**Data Analysis Project-1**

Dataset Overview

This dataset contains details of the used cars in Germany which are on sale.

The dataset is not clear and hence a lot of data cleaning should be carried out

Dataset link:

<https://drive.google.com/file/d/1NFpNI-AiSLmta139rYaWrNy8xZ0VrbM9/view?usp=sharing>

Analysis 1:

**1. Perform general Data analysis**

Import all the libraries:

Import numpy as np

Import pandas as pd

Import seaborn as sns

Load the dataset

* data=pd.read\_csv(r"C:\Users\user\Downloads\autos.csv",sep=",",encoding="latin-1")
* For finding duplicates as data.duplicates(),For unwanted columns I used the code for individual column as

Data[colmn name].value\_counts()

* And I have done the basic analysis like removing duplicates as

Data.drop duplicate(inplace=True)

and to remove unwanted coulmns and changing the data types of columns on requirement.

* I have used the code as
* data.drop([“nrOfpictures”],axis=1,inplace=True)
* data.astype({“price”:int64,”kilometre”:float})
* data["dateCrawled"]=pd.to\_datetime(data["dateCrawled"])
* data["yearOfRegistration"]=pd.to\_datetime(data["yearOfRegistration "])
* data["dateCreated"]=pd.to\_datetime(data["dateCreated"])
* data["lastSeen"]=pd.to\_datetime(data["lastSeen"])
* Finally I have assigned the nan value with its object as data.fillna({“notrepairedDamage”:”no name”,”fuelType”:”no gearbox”,”model”:”no model”},inplace=True)
* Finally I have reviewed the data to check whether it is clear or not by using the code as data.info()

So the basic data analysis is done.

**2. Can you tell me the Distribution of Vehicles based on Year of Registration with the help of a plot.**

* using matplotlib and seaborn to create a histogram of the distribution of vehicles by year of registration.
* Using the code:
* datac["yearOfRegistration"] = datac["yearOfRegistration"].where((datac["yearOfRegistration"] >= 1950) & (datac["yearOfRegistration"] <= 2023),datac["yearOfRegistration"].mode()[0])

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

sns.histplot(data=datac, x=datac["yearOfRegistration"])

plt.title("Distribution of Vehicles by Year of Registration")

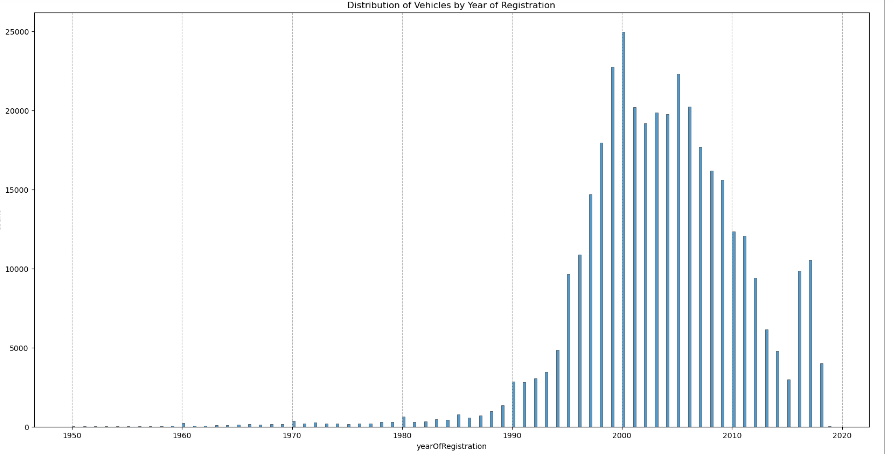
plt.ylabel("count")

plt.xlabel("yearOfRegistration")

plt.grid(axis='x', linestyle='--')

plt.show()

* The figsize parameter sets the size of the figure, and sns.histplot() plots the histogram. The title(), xlabel(), and ylabel() functions are used to add labels to the plot. The grid() function adds gridlines to the x-axis.
* The `plt.figure(figsize=(20, 10))` line sets the size of the figure to be 20 units wide and 10 units . Then, `sns.histplot(data=datac, x=datac["yearOfRegistration"])` creates the histogram plot using the "yearOfRegistration" data from the "datac" dataset.
* The `plt.title()`, `plt.ylabel()`, and `plt.xlabel()` functions set the title, y-axis label, and x-axis label of the plot, respectively. Finally, `plt.grid(axis='x', linestyle='--')` adds gridlines to the x-axis with dashed lines, and `plt.show()` displays the plot.



**3. Create a plot based on the Variation of the price range by the vehicle type.**

* I am using seaborn's barplot function to create a bar plot.
* Using the code :
* sns.barplot(datac,x=datac["vehicleType"],y=datac["price"])

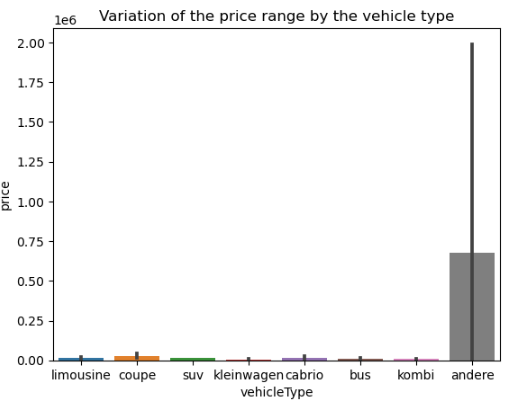
plt.title("Variation of the price range by the vehicle type")

plt.xlabel("vehicleType")

plt.ylabel("price")

plt.show()

* The `x` parameter specifies the data to be plotted on the x-axis, which is the "vehicleType" column in the "datac" dataset. The `y` parameter specifies the data to be plotted on the y-axis, which is the "price" column in the "datac" dataset. The `plt.title()`, `plt.xlabel()`, and `plt.ylabel()` functions set the title, x-axis label, and y-axis label of the plot, respectively. The `plt.show()` function displays the plot. This visualization will show the variation of price across different vehicle types.



**4. Find out Total count of vehicles by type available on ebay for sale.As well as create a visualization for the client.**

* Usingthe code :
* datac["vehicleType"].value\_counts()

sns.countplot(x=datac["vehicleType"])

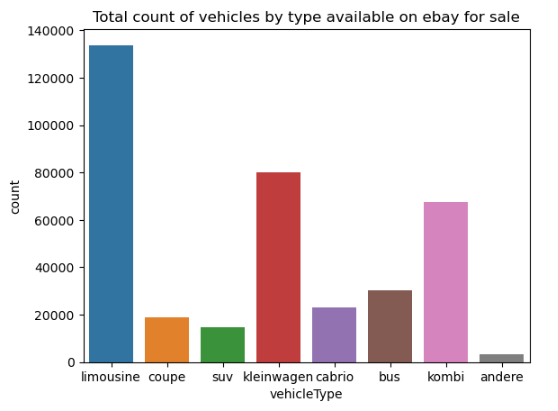
plt.title("Total count of vehicles by type available on ebay for sale")

plt.xlabel("vehicleType")

plt.ylabel("count")

plt.show()

* datac["vehicleType"].value\_counts()` is used to get the count of each vehicle type in the "datac" dataset. This will give you the total number of vehicles for each type.
* The `sns.countplot()` function is then used to create a bar chart based on the counts. The x-axis is set to `datac["vehicleType"]`, and `plt.title()`, `plt.xlabel()`, and `plt.ylabel()` functions are used to set the title, x-axis label, and y-axis label of the plot respectively. Finally, `plt.show()` is used to display the plot.
* This plot visualizes the total count of vehicles available for sale on eBay by their type, giving you an idea of the distribution of vehicle types in the dataset.



**5. Is there any relationship between dollar\_price and kilometer? (Explain with appropriate analysis)**

* I have used this code **:**
* datac[["price","kilometer"]].corr(method="spearman")

sns.heatmap(datac[["price","kilometer"]].corr(method="spearman"),annot=True)

plt.title("relationship between dollar\_price and kilometer")

plt.show()

* datac[["price","kilometer"]].corr(method="spearman")` is used to calculate the Spearman's rank correlation coefficient between the "price" and "kilometer" columns in the "datac" dataset.
* To visualize this correlation, I have created a heatmap using seaborn's `heatmap()` function. The heatmap represents the correlation values between "price" and "kilometer" from the "datac" dataset. The `annot=True` parameter adds the correlation values to the heatmap.
* Next, the title of the plot with `plt.title()` and displayed the plot using `plt.show()`. This visualization helps to understand the relationship between the dollar price and the kilometer of the vehicles in the dataset.

