_k as the weighted average of all points assigned to cluster k, where the weights w_ik indicate cluster membership.

Key Insights:

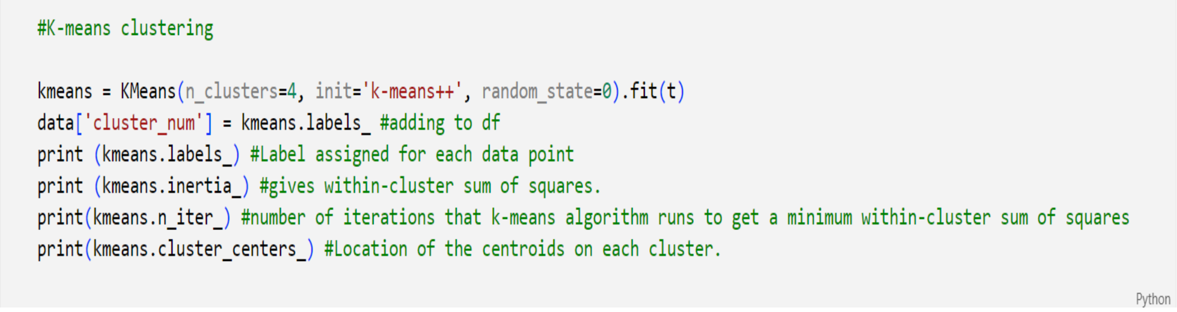
- The **E-Step** ensures data points are grouped with the closest centroid.
- The **M-Step** optimizes centroids to better represent the current cluster members.
- The algorithm repeats these steps until convergence, where cluster assignments no longer change or the improvement in the objective function becomes negligible.

Applications

K means algorithm is very popular and used in a variety of applications such as market segmentation, document clustering, image segmentation and image compression, etc. The goal usually when we undergo a cluster analysis is either:

1. Get a meaningful intuition of the structure of the data we’re dealing with.
2. Cluster-then-predict where different models will be built for different subgroups if we believe there is a wide variation in the behaviors of different subgroups.

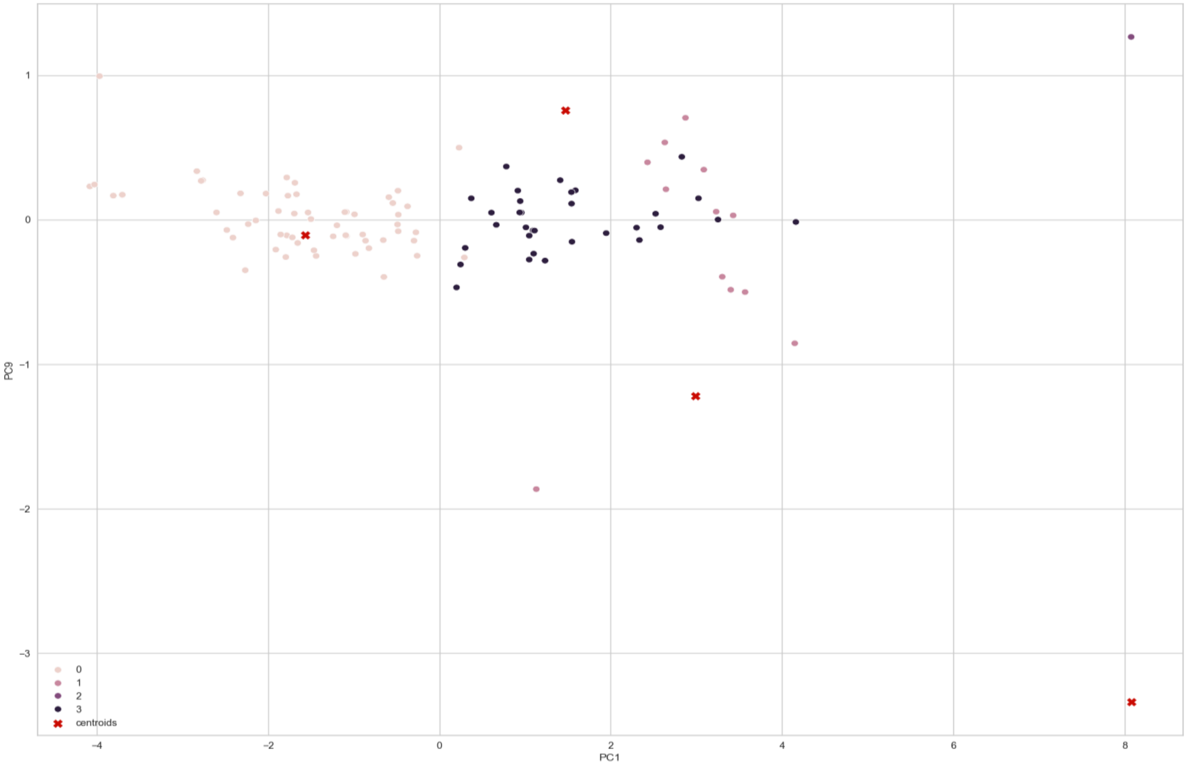
According to the Elbow method, here we take K=4 clusters to train KMeans model. The derived clusters are shown in the following figure



```
#K-means clustering

kmeans = KMeans(n_clusters=4, init='k-means++', random_state=0).fit(t)
data['cluster_num'] = kmeans.labels_ #adding to df
print (kmeans.labels_) #Label assigned for each data point
print (kmeans.inertia_) #gives within-cluster sum of squares.
print(kmeans.n_iter_) #number of iterations that k-means algorithm runs to get a minimum within-cluster sum of squares
print(kmeans.cluster_centers_) #Location of the centroids on each cluster.
```

Visualizing clusters



Predicting Prices of Commonly Used Cars: Linear Regression Approach

Linear Regression is a supervised machine learning algorithm primarily used for regression tasks. It predicts a target variable (dependent variable) based on the values of one or more independent variables. The main purpose of this algorithm is to establish a relationship between variables and make future predictions.

In this scenario, **Linear Regression** is applied to predict the prices of electric cars across various companies. Here's how the process works:

Model Setup and Training

1. **Features and Target Variables**:
   - **Independent Variables (X)**: These are the factors influencing car prices (e.g., brand, mileage, battery capacity).
   - **Dependent Variable (y)**: The price of the car, which the model is tasked to predict.
2. **Data Splitting**:
   - The dataset is split into **training and testing subsets** in a 40:60 ratio.
   - 40% of the data is used to train the model, ensuring the model learns patterns in the data.
3. **Training the Model**:
   - The LinearRegression().fit(X_train, y_train) method is used to fit the training data to the model.
   - During this step, the algorithm learns the relationship between the features XXX and the target variable yyy.

Key Outputs

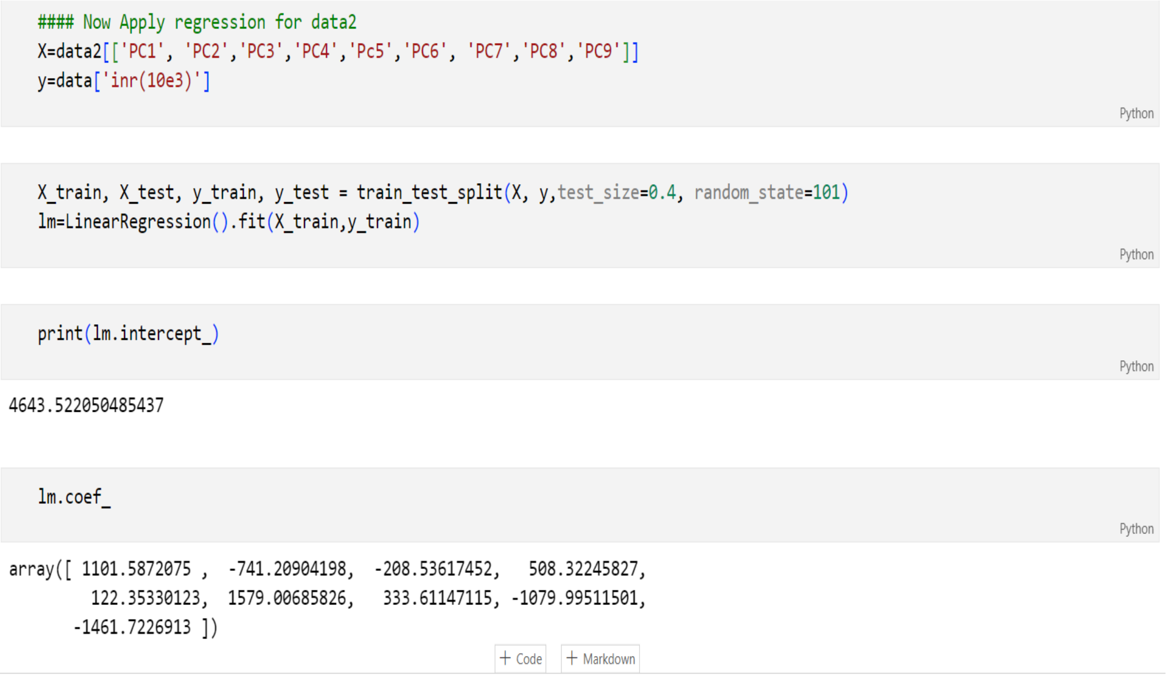
1. **Intercept**:
   - Represents the base value of the dependent variable when all independent variables are zero.
2. **Coefficients**:

- Indicates how much the dependent variable changes with a one-unit change in an independent variable, assuming other variables remain constant.

3. **Cumulative Distribution Function (CDF)**:
   - CDF provides a probability-based interpretation, showing the likelihood of a predicted price falling within a specific range.

Advantages of Linear Regression in Car Price Prediction

- It provides interpretable results, allowing us to understand how each feature influences car prices.
- The simplicity of the model ensures faster computations, even with large datasets.
- Forecasting capabilities make it ideal for predicting car prices based on company-specific or feature-specific data.



```
#### Now Apply regression for data2
X=data2[['PC1', 'PC2','PC3','PC4','Pc5','PC6', 'PC7','PC8','PC9']]
y=data['inr(10e3)']
```

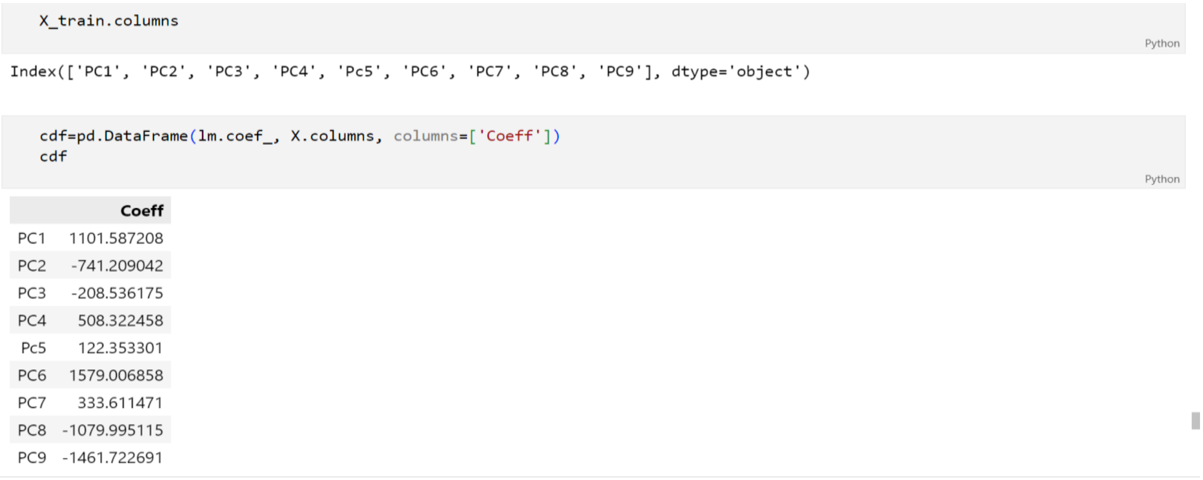
```
X_train, X_test, y_train, y_test = train_test_split(X, y,test_size=0.4, random_state=101)
lm=LinearRegression().fit(X_train,y_train)
```

```
print(lm.intercept_)
```

```
4643.522050485437
```

```
lm.coef_
```

```
array([ 1101.5872075 ,  -741.20904198,  -208.53617452,   508.32245827,
         122.35330123,  1579.00685826,   333.61147115, -1079.99511501,
       -1461.7226913 ])
```



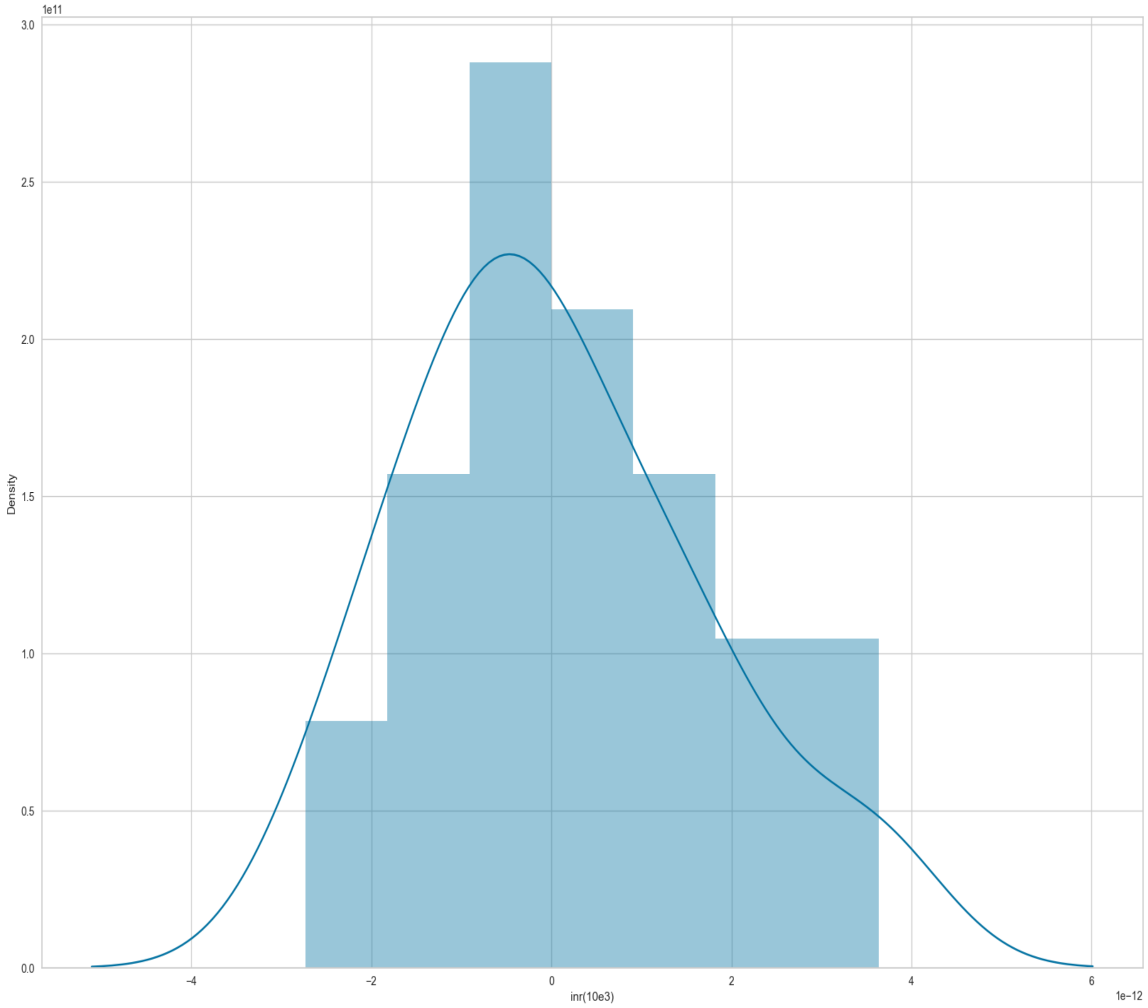
```
X_train.columns
```

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

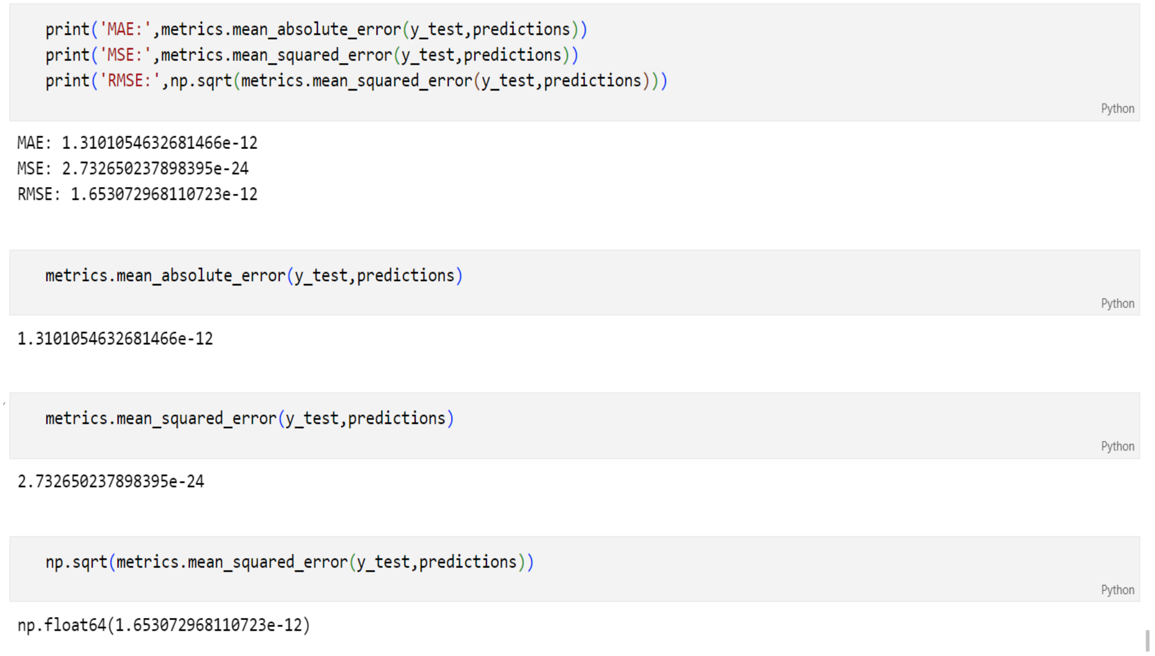
```
cdf=pd.DataFrame(lm.coef_, X.columns, columns=['Coeff'])
cdf
```

	Coeff
PC1	1101.587208
PC2	-741.209042
PC3	-208.536175
PC4	508.322458
Pc5	122.353301
PC6	1579.006858
PC7	333.611471
PC8	-1079.995115
PC9	-1461.722691

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



The metrics of the algorithm, Mean absolute error, Mean squared error and mean square root error are described in the below figure:



```
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: 1.3101054632681466e-12
MSE: 2.732650237898395e-24
RMSE: 1.653072968110723e-12
```

```
metrics.mean_absolute_error(y_test,predictions)
```

```
1.3101054632681466e-12
```

```
metrics.mean_squared_error(y_test,predictions)
```

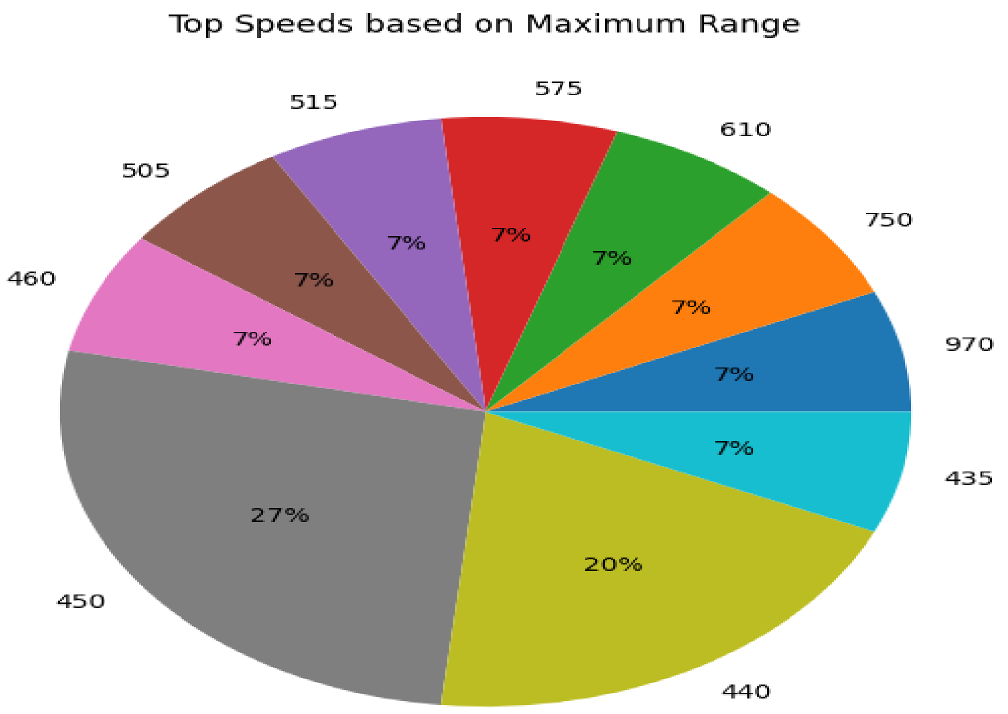
```
2.732650237898395e-24
```

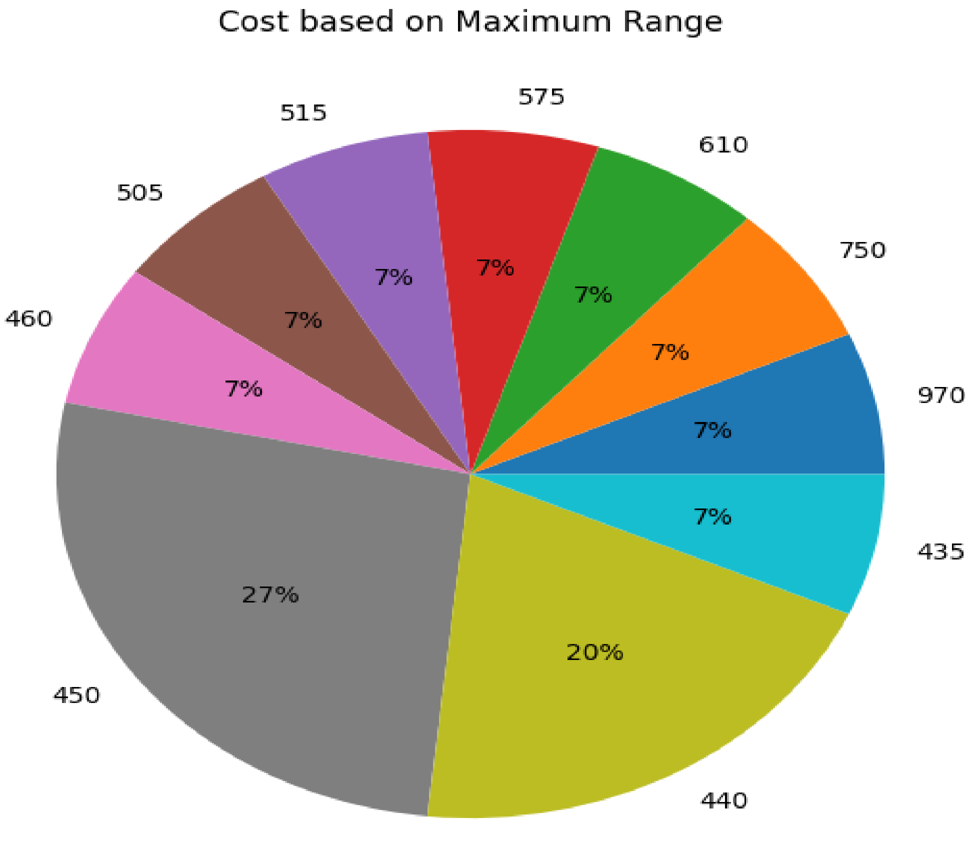
```
np.sqrt(metrics.mean_squared_error(y_test,predictions))
```

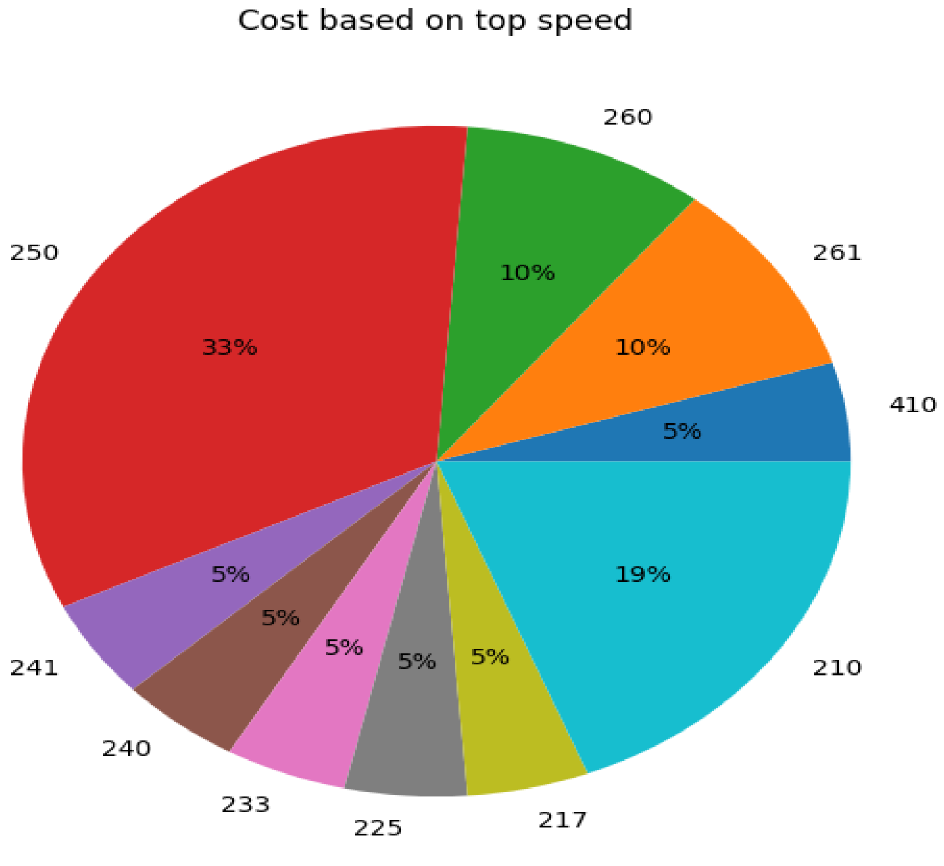
```
np.float64(1.653072968110723e-12)
```

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:




Cost based on Maximum Range
515
575
505
610
750
460
970
435
450
440
7%
7%
7%
7%
7%
7%
7%
7%
27%
20%


Cost based on top speed
260
250
261
410
241
210
240
233
225
217
10%
10%
33%
5%
5%
5%
5%
5%
5%
5%
19%

Target Segments for Optimal Focus

Based on the analysis, the optimal target segment for cars aligns with specific behavioral, demographic, and psychographic factors. These factors help define a customer base that aligns with the product's offerings and market demands:

Behavioral Segment

- **Seating Capacity**: Most cars in the target segment are observed to have **5 seats**, indicating a preference for mid-sized, family-friendly vehicles.

Demographic Segment

- **Top Speed and Range**:
    - Cars with **high top speeds** and **maximum range** are in higher demand.
    - These factors significantly influence the cost and appeal to a broader market.
- **Efficiency**:
    - The majority of cars in the target segment exhibit **high efficiency**, making them attractive to environmentally-conscious and cost-aware consumers.

Psychographic Segment

- **Price Range**:
    - The target price range for the segment is between **₹16,00,000 to ₹1,80,00,000**, appealing to upper-middle-class to premium buyers who value performance and luxury.

Final Target Segment Recommendation

The ideal target segment should focus on:

1. Cars with **high efficiency** for cost-effectiveness and sustainability.
2. Vehicles that offer **top speed** and **long-range capabilities**, catering to performance-oriented buyers.

3. A **price range of ₹16 to ₹180 lakhs**, ensuring competitiveness in the premium and high-performance market.

4. **5-seat configuration**, which meets the needs of families and small groups.

This focus allows for a balanced approach, aligning with market preferences while addressing customer needs for performance, efficiency, and affordability.

References

1. **Google Images**: Visual aids and diagrams were sourced from Google Images to better understand the concepts and methodologies related to market segmentation.

2. **McDonald's Case Study**: The McDonald's case study was referred to as an example of practical application of market segmentation, providing insights into real-world strategies used by a global brand.

3. **Dolnicar, S., Grün, B., & Leisch, F. (2018).** *Market Segmentation Analysis: Understanding It, Doing It, and Making It Useful.* Management for Professionals Series. Springer International Publishing.

    - This book served as a foundational reference for understanding the theoretical and practical aspects of market segmentation, offering in-depth coverage of methods and applications.

These references provided valuable insights and helped structure the analysis and findings in this report.

Linkedin: https://www.linkedin.com/in/amrendra-kumar-9954b9225/

Github: https://github.com/Amrendra-kum