Exploratory Data Analysis(EDA) from scratch in Python!

BEGINNER DATA EXPLORATION DATA VISUALIZATION PYTHON STRUCTURED DATA TECHNIQUE

Introduction

Exploratory data analysis is one of the best practices used in data science today. While starting a career in Data Science, people generally don't know the difference between Data analysis and exploratory data analysis. There is not a very big difference between the two, but both have different purposes.

Exploratory Data Analysis(EDA): Exploratory data analysis is a complement to inferential statistics, which tends to be fairly rigid with rules and formulas. At an advanced level, EDA involves looking at and describing the data set from different angles and then summarizing it.

Data Analysis: Data Analysis is the statistics and probability to figure out trends in the data set. It is used to show historical data by using some analytics tools. It helps in drilling down the information, to transform metrics, facts, and figures into initiatives for improvement.

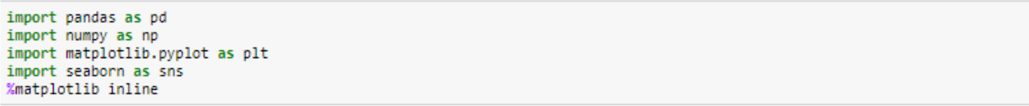
Exploratory Data Analysis(EDA)

We will explore a Data set and perform the exploratory data analysis. The major topics to be covered are below:

– Handle Missing value
– Removing duplicates
– Outlier Treatment

– Normalizing and Scaling(Numerical Variables)
– Encoding Categorical variables(Dummy Variables)
– Bivariate Analysis

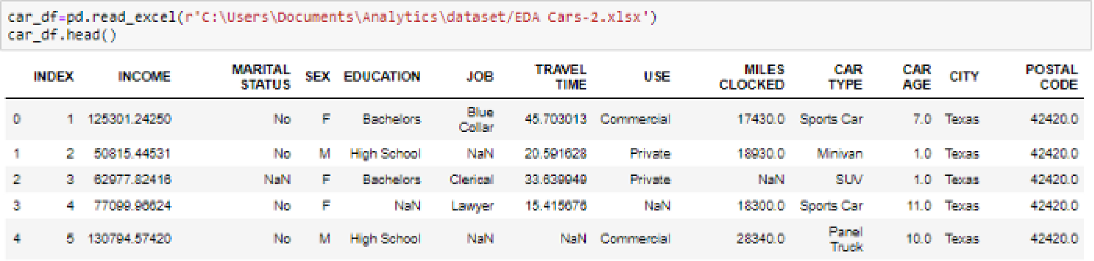
# Importing Libraries



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

# Loading the data set

We will be loading the EDA cars excel file using pandas. For this, we will be using read_excel file.



```
car_df=pd.read_excel(r'C:\Users\Documents\Analytics\dataset/EDA Cars-2.xlsx')
car_df.head()
```

	INDEX	INCOME	MARITAL STATUS	SEX	EDUCATION	JOB	TRAVEL TIME	USE	MILES CLOCKED	CAR TYPE	CAR AGE	CITY	POSTAL CODE
0	1	125301.24250	No	F	Bachelors	Blue Collar	45.703013	Commercial	17430.0	Sports Car	7.0	Texas	42420.0
1	2	50815.44531	No	M	High School	NaN	20.591628	Private	18930.0	Minivan	1.0	Texas	42420.0
2	3	62977.82416	NaN	F	Bachelors	Clerical	33.639949	Private	NaN	SUV	1.0	Texas	42420.0
3	4	77099.96624	No	F	NaN	Lawyer	15.415676	NaN	18300.0	Sports Car	11.0	Texas	42420.0
4	5	130794.57420	No	M	High School	NaN	NaN	Commercial	28340.0	Panel Truck	10.0	Texas	42420.0

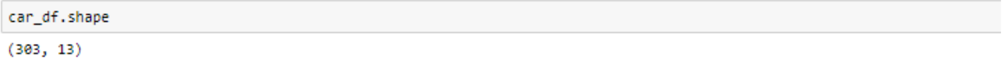
# Basic Data Exploration

In this step, we will perform the below operations to check what the data set comprises of. We will check the below things:

– head of the dataset
– the shape of the dataset
– info of the dataset
– summary of the dataset

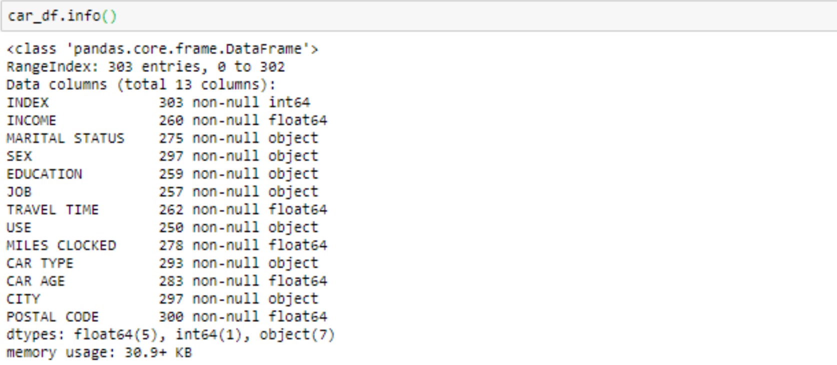
1. The head function will tell you the top records in the data set. By default, python shows you only the top 5 records.

2. The shape attribute tells us a number of observations and variables we have in the data set. It is used to check the dimension of data. The cars data set has 303 observations and 13 variables in the data set.



```
car_df.shape
(303, 13)
```

3. info() is used to check the Information about the data and the datatypes of each respective attribute.



```
car_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 13 columns):
INDEX             303 non-null int64
INCOME            260 non-null float64
MARITAL STATUS    275 non-null object
SEX               297 non-null object
EDUCATION         259 non-null object
JOB               257 non-null object
TRAVEL TIME       262 non-null float64
USE               250 non-null object
MILES CLOCKED     278 non-null float64
CAR TYPE          293 non-null object
CAR AGE           283 non-null float64
CITY              297 non-null object
POSTAL CODE       300 non-null float64
dtypes: float64(5), int64(1), object(7)
memory usage: 30.9+ KB
```

Looking at the data in the head function and in info, we know that the variable Income and travel time are of float data type instead of the object. So we will convert it into the float. Also, there are some invalid values like @@ and '*' in the data which we will be treating as missing values.

4. The described method will help to see how data has been spread for numerical values. We can clearly see the minimum value, mean values, different percentile values, and maximum values.

Handling missing value

We can see that we have various missing values in the respective columns. There are various ways of treating your missing values in the data set. And which technique to use when is actually dependent on the type of data you are dealing with.

- Drop the missing values: In this case, we drop the missing values from those variables. In case there are very few missing values you can drop those values.
- Impute with mean value: For the numerical column, you can replace the missing values with mean values. Before replacing with mean value, it is advisable to check that the variable shouldn't have extreme values .i.e. outliers.
- Impute with median value: For the numerical column, you can also replace the missing values with median values. In case you have extreme values such as outliers it is advisable to use the median approach.
- Impute with mode value: For the categorical column, you can replace the missing values with mode values i.e the frequent ones.

In this exercise, we will replace the numerical columns with median values and for categorical columns, we will drop the missing values.

Handling Duplicate records

Since we have 14 duplicate records in the data, we will remove this from the data set so that we get only distinct records. Post removing the duplicate, we will check whether the duplicates have been removed from the data set or not.

Handling Outlier

Outliers, being the most extreme observations, may include the sample maximum or sample minimum, or both, depending on whether they are extremely high or low. However, the sample maximum and minimum are not always outliers because they may not be unusually far from other observations.

We Generally identify outliers with the help of boxplot, so here box plot shows some of the data points outside the range of the data.

Box-plot before removing outliers

Looking at the box plot, it seems that the variables INCOME, have outlier present in the variables. These outliers value needs to be teated and there are several ways of treating them:

- Drop the outlier value
- Replace the outlier value using the IQR

#Boxplot After removing outlier

Box-plot after removing outliers

Bivariate Analysis

When we talk about bivariate analysis, it means analyzing 2 variables. since we know there are Numerical and categorical variables, there is a way of analyzing these variables as shown below:

1. **Numerical vs. Numerical**

    1. Scatterplot
    2. Line plot
    3. Heatmap for correlation
    4. Joint plot

2. **Categorical vs. Numerical**

    1. Bar chart
    2. Violin plot
    3. Categorical box plot
    4.Swarm plot

3. **Two Categorical Variables**

    1. Bar chart
    2. Grouped bar chart
    3. Point plot

If we need to find the correlation-

Correlation between all the variables

Normalizing and Scaling

Often the variables of the data set are of different scales i.e. one variable is in millions and others in only 100. For e.g. in our data set Income is having values in thousands and age in just two digits. Since the data in these variables are of different scales, it is tough to compare these variables.

Feature scaling (also known as data normalization) is the method used to standardize the range of features of data. Since the range of values of data may vary widely, it becomes a necessary step in data preprocessing while using machine learning algorithms.

In this method, we convert variables with different scales of measurements into a single scale. StandardScaler normalizes the data using the formula (x-mean)/standard deviation. We will be doing this only for the numerical variables.

ENCODING

One-Hot-Encoding is used to create dummy variables to replace the categories in a categorical variable into features of each category and represent it using 1 or 0 based on the presence or absence of the categorical value in the record.

This is required to do since the machine learning algorithms only work on the numerical data. That is why there is a need to convert the categorical column into a numerical one.

get_dummies is the method that creates a dummy variable for each categorical variable.

About the Author

Ritika Singh – Data Scientist

I am a Data scientist by profession and a Blogger by passion. I have been working on machine learning projects for more than 2 years. Here you will find articles on “Machine Learning, Statistics, Deep Learning, NLP and Artificial Intelligence”.

Article Url - https://www.analyticsvidhya.com/blog/2020/08/exploratory-data-analysiseda-from-scratch-in-python/

[Guest Blog](#)