SEGMENTATION ANALYSIS ON EV MARKET IN INDIA

-JILAKARA MADHU

**OVERVIEW:**

The electric vehicle (EV) market in India has experienced significant growth and development in recent years. The government's push for clean and sustainable transportation, coupled with increasing awareness about environmental issues, has driven the demand for electric vehicles in the country. Here's an overview of the EV market in India:

1. Government Initiatives: The Indian government has implemented various policies and initiatives to promote the adoption of electric vehicles. These include the Faster Adoption and Manufacturing of (Hybrid &) Electric Vehicles (FAME) scheme, which provides financial incentives for EV buyers and support for charging infrastructure development. The government has also set a target to achieve 30% EV penetration in new vehicle sales by 2030.

2. Rising Demand: The demand for electric vehicles in India has been steadily increasing. Factors such as rising fuel prices, concerns over air pollution, and favorable government policies have contributed to the growing interest in EVs. Additionally, the availability of a wide range of electric vehicle models across different segments has further fueled consumer demand.

3. Vehicle Segments: The EV market in India consists of various segments, including two-wheelers, three-wheelers, passenger cars, and commercial vehicles. Two-wheelers, especially electric scooters, have seen significant adoption due to their affordability and suitability for urban commuting. Three-wheelers, such as e-rickshaws and e-autos, are also popular for short-distance transportation. The adoption of electric passenger cars and commercial vehicles is gradually gaining traction.

4. Charging Infrastructure: The availability of a robust charging infrastructure is essential for the growth of the EV market. India has witnessed an increase in the number of public and private charging stations across major cities. The government aims to establish a network of charging stations at regular intervals on highways, as well as in urban areas and public parking lots.

5. Manufacturing and Investments: Several domestic and international automakers have entered the Indian EV market or announced plans to do so. Major automotive companies are investing in the development and manufacturing of electric vehicles in India. The government has also encouraged domestic manufacturing through incentives and initiatives such as the "Make in India" campaign.

6. Challenges: Despite the growth, the EV market in India faces certain challenges. These include high upfront costs of electric vehicles compared to conventional vehicles, limited driving range, lack of awareness about EV technology, and the need for further infrastructure development, especially in rural areas. Addressing these challenges is crucial to accelerate the adoption of electric vehicles in India.

The EV market in India is witnessing significant growth and government support. With favorable policies, increasing investments, and a growing charging infrastructure, the country is poised to become a major market for electric vehicles, contributing to sustainable and eco-friendly transportation.

**DATA SOURCE:**

Collected from various source

Google

Real world problem statement

New startups

**DATA PRE-PROCESSING:**

Data pre-processing is an essential step in machine learning that involves transforming raw data into a format suitable for analysis and model training. It includes several techniques to clean, normalize, and prepare the data to improve the performance and accuracy of machine learning models. Here are some common data pre-processing steps in machine learning:

1. Data Cleaning: This step involves handling missing data, outliers, and inconsistencies in the dataset. Missing data can be imputed using techniques like mean, median, or regression imputation. Outliers can be identified and treated by either removing them or replacing them with more appropriate values. Inconsistencies and errors in the data can be corrected or eliminated through data validation and data quality checks.

2. Data Integration: In many cases, data may be spread across multiple sources or in different formats. Data integration involves combining data from various sources into a consistent format to create a unified dataset. This ensures that the data is complete and suitable for analysis.

3. Data Transformation: Data transformation involves converting the data into a suitable format for analysis. This includes scaling and normalizing numerical features to ensure they are on a similar scale. Common techniques for scaling include min-max scaling and standardization. Categorical variables may be encoded into numerical values using techniques like one-hot encoding or label encoding.

4. Feature Selection: Feature selection is the process of selecting the most relevant features from the dataset for model training. Irrelevant or redundant features can negatively impact model performance and increase complexity. Techniques such as correlation analysis, feature importance estimation, and dimensionality reduction methods like Principal Component Analysis (PCA) can be used for feature selection.

5. Data Splitting: The dataset is typically divided into training and testing sets. The training set is used to train the model, while the testing set is used to evaluate the model's performance. This step helps assess how well the model generalizes to new, unseen data. Additionally, techniques like cross-validation can be employed to validate the model's performance and reduce overfitting.

6. Handling Imbalanced Data: In cases where the dataset is imbalanced, meaning some classes have significantly more or fewer instances than others, techniques like oversampling, undersampling, or using synthetic data generation methods (e.g., SMOTE - Synthetic Minority Over-sampling Technique) can be applied to balance the dataset.

7. Handling Text or Categorical Data: Text or categorical data may require additional preprocessing steps such as tokenization, stemming or lemmatization (for text data), and encoding categorical variables using techniques like one-hot encoding or ordinal encoding.

These are some common data pre-processing steps in machine learning, but the specific techniques used may vary depending on the nature of the dataset and the requirements of the machine learning task at hand. Effective data pre-processing helps ensure that the data is in a suitable form for model training, leading to better model performance and more accurate predictions.

**SEGMENT EXTRACTION:**

Here, I use k-means clustering for analysis the problem statement.

Segment extraction using K-means clustering is a technique used in machine learning to group similar data points into distinct segments or clusters. K-means clustering is an unsupervised learning algorithm that aims to partition data into K clusters based on their similarity. Here's a step-by-step overview of segment extraction using K-means clustering:

1. Data Preparation: First, you need to prepare your dataset for clustering. Ensure that your dataset is pre-processed, cleaned, and transformed into a suitable format for analysis. Numeric features should be scaled or normalized if necessary.

2. Determine the Number of Clusters (K): Decide on the number of clusters you want to extract from the data. This can be based on prior knowledge or using techniques like the elbow method or silhouette analysis to find an optimal value of K.

3. Initialize Cluster Centers: Randomly select K data points as initial cluster centers or use a more advanced initialization method like K-means++.

4. Assign Data Points to Clusters: Calculate the distance (e.g., Euclidean distance) between each data point and the cluster centers. Assign each data point to the cluster with the nearest center.

5. Update Cluster Centers: Recalculate the cluster centers by computing the mean of the data points assigned to each cluster. This will update the centroids of the clusters.

6. Repeat Steps 4 and 5: Iteratively repeat steps 4 and 5 until convergence criteria are met. Convergence criteria can be defined based on a maximum number of iterations or when the cluster centers no longer significantly change.

7. Evaluate Cluster Results: Assess the quality and interpretability of the obtained clusters. You can analyze the characteristics of each cluster, such as the mean or mode of the features, to understand the unique properties of each segment.

8. Assign New Data Points: Once you have obtained the final cluster centers, you can use the model to assign new data points to the appropriate clusters based on their similarity to the cluster centers.

K-means clustering is an iterative algorithm, and the results can vary based on the initial cluster centers and the convergence criteria. It is important to consider the limitations of K-means clustering, such as its sensitivity to the initial conditions and the assumption that clusters are spherical and have similar sizes. Additionally, feature selection and scaling can impact the clustering results, so it's important to choose relevant features and preprocess the data appropriately before applying K-means clustering. Overall, segment extraction using K-means clustering can provide valuable insights into the structure and patterns present in your data, helping you understand and categorize different groups within the dataset.

**PROFILING AND DESCRIBING POTENTIAL SEGMENTS:**

**IMPORT LIBRARIES**

*#Import Libraries*

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import sklearn

%matplotlib inline

import warnings

warnings.filterwarnings('ignore')

*#Read File*

df\_Original = pd.read\_csv('/kaggle/input/ev-data/EV\_Data.csv')

pd.set\_option('display.float\_format', lambda x: '**%.3f**' % x)

df = df\_Original.copy()

df.head()

|  | Unnamed: 0 | Age | City | Profession | Marital Status | Education | No. of Family members | Annual Income | Would you prefer replacing all your vehicles to Electronic vehicles? | If Yes/Maybe what type of EV would you prefer? | Do you think Electronic Vehicles are economical? | Which brand of vehicle do you currently own? | How much money could you spend on an Electronic vehicle? | Preference for wheels in EV | Do you think Electronic vehicles will replace fuel cars in India? |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 30 | Nabha | None | Single | Graduate | 5 | 1193875.647 | Maybe | SUV | Yes | Hyundai | <5 lakhs | 2 | I don't think so |
| 1 | 1 | 27 | Pune | None | Single | Graduate | 4 | 1844540.398 | Yes | SUV | Yes | Honda | <15 lakhs | 4 | Yes, in <20years |
| 2 | 2 | 32 | Kashipur | None | Single | Graduate | 4 | 2948150.113 | Yes | Hatchback | Yes | KIA | <15 lakhs | 4 | Yes, in <20years |
| 3 | 3 | 55 | Pune | Business | Single | Graduate | 3 | 2832379.739 | Maybe | Hatchback | No | Hyundai | <5 lakhs | 4 | Yes, in <10 years |
| 4 | 4 | 26 | Satara | None | Single | Graduate | 4 | 2638750.576 | Yes | Sedan | Yes | McLaren | <15 lakhs | 4 | Yes, in <20years |

**DATA PREPROCESSING**

df.isnull().sum()

Unnamed: 0 0

Age 0

City 0

Profession 0

Marital Status 0

Education 0

No. of Family members 0

Annual Income 0

Would you prefer replacing all your vehicles to Electronic vehicles? 0

If Yes/Maybe what type of EV would you prefer? 0

Do you think Electronic Vehicles are economical? 0

Which brand of vehicle do you currently own? 0

How much money could you spend on an Electronic vehicle? 0

Preference for wheels in EV 0

Do you think Electronic vehicles will replace fuel cars in India? 0

dtype: int64

In [5]:

df.columns

Out[5]:

Index(['Unnamed: 0', 'Age', 'City', 'Profession', 'Marital Status',

'Education', 'No. of Family members', 'Annual Income',

'Would you prefer replacing all your vehicles to Electronic vehicles?',

'If Yes/Maybe what type of EV would you prefer?',

'Do you think Electronic Vehicles are economical?',

'Which brand of vehicle do you currently own?',

'How much money could you spend on an Electronic vehicle?',

'Preference for wheels in EV',

'Do you think Electronic vehicles will replace fuel cars in India?'],

dtype='object')

In [6]:

df.shape

Out[6]:

(1000, 15)

In [7]:

*#Data Preprocessing*

df.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1000 entries, 0 to 999

Data columns (total 15 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Unnamed: 0 1000 non-null int64

1 Age 1000 non-null int64

2 City 1000 non-null object

3 Profession 1000 non-null object

4 Marital Status 1000 non-null object

5 Education 1000 non-null object

6 No. of Family members 1000 non-null int64

7 Annual Income 1000 non-null float64

8 Would you prefer replacing all your vehicles to Electronic vehicles? 1000 non-null object

9 If Yes/Maybe what type of EV would you prefer? 1000 non-null object

10 Do you think Electronic Vehicles are economical? 1000 non-null object

11 Which brand of vehicle do you currently own? 1000 non-null object

12 How much money could you spend on an Electronic vehicle? 1000 non-null object

13 Preference for wheels in EV 1000 non-null int64

14 Do you think Electronic vehicles will replace fuel cars in India? 1000 non-null object

dtypes: float64(1), int64(4), object(10)

memory usage: 117.3+ KB

In [8]:

df['Age'].unique()

Out[8]:

array([ 30, 27, 32, 55, 26, 28, 23, 25, 43, 59, 21, 29, 56,

70, 50, 24, 61, 39, 31, 40, 18, 58, 22, 96, 64, 52,

54, 42, 49, 57, 46, 36, 20, 19, 65, 17, 60, 44, 45,

47, 82, 33, 37, 48, 69, 67, 86, 62, 66, 34, 63, 41,

68, 16, 53, 15, 118, 38])

In [9]:

df['City'].unique()

Out[9]:

array(['Nabha', 'Pune', 'Kashipur ', 'Satara', 'Noida', 'Delhi', 'Mumbai',

'pune', 'solapur', 'Haldwani ', 'Nellore ', 'Pune ', 'Haldwani',

'Banglore ', 'Faridabad ', 'Nagpur', 'Chandrapur ', 'Chennai',

'Gurugram ', 'Nashik', 'Bengaluru', 'Mumbai ', 'Hakdwani',

'Patiyala', 'pUNE', 'Ahmedabad', 'Karnal', 'Rewari', 'New Delhi',

'Serampore', 'Jhansi', 'New Delhi ', 'Jalandhar', 'Delhi ',

'nashik'], dtype=object)

In [10]:

df["City"] = df["City"].replace({"Pune":"Pune", "pUNE": "Pune", "pune": "Pune", "Pune ": "Pune"})

df["City"] = df["City"].replace({"Mumbai ":"Mumbai", "Mumbai": "Mumbai"})

df["City"] = df["City"].replace({"Banglore ":"Bengaluru"})

df["City"] = df["City"].replace({"Delhi":"New Delhi", "Delhi ": "New Delhi", "New Delhi ": "New Delhi"})

df["City"] = df["City"].replace({"Hakdwani":"Haldwani", "Haldwani ": "Haldwani"})

df["City"] = df["City"].replace({"nashik":"Nashik"})

In [11]:

df['No. of Family members'].unique()

Out[11]:

array([5, 4, 3, 2, 8, 6, 0, 1, 7])

In [12]:

df['How much money could you spend on an Electronic vehicle?'].unique()

Out[12]:

array(['<5 lakhs', '<15 lakhs', '<25 lakhs', '700000', '>25 lakhs',

'2000000', '1200000', '1500000'], dtype=object)

In [13]:

df.columns

Out[13]:

Index(['Unnamed: 0', 'Age', 'City', 'Profession', 'Marital Status',

'Education', 'No. of Family members', 'Annual Income',

'Would you prefer replacing all your vehicles to Electronic vehicles?',

'If Yes/Maybe what type of EV would you prefer?',

'Do you think Electronic Vehicles are economical?',

'Which brand of vehicle do you currently own?',

'How much money could you spend on an Electronic vehicle?',

'Preference for wheels in EV',

'Do you think Electronic vehicles will replace fuel cars in India?'],

dtype='object')

In [14]:

df.drop('Unnamed: 0', axis=1, inplace = True)

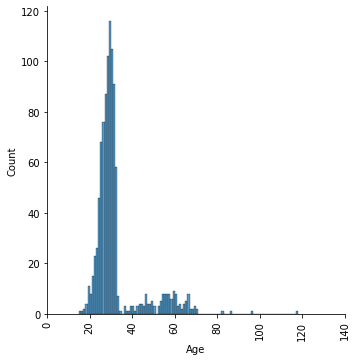
**DATA VISUALIZATION**

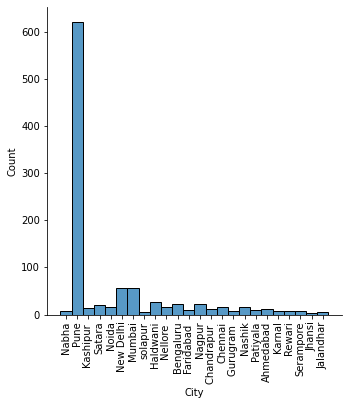
In [15]:

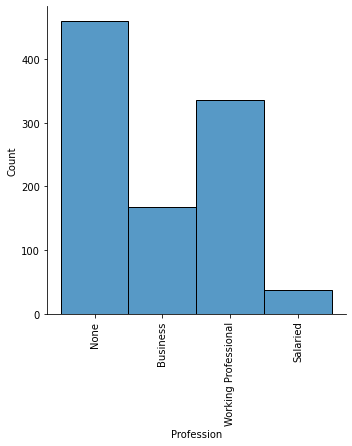
for col **in** df.columns:

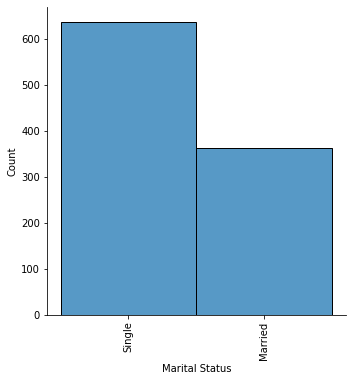
ax= sns.displot(df[col])

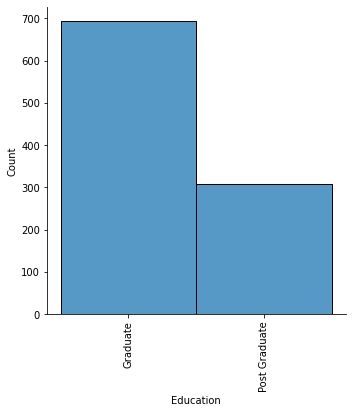
ax.set\_xticklabels(rotation=90)

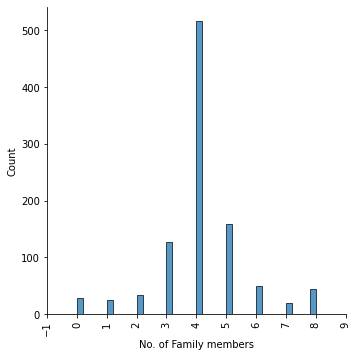


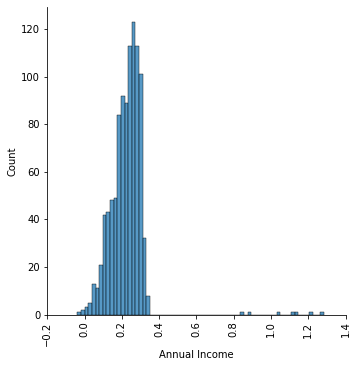


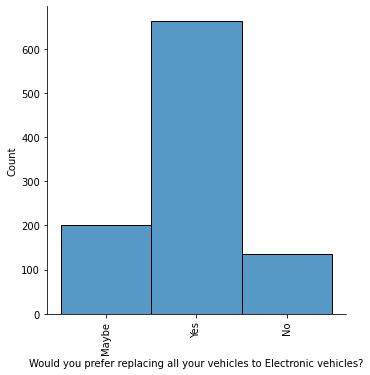


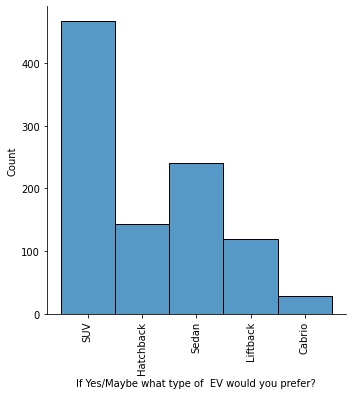


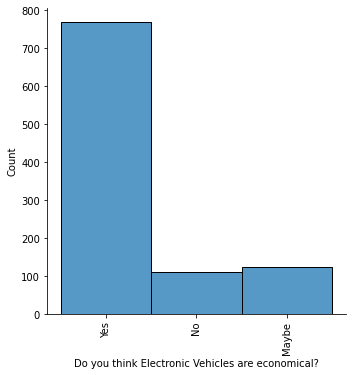


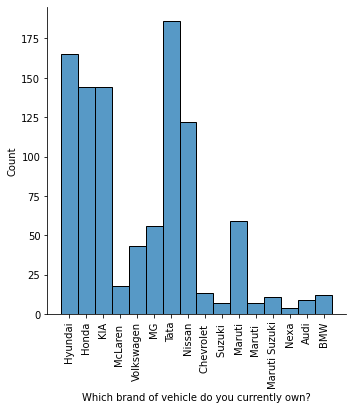


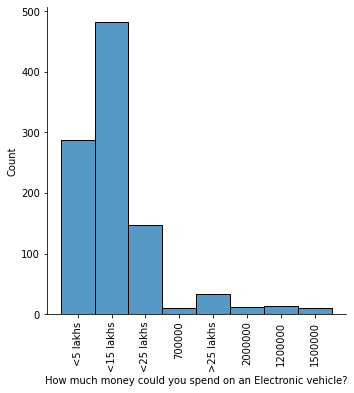


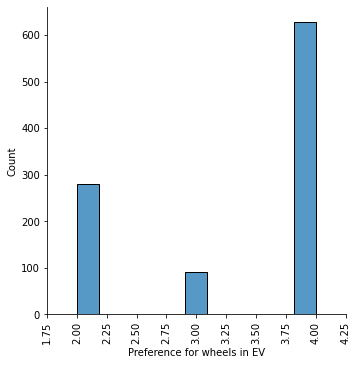


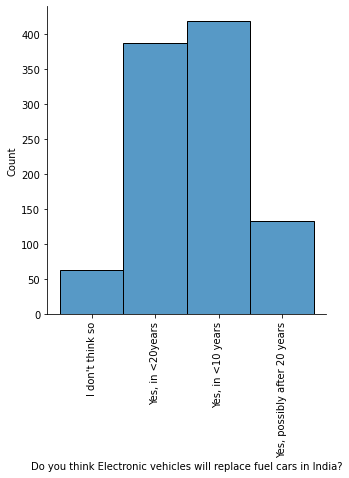












In [16]:

*#Most people prefer 4 wheels vehicle.*

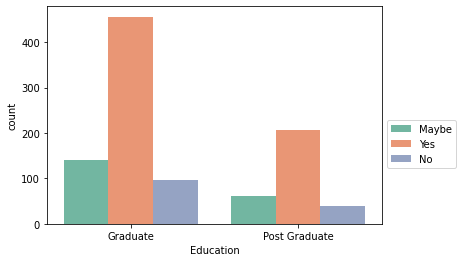
*#Most people believe Electronic vehicle will replace fuel cars in India in less than 20 years.*

In [17]:

sns.countplot(x ='Education', hue = 'Would you prefer replacing all your vehicles to Electronic vehicles?', data = df, palette = 'Set2')

plt.legend(bbox\_to\_anchor=(1, 0.5))

plt.show()

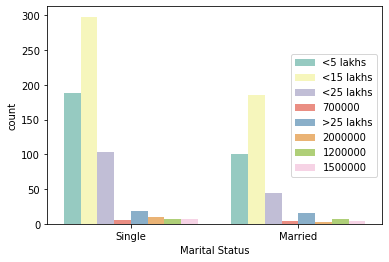


In [18]:

sns.countplot(x ='Marital Status', hue = 'How much money could you spend on an Electronic vehicle?', data = df, palette = 'Set3')

plt.legend(loc='center right')

plt.show()



In [19]:

df.describe()

Out[19]:

|  | Age | No. of Family members | Annual Income | Preference for wheels in EV |
| --- | --- | --- | --- | --- |
| count | 1000.000 | 1000.000 | 1000.000 | 1000.000 |
| mean | 31.800 | 4.118 | 2258341.824 | 3.349 |
| std | 11.295 | 1.470 | 999355.758 | 0.888 |
| min | 15.000 | 0.000 | -376150.863 | 2.000 |
| 25% | 26.000 | 4.000 | 1782115.520 | 2.000 |
| 50% | 29.000 | 4.000 | 2329246.375 | 4.000 |
| 75% | 31.000 | 5.000 | 2753169.612 | 4.000 |
| max | 118.000 | 8.000 | 12821282.030 | 4.000 |

In [20]:

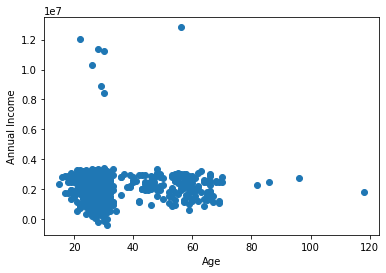
plt.xlabel('Age')

plt.ylabel('Annual Income')

plt.scatter(df['Age'],df['Annual Income'])

Out[20]:

<matplotlib.collections.PathCollection at 0x7f2af9fb96d0>



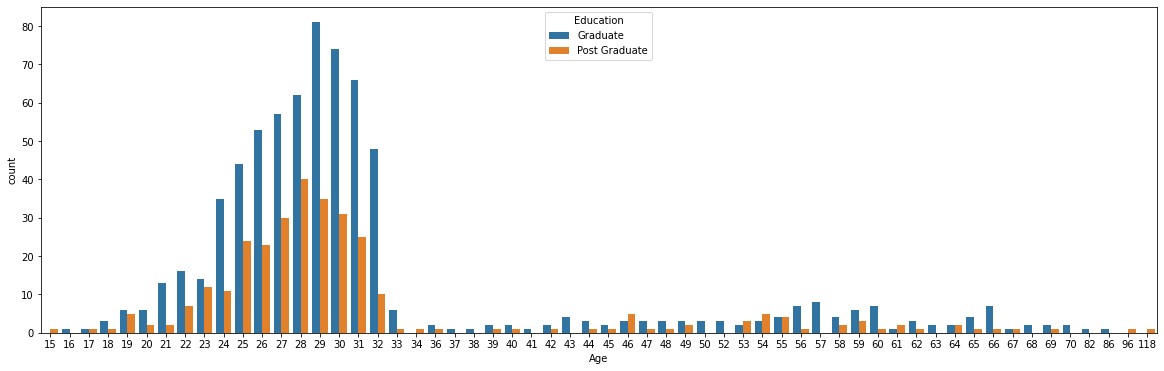
In [21]:

plt.figure(figsize=(20,6))

sns.countplot(x="Age", data=df, hue="Education")

Out[21]:

<AxesSubplot:xlabel='Age', ylabel='count'>



In [22]:

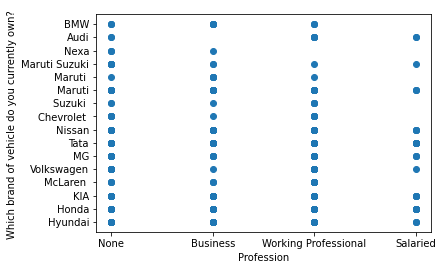
plt.xlabel('Profession')

plt.ylabel('Which brand of vehicle do you currently own? ')

plt.scatter(df['Profession'],df['Which brand of vehicle do you currently own?'])

Out[22]:

<matplotlib.collections.PathCollection at 0x7f2af9fac9d0>



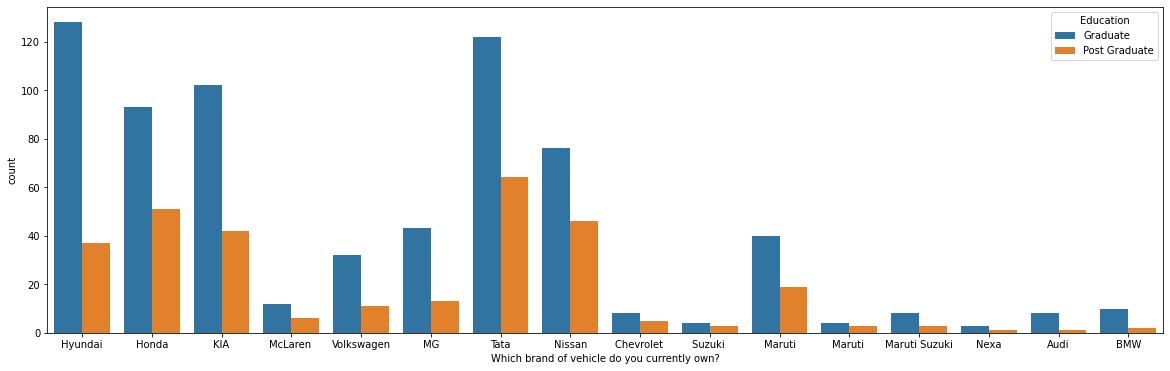
In [23]:

plt.figure(figsize=(20,6))

sns.countplot(x="Which brand of vehicle do you currently own?", data=df, hue="Education")

Out[23]:

<AxesSubplot:xlabel='Which brand of vehicle do you currently own?', ylabel='count'>



In [24]:

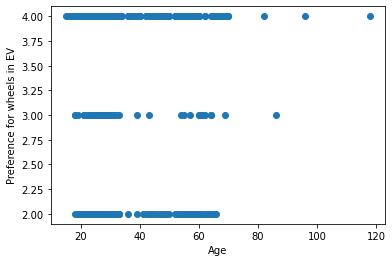
plt.xlabel('Age')

plt.ylabel('Preference for wheels in EV')

plt.scatter(df['Age'],df['Preference for wheels in EV'])

Out[24]:

<matplotlib.collections.PathCollection at 0x7f2af9c9d590>



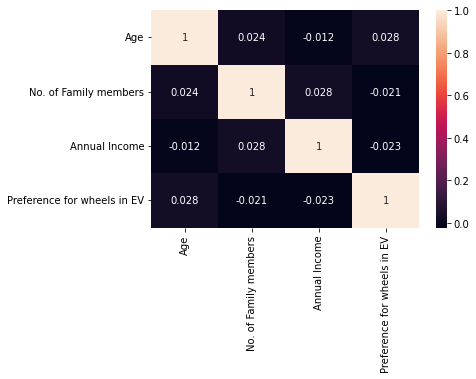
In [25]:

*# Heatmap of Correlation*

sns.heatmap(df.corr(), annot=True)

Out[25]:

<AxesSubplot:>

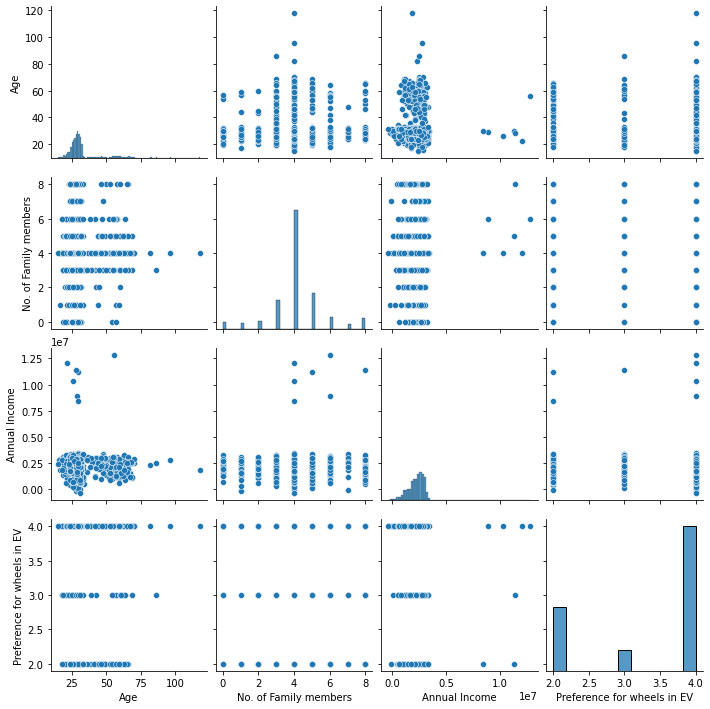


In [26]:

sns.pairplot(df)

Out[26]:

<seaborn.axisgrid.PairGrid at 0x7f2af9bcdb50>



In [27]:

from sklearn.preprocessing import LabelEncoder

def label\_encode(data,column):

label\_encoder=LabelEncoder()

return label\_encoder.fit\_transform(data[column].astype(str))

In [28]:

df.columns

Out[28]:

Index(['Age', 'City', 'Profession', 'Marital Status', 'Education',

'No. of Family members', 'Annual Income',

'Would you prefer replacing all your vehicles to Electronic vehicles?',

'If Yes/Maybe what type of EV would you prefer?',

'Do you think Electronic Vehicles are economical?',

'Which brand of vehicle do you currently own?',

'How much money could you spend on an Electronic vehicle?',

'Preference for wheels in EV',

'Do you think Electronic vehicles will replace fuel cars in India?'],

dtype='object')

In [29]:

df['City']= label\_encode(df,'City')

df['Profession']= label\_encode(df,'Profession')

df['Marital Status']= label\_encode(df,'Marital Status')

df['Education']= label\_encode(df,'Education')

df['Would you prefer replacing all your vehicles to Electronic vehicles?']= label\_encode(df,'Would you prefer replacing all your vehicles to Electronic vehicles?')

df['If Yes/Maybe what type of EV would you prefer?']= label\_encode(df,'If Yes/Maybe what type of EV would you prefer?')

df['Do you think Electronic Vehicles are economical?']= label\_encode(df,'Do you think Electronic Vehicles are economical?')

df['Which brand of vehicle do you currently own?']= label\_encode(df,'Which brand of vehicle do you currently own?')

df['Do you think Electronic vehicles will replace fuel cars in India?']= label\_encode(df,'Do you think Electronic vehicles will replace fuel cars in India?')

df['How much money could you spend on an Electronic vehicle?']= label\_encode(df,'How much money could you spend on an Electronic vehicle?')

In [30]:

df.head()

Out[30]:

|  | Age | City | Profession | Marital Status | Education | No. of Family members | Annual Income | Would you prefer replacing all your vehicles to Electronic vehicles? | If Yes/Maybe what type of EV would you prefer? | Do you think Electronic Vehicles are economical? | Which brand of vehicle do you currently own? | How much money could you spend on an Electronic vehicle? | Preference for wheels in EV | Do you think Electronic vehicles will replace fuel cars in India? |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 30 | 12 | 1 | 1 | 0 | 5 | 1193875.647 | 0 | 3 | 2 | 4 | 6 | 2 | 0 |
| 1 | 27 | 19 | 1 | 1 | 0 | 4 | 1844540.398 | 2 | 3 | 2 | 3 | 4 | 4 | 2 |
| 2 | 32 | 10 | 1 | 1 | 0 | 4 | 2948150.113 | 2 | 1 | 2 | 5 | 4 | 4 | 2 |
| 3 | 55 | 19 | 0 | 1 | 0 | 3 | 2832379.739 | 0 | 1 | 1 | 4 | 6 | 4 | 1 |
| 4 | 26 | 21 | 1 | 1 | 0 | 4 | 2638750.576 | 2 | 4 | 2 | 10 | 4 | 4 | 2 |

In [31]:

from statsmodels.stats.outliers\_influence import variance\_inflation\_factor

In [32]:

y = df['City']

X = df.drop(['City'],axis=1,inplace = True)

In [33]:

def calc\_vif(X):

*# Calculating VIF*

vif = pd.DataFrame()

vif["variables"] = X.columns

vif["VIF"] = [variance\_inflation\_factor(X.values, i) for i **in** range(X.shape[1])]

return(vif)

In [34]:

X = df.iloc[:,:-1]

In [35]:

calc\_vif(X)

Out[35]:

|  | variables | VIF |
| --- | --- | --- |
| 0 | Age | 7.954 |
| 1 | Profession | 2.900 |
| 2 | Marital Status | 2.704 |
| 3 | Education | 1.445 |
| 4 | No. of Family members | 8.199 |
| 5 | Annual Income | 5.669 |
| 6 | Would you prefer replacing all your vehicles t... | 4.127 |
| 7 | If Yes/Maybe what type of EV would you prefer? | 7.080 |
| 8 | Do you think Electronic Vehicles are economical? | 6.238 |
| 9 | Which brand of vehicle do you currently own? | 3.854 |
| 10 | How much money could you spend on an Electroni... | 12.832 |
| 11 | Preference for wheels in EV | 12.317 |

**VIF equal to 1 = variables are not correlated, VIF between 1 and 5 = variables are moderately correlated, VIF greater than 5 = variables are highly correlated**

**K MEANS MODEL**

In [36]:

from sklearn.cluster import KMeans

wcss = []

for k **in** range(1,11):

kmeans = KMeans(n\_clusters=k, init="k-means++",random\_state=28)

kmeans.fit(X)

wcss.append(kmeans.inertia\_)

plt.figure(figsize=(12,6))

plt.grid()

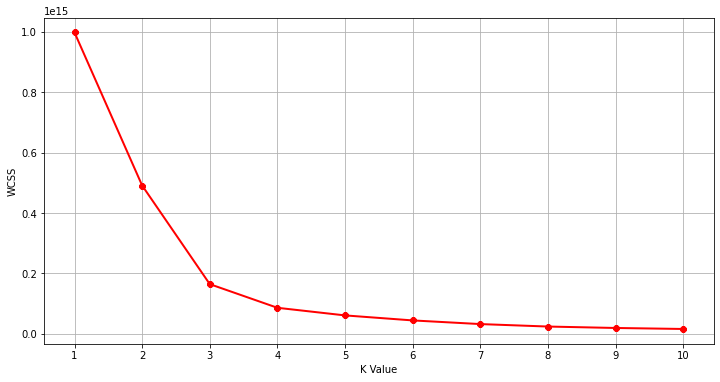
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker ="8")

plt.xlabel("K Value")

plt.xticks(np.arange(1,11,1))

plt.ylabel("WCSS")

plt.show()



In [37]:

km = KMeans(n\_clusters=4, random\_state=28)

clusters = km.fit\_predict(df)

df["Cluster"] = clusters

df\_Original["Cluster"] = clusters

from mpl\_toolkits.mplot3d import Axes3D

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

fig = plt.figure(figsize=(20,10))

ax = fig.add\_subplot(111, projection='3d')

ax.scatter(df.Age[df.Cluster == 0], df["Annual Income"][df.Cluster == 0], df["How much money could you spend on an Electronic vehicle?"][df.Cluster == 0], c='blue', s=60)

ax.scatter(df.Age[df.Cluster == 1], df["Annual Income"][df.Cluster == 1], df["How much money could you spend on an Electronic vehicle?"][df.Cluster == 1], c='red', s=60)

ax.scatter(df.Age[df.Cluster == 2], df["Annual Income"][df.Cluster == 2], df["How much money could you spend on an Electronic vehicle?"][df.Cluster == 2], c='green', s=60)

ax.scatter(df.Age[df.Cluster == 3], df["Annual Income"][df.Cluster == 3], df["How much money could you spend on an Electronic vehicle?"][df.Cluster == 3], c='orange', s=60)

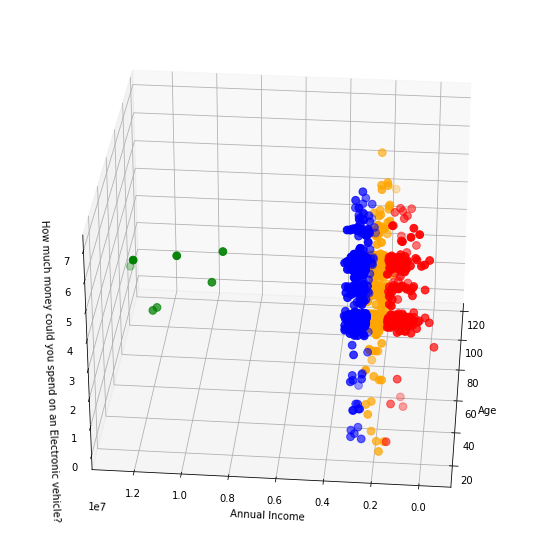
ax.view\_init(30, 185)

plt.xlabel("Age")

plt.ylabel("Annual Income")

ax.set\_zlabel('How much money could you spend on an Electronic vehicle?')

plt.show()



In [38]:

df.head()

Out[38]:

|  | Age | Profession | Marital Status | Education | No. of Family members | Annual Income | Would you prefer replacing all your vehicles to Electronic vehicles? | If Yes/Maybe what type of EV would you prefer? | Do you think Electronic Vehicles are economical? | Which brand of vehicle do you currently own? | How much money could you spend on an Electronic vehicle? | Preference for wheels in EV | Do you think Electronic vehicles will replace fuel cars in India? | Cluster |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 30 | 1 | 1 | 0 | 5 | 1193875.647 | 0 | 3 | 2 | 4 | 6 | 2 | 0 | 1 |
| 1 | 27 | 1 | 1 | 0 | 4 | 1844540.398 | 2 | 3 | 2 | 3 | 4 | 4 | 2 | 3 |
| 2 | 32 | 1 | 1 | 0 | 4 | 2948150.113 | 2 | 1 | 2 | 5 | 4 | 4 | 2 | 0 |
| 3 | 55 | 0 | 1 | 0 | 3 | 2832379.739 | 0 | 1 | 1 | 4 | 6 | 4 | 1 | 0 |
| 4 | 26 | 1 | 1 | 0 | 4 | 2638750.576 | 2 | 4 | 2 | 10 | 4 | 4 | 2 | 0 |

In [39]:

df1=df\_Original.copy()

df1

Out[39]:

|  | Unnamed: 0 | Age | City | Profession | Marital Status | Education | No. of Family members | Annual Income | Would you prefer replacing all your vehicles to Electronic vehicles? | If Yes/Maybe what type of EV would you prefer? | Do you think Electronic Vehicles are economical? | Which brand of vehicle do you currently own? | How much money could you spend on an Electronic vehicle? | Preference for wheels in EV | Do you think Electronic vehicles will replace fuel cars in India? | Cluster |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 30 | Nabha | None | Single | Graduate | 5 | 1193875.647 | Maybe | SUV | Yes | Hyundai | <5 lakhs | 2 | I don't think so | 1 |
| 1 | 1 | 27 | Pune | None | Single | Graduate | 4 | 1844540.398 | Yes | SUV | Yes | Honda | <15 lakhs | 4 | Yes, in <20years | 3 |
| 2 | 2 | 32 | Kashipur | None | Single | Graduate | 4 | 2948150.113 | Yes | Hatchback | Yes | KIA | <15 lakhs | 4 | Yes, in <20years | 0 |
| 3 | 3 | 55 | Pune | Business | Single | Graduate | 3 | 2832379.739 | Maybe | Hatchback | No | Hyundai | <5 lakhs | 4 | Yes, in <10 years | 0 |
| 4 | 4 | 26 | Satara | None | Single | Graduate | 4 | 2638750.576 | Yes | Sedan | Yes | McLaren | <15 lakhs | 4 | Yes, in <20years | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 995 | 995 | 31 | Pune | None | Married | Graduate | 7 | 2110722.120 | Yes | SUV | Yes | KIA | <25 lakhs | 4 | Yes, in <10 years | 3 |
| 996 | 996 | 29 | Pune | None | Married | Post Graduate | 4 | 1616287.706 | No | SUV | Yes | KIA | <5 lakhs | 4 | Yes, in <20years | 3 |
| 997 | 997 | 30 | Mumbai | Business | Single | Graduate | 4 | 2202829.029 | Yes | SUV | Yes | Honda | <15 lakhs | 4 | Yes, in <20years | 3 |
| 998 | 998 | 24 | Ahmedabad | None | Married | Graduate | 4 | 1764744.068 | Yes | SUV | Yes | Maruti | <15 lakhs | 4 | Yes, in <20years | 3 |
| 999 | 999 | 30 | Pune | Business | Single | Graduate | 4 | 2486664.468 | No | Liftback | Yes | Maruti | <5 lakhs | 4 | Yes, in <10 years | 0 |

1000 rows × 16 columns

In [40]:

df1["City"] = df1["City"].replace({"Pune":"Pune", "pUNE": "Pune", "pune": "Pune", "Pune ": "Pune"})

df1["City"] = df1["City"].replace({"Mumbai ":"Mumbai", "Mumbai": "Mumbai"})

df1["City"] = df1["City"].replace({"Banglore ":"Bengaluru"})

df1["City"] = df1["City"].replace({"Delhi":"New Delhi", "Delhi ": "New Delhi", "New Delhi ": "New Delhi"})

df1["City"] = df1["City"].replace({"Hakdwani":"Haldwani", "Haldwani ": "Haldwani"})

df1["City"] = df1["City"].replace({"nashik":"Nashik"})

In [41]:

df1['Cluster'].value\_counts()

Out[41]:

0 430

3 368

1 195

2 7

Name: Cluster, dtype: int64

More customers belong to **Cluster** **0 & 3**

Cluster 2 & 1 has very less customers

In [42]:

*# Segregrating each cluster*

Cluster\_0 = df1[df1.Cluster==0]

Cluster\_1 = df1[df1.Cluster==1]

Cluster\_2 = df1[df1.Cluster==2]

Cluster\_3 = df1[df1.Cluster==3]

Cluster\_2.shape

Out[42]:

(7, 16)

In [43]:

[Cluster\_0['Age'].value\_counts().head(3),

Cluster\_1['Age'].value\_counts().head(3),

Cluster\_2['Age'].value\_counts().head(3),

Cluster\_3['Age'].value\_counts().head(3)]

Out[43]:

[29 47

28 43

30 43

Name: Age, dtype: int64,

31 28

30 23

29 23

Name: Age, dtype: int64,

30 2

26 1

29 1

Name: Age, dtype: int64,

29 45

28 42

30 37

Name: Age, dtype: int64]

Based on Cluster 0 & 3, Age group of **28-31** are to be targeted

In [44]:

[Cluster\_0['City'].value\_counts().head(),

Cluster\_1['City'].value\_counts().head(),

Cluster\_2['City'].value\_counts().head(),

Cluster\_3['City'].value\_counts().head()]

Out[44]:

[Pune 279

New Delhi 28

Mumbai 23

Bengaluru 12

Haldwani 10

Name: City, dtype: int64,

Pune 107

Mumbai 14

New Delhi 11

Haldwani 6

Chennai 6

Name: City, dtype: int64,

Pune 4

Mumbai 2

Ahmedabad 1

Name: City, dtype: int64,

Pune 231

New Delhi 18

Mumbai 17

Haldwani 11

Satara 9

Name: City, dtype: int64]

In [45]:

*#Pune & Mumbai are present in all the clusters,*

*#New Delhi & Haldwani are present in 3 of 4 clusters,*

*#Satara & Bengaluru are present in 1 of the 4 clusters.*

1. **'Pune' & 'Mumbai'** are the most suitable location to create the early market in EV segment.

2. **'New Delhi' & 'Haldwani'** should be next priority.

3. Followed by **'Satara' & 'Bengaluru'**.

***The City should also have enough charging stations, will try to analyse Sanctioned Charging stations in India Dataset.***

In [46]:

[Cluster\_0['Profession'].value\_counts().head(),

Cluster\_1['Profession'].value\_counts().head(),

Cluster\_2['Profession'].value\_counts().head(),

Cluster\_3['Profession'].value\_counts().head()]

Out[46]:

[None 207

Working Professional 135

Business 73

Salaried 15

Name: Profession, dtype: int64,

None 87

Working Professional 68

Business 32

Salaried 8

Name: Profession, dtype: int64,

None 5

Working Professional 1

Business 1

Name: Profession, dtype: int64,

None 161

Working Professional 132

Business 61

Salaried 14

Name: Profession, dtype: int64]

**Salaried people are very less interested in EV vehicles.**

In [47]:

[Cluster\_0['Annual Income'].mean(),

Cluster\_1['Annual Income'].mean(),

Cluster\_2['Annual Income'].mean(),

Cluster\_3['Annual Income'].mean()]

Out[47]:

[2812149.7606071676, 1097632.5625596757, 10737228.104106281, 2064995.870350378]

**Cluster 0** customers have **Higher Average Income(Rs. 2812149.76)**, Cluster 1 & 2 have lower Average Income

In [48]:

[Cluster\_0['Marital Status'].value\_counts().head(),

Cluster\_1['Marital Status'].value\_counts().head(),

Cluster\_2['Marital Status'].value\_counts().head(),

Cluster\_3['Marital Status'].value\_counts().head()]

Out[48]:

[Single 283

Married 147

Name: Marital Status, dtype: int64,

Single 133

Married 62

Name: Marital Status, dtype: int64,

Married 5

Single 2

Name: Marital Status, dtype: int64,

Single 220

Married 148

Name: Marital Status, dtype: int64]

In [49]:

m=Cluster\_3[Cluster\_3['Marital Status']=='Single']

m['Would you prefer replacing all your vehicles to Electronic vehicles?'].value\_counts()

Out[49]:

Yes 147

Maybe 40

No 33

Name: Would you prefer replacing all your vehicles to Electronic vehicles?, dtype: int64

**Single** Marital Status people should be Targeted

In [50]:

[Cluster\_0['No. of Family members'].value\_counts().head(),

Cluster\_1['No. of Family members'].value\_counts().head(),

Cluster\_2['No. of Family members'].value\_counts().head(),

Cluster\_3['No. of Family members'].value\_counts().head()]

Out[50]:

[4 231

5 63

3 59

6 22

2 14

Name: No. of Family members, dtype: int64,

4 88

5 37

3 22

6 14

8 12

Name: No. of Family members, dtype: int64,

4 3

6 2

5 1

8 1

Name: No. of Family members, dtype: int64,

4 194

5 57

3 46

8 22

6 12

Name: No. of Family members, dtype: int64]

Family with **3-5** members are more interested in EV.

In [51]:

[Cluster\_0['Which brand of vehicle do you currently own?'].value\_counts().head(),

Cluster\_1['Which brand of vehicle do you currently own?'].value\_counts().head(),

Cluster\_2['Which brand of vehicle do you currently own?'].value\_counts().head(),

Cluster\_3['Which brand of vehicle do you currently own?'].value\_counts().head()]

Out[51]:

[Tata 78

Hyundai 70

Honda 65

KIA 61

Nissan 53

Name: Which brand of vehicle do you currently own?, dtype: int64,

Tata 36

Hyundai 33

KIA 28

Honda 26

Nissan 22

Name: Which brand of vehicle do you currently own?, dtype: int64,

Tata 3

KIA 1

MG 1

Honda 1

Hyundai 1

Name: Which brand of vehicle do you currently own?, dtype: int64,

Tata 69

Hyundai 61

KIA 54

Honda 52

Nissan 47

Name: Which brand of vehicle do you currently own?, dtype: int64]

**Tata,Hyundai,Honda,KIA are the Brands that Customers own most.**

*After having discussions with Automobile Domain Experts. Below is the details about what each brand is well known for,*

*Tata - Robust, Safety, Affordable*

*Hyundai - Unique design, Comfort*

*Honda - Durability, Performance*

*KIA - Innovative Features, efficiency.*

In [52]:

[Cluster\_0['Education'].value\_counts().head(),

Cluster\_1['Education'].value\_counts().head(),

Cluster\_2['Education'].value\_counts().head(),

Cluster\_3['Education'].value\_counts().head()]

Out[52]:

[Graduate 298

Post Graduate 132

Name: Education, dtype: int64,

Graduate 130

Post Graduate 65

Name: Education, dtype: int64,

Graduate 5

Post Graduate 2

Name: Education, dtype: int64,

Graduate 260

Post Graduate 108

Name: Education, dtype: int64]

In [53]:

e=Cluster\_3[Cluster\_3['Education']=='Graduate']

e['Would you prefer replacing all your vehicles to Electronic vehicles?'].value\_counts()

Out[53]:

Yes 176

Maybe 50

No 34

Name: Would you prefer replacing all your vehicles to Electronic vehicles?, dtype: int64

In [54]:

y= df\_Original['Would you prefer replacing all your vehicles to Electronic vehicles?']

z=df\_Original['Cluster']

x = list(zip(y, z))

i = pd.DataFrame(x,columns=['Would you prefer replacing all your vehicles to Electronic vehicles?', 'Cluster'])

(i['Would you prefer replacing all your vehicles to Electronic vehicles?']=='Yes').sum() + (i['Would you prefer replacing all your vehicles to Electronic vehicles?']=='Maybe').sum()

Out[54]:

864

**864 customers out of 1000 are intersted in EV.**

**Graduates** are more interested in replacing vehicles to EV

In [55]:

print(Cluster\_0['If Yes/Maybe what type of EV would you prefer?'].value\_counts().head(3))

print(Cluster\_1['If Yes/Maybe what type of EV would you prefer?'].value\_counts().head(3))

print(Cluster\_2['If Yes/Maybe what type of EV would you prefer?'].value\_counts().head(3))

print(Cluster\_3['If Yes/Maybe what type of EV would you prefer?'].value\_counts().head(3))

SUV 191

Sedan 115

Hatchback 62

Name: If Yes/Maybe what type of EV would you prefer?, dtype: int64

SUV 97

Sedan 39

Liftback 30

Name: If Yes/Maybe what type of EV would you prefer?, dtype: int64

SUV 3

Sedan 2

Hatchback 1

Name: If Yes/Maybe what type of EV would you prefer?, dtype: int64

SUV 176

Sedan 84

Hatchback 55

Name: If Yes/Maybe what type of EV would you prefer?, dtype: int64

People are more interested in buying **SUV**'s, followed by **Sedan** and **Hatchback** in EV Segments

In [56]:

print(Cluster\_0['How much money could you spend on an Electronic vehicle?'].value\_counts().head())

print(Cluster\_1['How much money could you spend on an Electronic vehicle?'].value\_counts().head())

print(Cluster\_2['How much money could you spend on an Electronic vehicle?'].value\_counts().head())

print(Cluster\_3['How much money could you spend on an Electronic vehicle?'].value\_counts().head())

<15 lakhs 216

<5 lakhs 127

<25 lakhs 49

>25 lakhs 15

1200000 7

Name: How much money could you spend on an Electronic vehicle?, dtype: int64

<15 lakhs 102

<5 lakhs 53

<25 lakhs 29

>25 lakhs 5

1200000 3

Name: How much money could you spend on an Electronic vehicle?, dtype: int64

<15 lakhs 3

<5 lakhs 3

<25 lakhs 1

Name: How much money could you spend on an Electronic vehicle?, dtype: int64

<15 lakhs 162

<5 lakhs 105

<25 lakhs 69

>25 lakhs 14

700000 6

Name: How much money could you spend on an Electronic vehicle?, dtype: int64

In [57]:

*#Cluster 0 & 3 have more number of customers(770). Condidering only 0 & 3 clusters,*

*#378 out of 770 that is 49.09% of people only wants to spend <15 lakhs for EV.*

*#232 out of 770 that is 30.12% of people only wants to spend <5 lakhs.*

*#118 out of 770 that is 15.32% of people only wants to spend <25 lakhs.*

* **49.09% of people only wants to spend <15 lakhs for EV**
* **Only 5.47% of people willing to spend more than 25 lakhs.**

By this,we can infer most customers want to invest **less than 15 lakhs** for their EV.

Considering SUV, Sedan and Hatchback as most preferred vehicle type.

Interestingly, Considering the money Customers can spend on EV and manufacturing cost for the company wrt. different EV types.

Company should focus on making '**Sedan**' in price range **less than 15 lakhs**.

***If company wants to make SUV's, only 15.32% of customers are willing to pay more than 15 lakhs.***

In [ ]:

Analysis on Active Vehicles as on 08.12.2021 in India

In [58]:

Active\_EVs = pd.read\_csv('/kaggle/input/ev-data/Active\_EVs.csv')

Active\_EVs.head()

Out[58]:

|  | Sl. No. | State/UT | Total Number of active Vehicle as on 08.12.2021 | Total number of active Electric Vehicles as on 08.12.2021 |
| --- | --- | --- | --- | --- |
| 0 | 1 | Andaman and Nicobar Island | 143529 | 157 |
| 1 | 2 | Arunachal Pradesh | 235189 | 20 |
| 2 | 3 | Assam | 4445227 | 43707 |
| 3 | 4 | Bihar | 9816784 | 58655 |
| 4 | 5 | Chandigarh | 720272 | 1791 |

In [59]:

Active\_EVs.isnull().sum()

Out[59]:

Sl. No. 0

State/UT 0

Total Number of active Vehicle as on 08.12.2021 0

Total number of active Electric Vehicles as on 08.12.2021 0

dtype: int64

In [60]:

plt.rcParams['figure.figsize'] = (20, 10)

f = sns.barplot(y=Active\_EVs['State/UT'], x=Active\_EVs['Total Number of active Vehicle as on 08.12.2021'])

plt.title('State wise active vehicle')

plt.show()

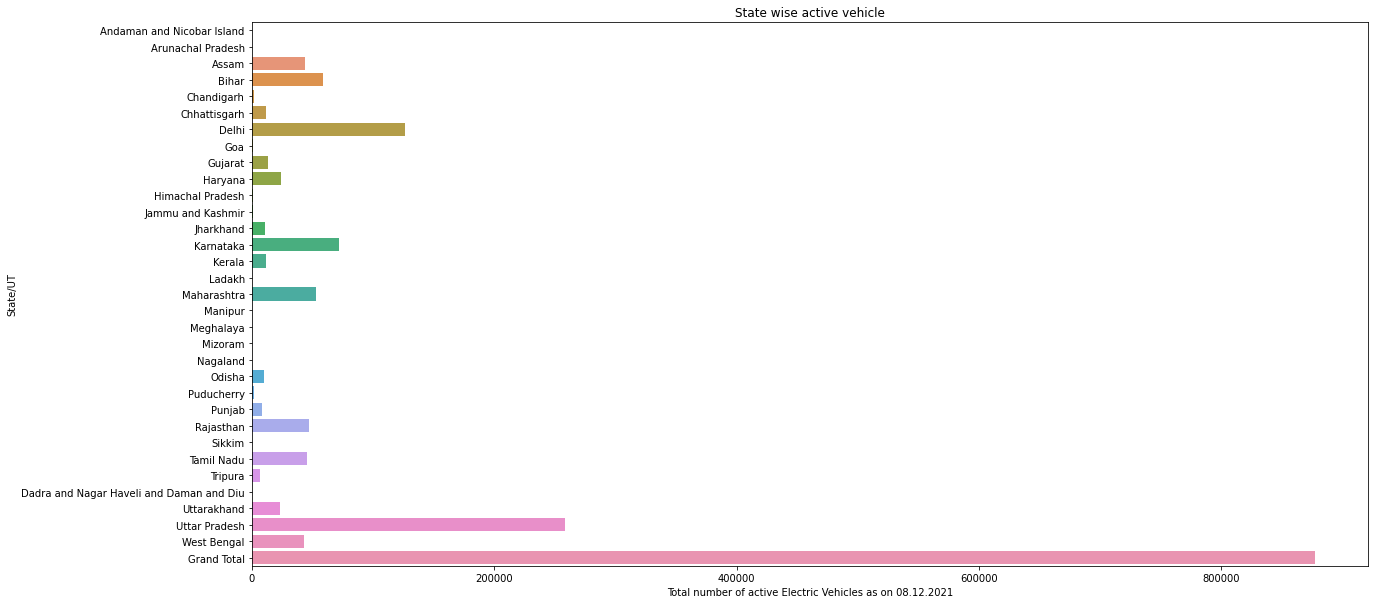
plt.rcParams['figure.figsize'] = (20, 10)

f = sns.barplot(y=Active\_EVs['State/UT'], x=Active\_EVs['Total number of active Electric Vehicles as on 08.12.2021'])

plt.title('State wise active vehicle')

plt.show()





In [61]:

sns.lmplot(data=Active\_EVs, y="Total number of active Electric Vehicles as on 08.12.2021", x="Total Number of active Vehicle as on 08.12.2021", hue='State/UT')

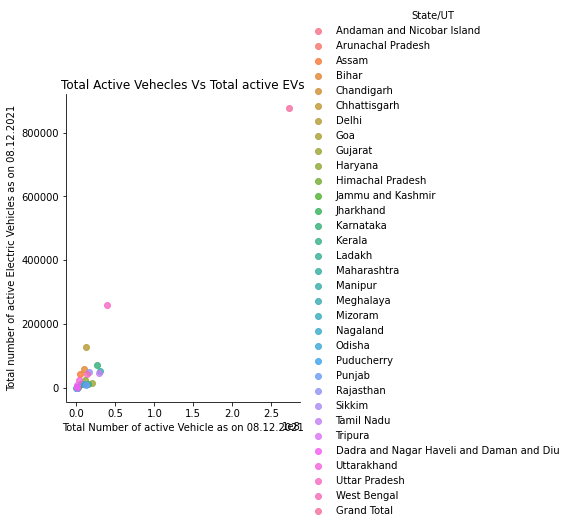
plt.title('Total Active Vehecles Vs Total active EVs')

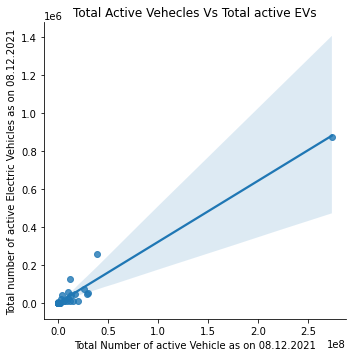
sns.lmplot(data=Active\_EVs, y="Total number of active Electric Vehicles as on 08.12.2021", x="Total Number of active Vehicle as on 08.12.2021")

plt.title('Total Active Vehecles Vs Total active EVs')

Out[61]:

Text(0.5, 1.0, 'Total Active Vehecles Vs Total active EVs')





Analysis on Statewise charging station

In [62]:

Statewise\_CS = pd.read\_excel('/kaggle/input/ev-data/Charging\_station\_statewise.xlsx')

Statewise\_CS.head()

Out[62]:

|  | State/UT | No. of EV Charging station Sanctioned |
| --- | --- | --- |
| 0 | Maharashtra | 405 |
| 1 | Andhra Pradesh | 331 |
| 2 | Tamil Nadu | 281 |
| 3 | Gujarat | 365 |
| 4 | Uttar Pradesh | 207 |

In [63]:

Statewise\_CS.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 33 entries, 0 to 32

Data columns (total 2 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 State/UT 33 non-null object

1 No. of EV Charging station Sanctioned 33 non-null int64

dtypes: int64(1), object(1)

memory usage: 656.0+ bytes

In [64]:

*# Figure Size*

fig, ax = plt.subplots(figsize =(14, 10))

*# Horizontal Bar Plot*

ax.barh(Statewise\_CS['State/UT'], Statewise\_CS['No. of EV Charging station Sanctioned'],color ='indigo')

*# Remove axes splines*

for s **in** ['top', 'bottom', 'left', 'right']:

ax.spines[s].set\_visible(False)

*# Remove x, y Ticks*

ax.xaxis.set\_ticks\_position('none')

ax.yaxis.set\_ticks\_position('none')

*# Add padding between axes and labels*

ax.xaxis.set\_tick\_params(pad = 5)

ax.yaxis.set\_tick\_params(pad = 10)

*# Add x, y gridlines*

ax.grid(b = True, color ='grey',

linestyle ='-.', linewidth = 0.5,

alpha = 0.2)

*# Show top values*

ax.invert\_yaxis()

*# Add annotation to bars*

for i **in** ax.patches:

plt.text(i.get\_width()+0.2, i.get\_y()+0.5,

str(round((i.get\_width()), 2)),

fontsize = 12, fontweight ='light',

color ='black')

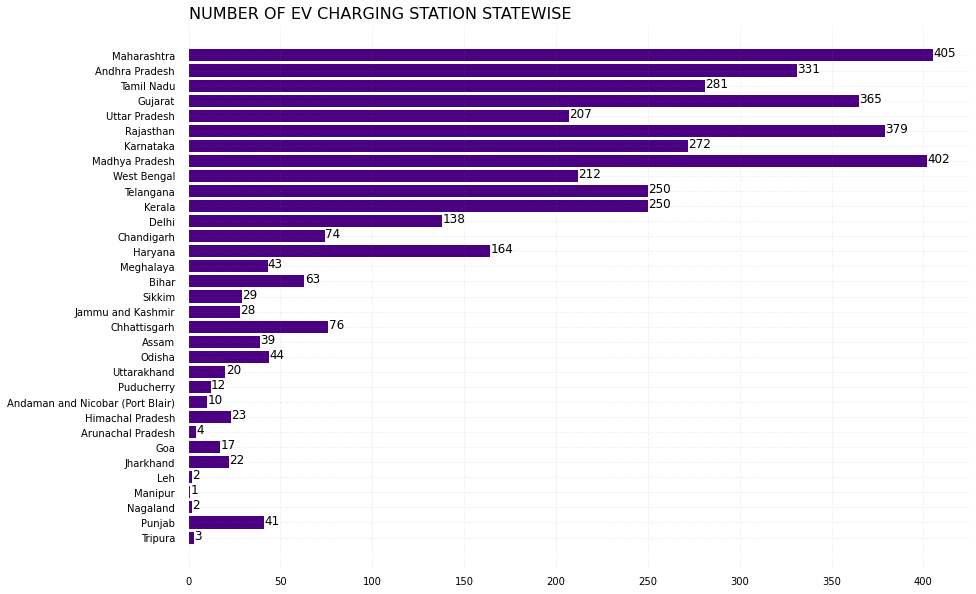
*# Add Plot Title*

ax.set\_title('NUMBER OF EV CHARGING STATION STATEWISE',

loc ='left',fontsize = 16)

*# Show Plot*

plt.show()



Analysis on Citywise charging station

In [65]:

Citywise\_CS= pd.read\_excel("/kaggle/input/ev-data/Charging\_station\_citywise.xlsx")

Citywise\_CS.tail()

Out[65]:

|  | Sl. No. | City | Charging Stations |
| --- | --- | --- | --- |
| 11 | 12.000 | Goa | 30 |
| 12 | 13.000 | Hyderabad | 57 |
| 13 | 14.000 | Agra | 15 |
| 14 | 15.000 | Shimla | 9 |
| 15 | NaN | Total | 710 |

In [66]:

Citywise\_CS.drop(Citywise\_CS.index[15], inplace= True)

Citywise\_CS.tail()

Out[66]:

|  | Sl. No. | City | Charging Stations |
| --- | --- | --- | --- |
| 10 | 11.000 | Lucknow | 1 |
| 11 | 12.000 | Goa | 30 |
| 12 | 13.000 | Hyderabad | 57 |
| 13 | 14.000 | Agra | 15 |
| 14 | 15.000 | Shimla | 9 |

In [67]:

*# Figure Size*

fig, ax = plt.subplots(figsize =(16, 12))

*# Bar Plot*

ax.bar(Citywise\_CS['City'], Citywise\_CS['Charging Stations'], color ='lightcoral')

*# Remove axes splines*

for s **in** ['top', 'bottom', 'left', 'right']:

ax.spines[s].set\_visible(False)

*# Remove x, y Ticks*

ax.xaxis.set\_ticks\_position('none')

ax.yaxis.set\_ticks\_position('none')

*# Add padding between axes and labels*

ax.xaxis.set\_tick\_params(pad = 5)

ax.yaxis.set\_tick\_params(pad = 10)

*# Add x, y gridlines*

ax.grid(b = True, color ='grey',

linestyle ='-.', linewidth = 0.5,

alpha = 0.2)

*# Show top values*

*#ax.invert\_yaxis()*

*# Add annotation to bars*

for bar **in** ax.patches:

ax.annotate(format(bar.get\_height()),

(bar.get\_x() + bar.get\_width() / 2,

bar.get\_height()), ha='center', va='center',

size=15, xytext=(0, 8),

textcoords='offset points')

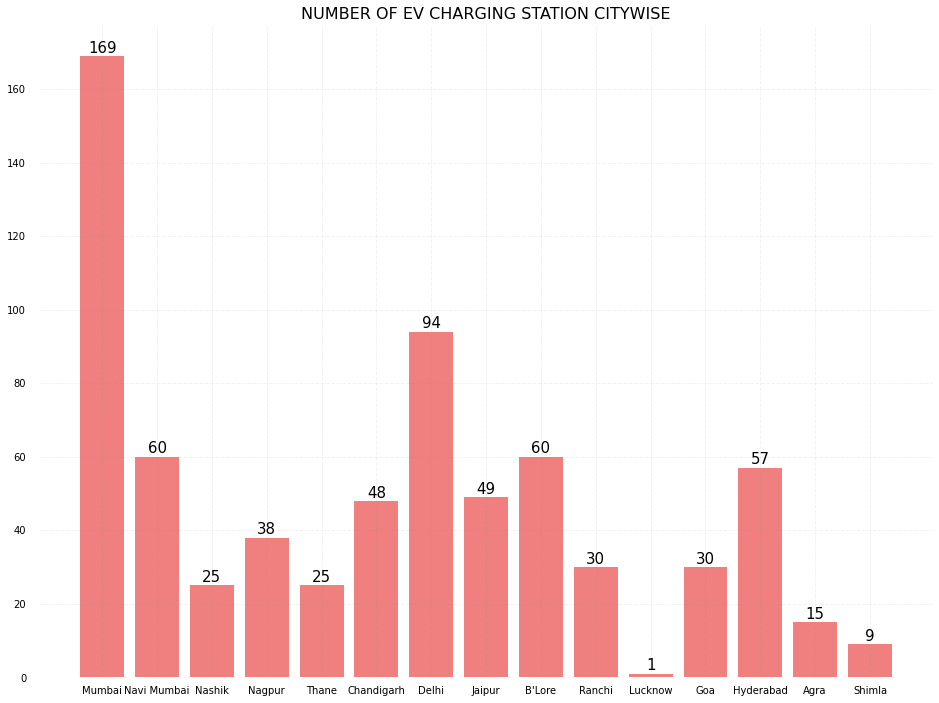
*# Add Plot Title*

ax.set\_title('NUMBER OF EV CHARGING STATION CITYWISE',

loc ='center',fontsize = 16)

*# Show Plot*

plt.show()



Analysis on Highway wise charging station

In [68]:

Highwise\_CS= pd.read\_csv('/kaggle/input/ev-data/Highwise\_charging\_station.csv',encoding='cp1252')

Highwise\_CS.tail()

Out[68]:

|  | Sl. No | Category | Expressways/Highways | EV Charging Stations Sanctioned |
| --- | --- | --- | --- | --- |
| 21 | 13 | Highways | Chennai - Trivendram | 74 |
| 22 | 14 | Highways | Chennai-Ballary | 62 |
| 23 | 15 | Highways | Chennai - Nagpur | 114 |
| 24 | 16 | Highways | Mangaldai - Wakro | 64 |
| 25 | Total | Total | Total | 1576 |

In [69]:

Highwise\_CS.drop(Highwise\_CS.index[25], inplace= True)

Highwise\_CS.tail()

Out[69]:

|  | Sl. No | Category | Expressways/Highways | EV Charging Stations Sanctioned |
| --- | --- | --- | --- | --- |
| 20 | 12 | Highways | Chennai-Bhubaneswar | 120 |
| 21 | 13 | Highways | Chennai - Trivendram | 74 |
| 22 | 14 | Highways | Chennai-Ballary | 62 |
| 23 | 15 | Highways | Chennai - Nagpur | 114 |
| 24 | 16 | Highways | Mangaldai - Wakro | 64 |

In [70]:

Highwise\_CS.isnull().sum()

Out[70]:

Sl. No 0

Category 0

Expressways/Highways 0

EV Charging Stations Sanctioned 0

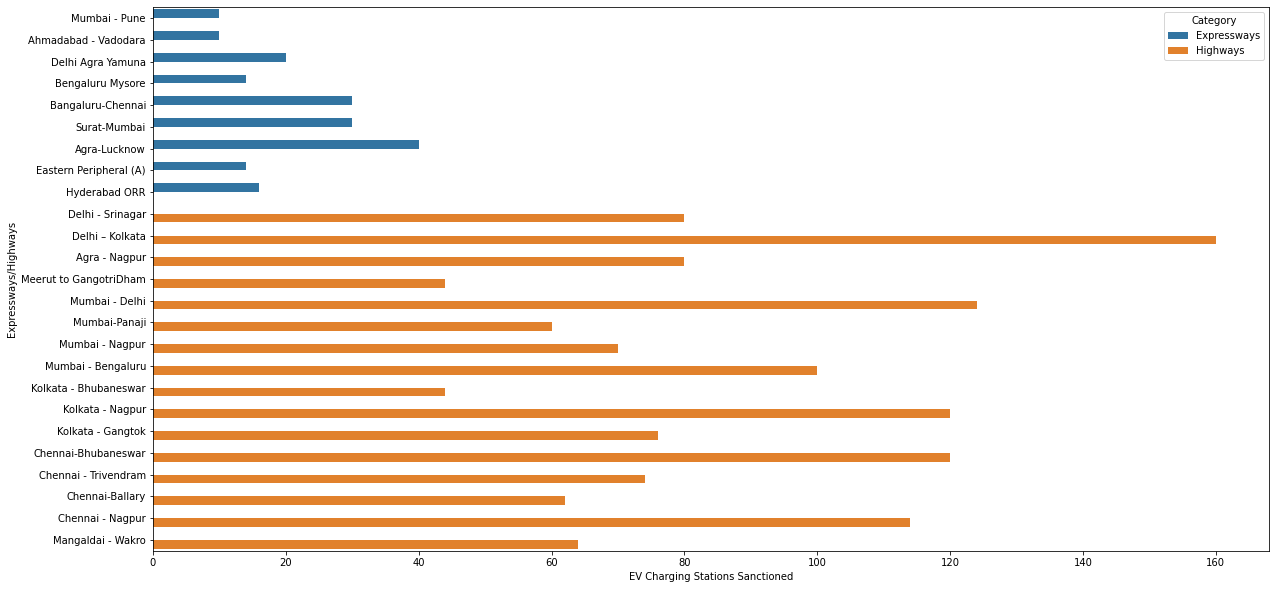
dtype: int64

In [71]:

sns.barplot(x=Highwise\_CS["EV Charging Stations Sanctioned"], y=Highwise\_CS['Expressways/Highways'] , hue=Highwise\_CS["Category"])

Out[71]:

<AxesSubplot:xlabel='EV Charging Stations Sanctioned', ylabel='Expressways/Highways'>



**Demographic segments:**

-Age

**Target Age group of 28-31**

-Income

**Target who earn Rs. 20,64,995 to Rs. 28,12,149 per year.**

-Education

**Target Graduates.**

**Geographic segments:**

Location

**Pune, Mumbai, New Delhi, Bengaluru Considering based on Segmentation Analysis and the Infrastructres sanctioned by Govt.**

**Psychographic segments:**

-Lifestyle

**Target Single Marital Status, Family with 3-5 members**

**CONCLUSION:**

From the problem statement I concluded that the above are the target parameters to attain the customers in the EV market segment.