NAME : Mehul Uttam
PRN : 1032222936
TY CSE AIDS - A (A1 Batch)

ML-1: Lab Assignment No. 01

Problem Statement: Implement various pre-processing techniques on a given dataset.

Objectives:

1. To learn python programming with different modules/libraries.
2. understand the concept of exploratory data analysis.

Theory:

Data Preprocessing:

1. **Data Quality**

Data quality refers to the condition of the data with respect to factors such as accuracy, consistency, completeness, reliability, and relevance. Ensuring high data quality is crucial for effective data analysis and decision-making. Poor data quality can lead to incorrect analysis and faulty predictions in machine learning models.

- Key Aspects:
    - **Accuracy**: Correctness of the data.
    - **Completeness**: No missing or incomplete data.
    - **Consistency**: Data should not have contradictions.
    - **Timeliness**: Data should be up-to-date.
    - **Relevance**: Data must serve the analysis purpose.

2. **Major Tasks in Data Preprocessing** :
   Data preprocessing involves preparing raw data for analysis by applying various techniques to clean, integrate, reduce, transform, and discretize the data.

Data Cleaning: Data cleaning involves detecting and correcting (or removing) inaccurate records from a dataset. This task ensures that the data is reliable and error-free for analysis.

- **Common Techniques**:
    - **Handling Missing Values**: Filling missing values using methods like mean, median, or interpolation, or deleting rows/columns with missing data.
    - **Removing Outliers**: Identifying and removing outliers using statistical

techniques like the Z-score or IQR.
    - **Noise Removal**: Using smoothing techniques to remove noise from the data, such as binning or clustering.
    - **Resolving Data Inconsistencies**: Standardizing formats, handling duplicates.

Data Integration : Data integration refers to combining data from different sources (e.g., databases, files, APIs) into a unified dataset. The challenge lies in resolving schema conflicts, duplicates, and different formats.

- **Tasks in Integration**:
    - **Entity Identification**: Identifying equivalent entities from multiple datasets.
    - **Schema Integration**: Merging different schemas while resolving conflicts.
    - **Handling Redundancy**: Removing duplicates and resolving inconsistencies between data from different sources.

Data Reduction : Data reduction aims to reduce the volume of data while retaining its essential characteristics for analysis. This is especially important for handling large datasets efficiently.

- **Techniques**:
    - **Dimensionality Reduction**: Reducing the number of features using methods like Principal Component Analysis (PCA) or Feature Selection.
    - **Numerosity Reduction**: Reducing the number of data records through sampling or clustering.
    - **Data Compression**: Applying data compression algorithms to reduce storage needs.

Data Transformation and Data Discretization : **Data Transformation**: This involves converting data into formats that are more appropriate for analysis. It can include scaling, normalization, and aggregation of data.

- **Scaling/Normalization**: Transforming data to a uniform scale, often to bring all features to the same range (e.g., Min-Max scaling).
- **Encoding Categorical Data**: Transforming categorical variables into numerical values (e.g., one-hot encoding, label encoding).
- **Aggregation**: Summarizing data, such as converting hourly data into daily averages.

Data Discretization: This process involves converting continuous data into discrete intervals or buckets, making it easier for certain algorithms (e.g., decision trees) to process.

- **Binning**: Dividing data into equal-width or equal-frequency bins.
- **Clustering**: Grouping similar data points to reduce complexity.

Various types of data:

- **Numerical**

  It represents quantitative measurement. Ex.: Height of a person, stock prices.

- **Discrete Data**

  Integer based, often counts of something. Ex.: How many times did I toss “Heads”?

- **Continuous Data**

  It has an infinite number of possible values. Ex.: How much rainfall on a given day?

- **Categorical Data**

  Qualitative data, Ex.: Gender, Yes/No, etc. Assign some number to categorical data but they don’t have any mathematical meaning

- **Ordinal Data**

  Mixture of numerical and categorical data. Categorical data has mathematical meaning. For example: Movie rating on a scale of 1–5. Rating must be 1,2,3,4,5. They have mathematical meaning. E.x. movie ratings, etc.

Label encoding:

In label encoding, each category is mapped to a number or a label. The labels chosen for the categories have no relationship. So, categories that have some ties or are close to each other lose such information after encoding. It supports the pandas dataframe as input and can transform data.

One-Hot Encoding:

A one hot encoding allows the representation of categorical data to be more expressive. Many Machine Learning algorithms cannot work with categorical data directly. The categories must be converted into numbers.

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Operations to be performed on dataset:

Steps in Preprocessing of Data

1. Importing Python Modules/Libraries
2. Importing data
3. Displaying data
4. Creating the Independent and Dependent variables
5. Replacing missing value with meaningful value
6. Encoding categorical data
7. Splitting the data into training and test set
8. Doing feature scaling on data
9. Use any 3-4 graphs/plots

Output:

In [ ]:
```
import pandas as pd
import numpy as np
```

In [ ]:
```
data=pd.read_csv('/content/data.csv')
display(data)
```

	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_Wheels	Number of Doors	Market Category
0	BMW	1 Series M	2011	premium unleaded (required)	335.0	6.0	MANUAL	rear wheel drive	2.0	Factory Tuner,Luxury,High-Performance
1	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance
2	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,High-Performance
3	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance
4	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury
...	...	...	...	...	...	...	...	...	...	...
11909	Acura	ZDX	2012	premium unleaded (required)	300.0	6.0	AUTOMATIC	all wheel drive	4.0	Crossover,Hatchback,Luxury
11910	Acura	ZDX	2012	premium unleaded (required)	300.0	6.0	AUTOMATIC	all wheel drive	4.0	Crossover,Hatchback,Luxury
11911	Acura	ZDX	2012	premium unleaded	300.0	6.0	AUTOMATIC	all wheel drive	4.0	Crossover,Hatchback,Luxury

In [ ]:
```
df.isnull().sum()#returns the total null values
```

Out[ ]:
```
Make                   0
Model                  0
Year                   0
Engine Fuel Type       3
Engine HP             69
Engine Cylinders      30
Transmission Type      0
Driven_Wheels          0
Number of Doors        6
Market Category     3742
Vehicle Size           0
Vehicle Style          0
highway MPG            0
city mpg               0
Popularity             0
MSRP                   0
dtype: int64
```

In [ ]:
```
df['Engine HP']
```

Out[ ]:
```
0        335.0
1        300.0
2        300.0
3        230.0
4        230.0
         ...
11909    300.0
11910    300.0
11911    300.0
11912    300.0
11913    221.0
Name: Engine HP, Length: 11914, dtype: float64
```

In [ ]:
```
df['Engine HP'].isnull().sum()
```

```
11911      all wheel drive
11912      all wheel drive
11913    front wheel drive
Name: Driven_Wheels, Length: 11914, dtype: object
```

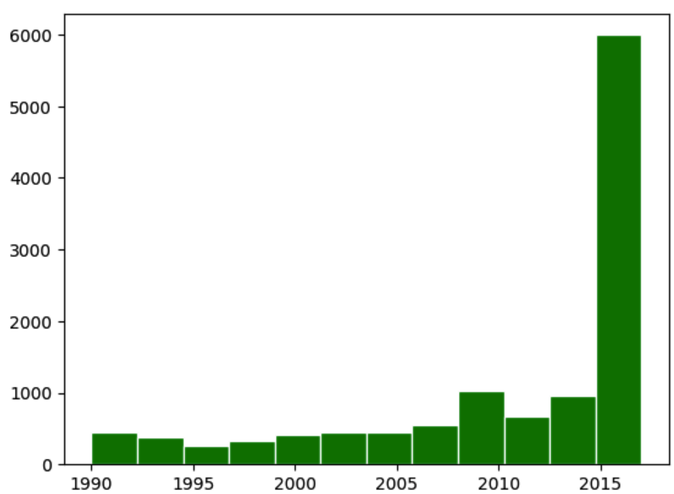
In [ ]:
```python
one_hot_encoded_data = pd.get_dummies(df, columns = ['Driven_Wheels'])
display(one_hot_encoded_data)
```

	Make	Model	Year	Engine Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Number of Doors	Market Category	Vehicle Size	V
0	BMW	1 Series M	2011	premium unleaded (required)	335.0	6.0	MANUAL	2.0	Factory Tuner,Luxury,High-Performance	Compact	
1	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	2.0	Luxury,Performance	Compact	Conv
2	BMW	1 Series	2011	premium unleaded (required)	300.0	6.0	MANUAL	2.0	Luxury,High-Performance	Compact	
3	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	2.0	Luxury,Performance	Compact	
4	BMW	1 Series	2011	premium unleaded (required)	230.0	6.0	MANUAL	2.0	Luxury	Compact	Conv
...	...	...	...	...	...	...	...	...	...	...	
11909	Acura	ZDX	2012	premium unleaded (required)	300.0	6.0	AUTOMATIC	4.0	Crossover,Hatchback,Luxury	Midsize	Hat
11910	Acura	ZDX	2012	premium unleaded (required)	300.0	6.0	AUTOMATIC	4.0	Crossover,Hatchback,Luxury	Midsize	Hat
11911	Acura	ZDX	2012	premium unleaded	300.0	6.0	AUTOMATIC	4.0	Crossover,Hatchback,Luxury	Midsize	Hat

In [ ]:
```python
import matplotlib.pyplot as plt
```

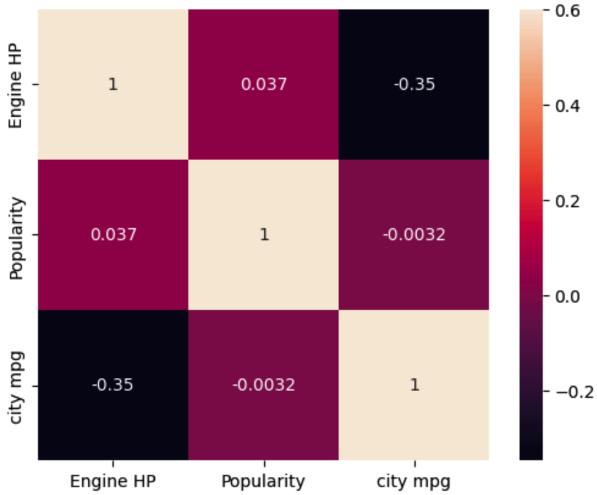
In [ ]:
```python
plt.hist(df['Year'],color='green',edgecolor='white',bins=12)
```

Out[ ]:
```
(array([ 452.,  372.,  266.,  329.,  406.,  443.,  448.,  550., 1026.,
         672.,  955., 5995.]),
 array([1990.  , 1992.25, 1994.5 , 1996.75, 1999.  , 2001.25, 2003.5 ,
        2005.75, 2008.  , 2010.25, 2012.5 , 2014.75, 2017.  ]),
 <BarContainer object of 12 artists>)
```



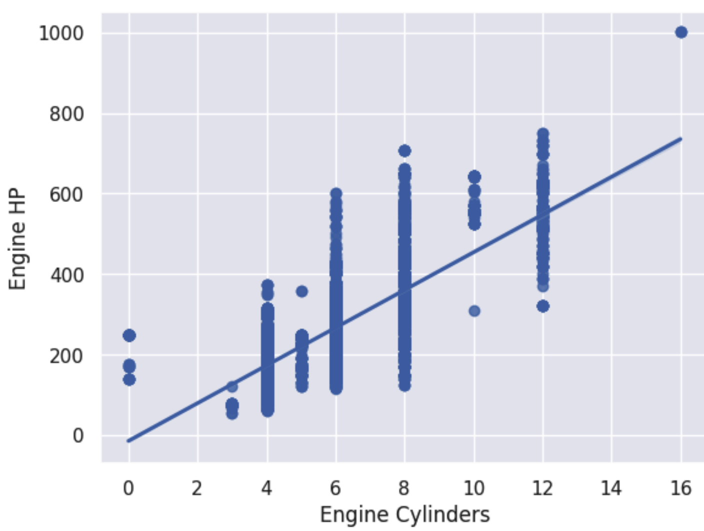
city mpg 0.346308 0.003217 1.000000

```
In [ ]: import seaborn as sns
        sns.heatmap(corel,vmax=.6 ,annot=True,square=True)
        plt.show()
```




```
In [ ]: sns.set(style="darkgrid")
        sns.regplot(x=df['Engine Cylinders'],y=df['Engine HP'])
        plt.show()
```

```
In [ ]: sns.set(style="darkgrid")
        sns.regplot(x=df['Engine Cylinders'],y=df['Engine HP'])
        plt.show()
```



FAQs:

1) List two common libraries for data manipulation. Give an example for each library.

Pandas: A popular library for data manipulation, commonly used for handling tabular data.

NumPy: A library that provides support for large, multi-dimensional arrays and matrices, along with mathematical functions.

2) Give an example on how ordinal data is handled in a Machine Learning algorithm.
Ordinal data, which has a meaningful order, is typically encoded as integers that reflect this order. For example, "low", "medium", and "high" can be represented as 1, 2, and 3, preserving the hierarchy for the algorithm.

3) Can one hot encoding be used for continuous data. If yes, give an example.
Yes, one-hot encoding can be used for continuous data if the data is first divided into categories or bins. For example, continuous age data can be grouped into bins like 18-25, 26-35, etc., and then one-hot encoded.

4) Why is it necessary to encode strings?
Strings need to be encoded into numerical values because most machine learning algorithms can only work with numbers. Encoding strings into numerical values allows the algorithms to process and analyze the data effectively.

5) State the significance of exploratory data analysis.
Exploratory Data Analysis (EDA) helps to understand the underlying structure of the data, identify patterns, detect anomalies, and test hypotheses. It also informs decisions about how to best preprocess the data for modeling.

6) 'Handling missing values of data is an important step in Data preprocessing.' Comment on the statement.
Handling missing values is crucial because missing data can lead to incorrect analysis or faulty model predictions. Strategies such as imputation or removing incomplete data ensure that the model works with a clean, reliable dataset.

7) State any 4 graphical techniques/plots used for exploratory data analysis.

1. Histogram
2. Scatter Plot
3. Box Plot
4. Heatmap

8) Describe the ***box-and-whisker*** plot
A box-and-whisker plot, or box plot, is a graphical representation of data distribution. It displays the minimum, first quartile, median, third quartile, and maximum values, helping identify outliers and the spread of the data.

9) Explain Central Tendency functions.

Central tendency functions measure the center or typical value of a dataset. The most common measures are:

- **Mean**: The average of the data.
- **Median**: The middle value when data is ordered.
- **Mode**: The most frequent value in the data.

Conclusion:

Data collection, data preparation, handling various data types was studied and exploratory data analysis was performed.