Assignment 3: Data Understanding

Estimated time needed: **60** minutes

Objectives

After completing this assignment you will be able to:

- Load a dataset into our Jupyter Notebook
- Obtain insights from dataset using functions provided by Pandas
- Pre-process and explore the features or characteristics to predict the price of car

Note: Please submit the PDF file of the Jupyter Notebook using the following instructions:

- `pip install notebook-as-pdf` in the command prompt (Windows) or Terminal (Mac OS)
- Restart the current notebook.
- File -> Download as -> PDF via HTML (.pdf)

Te
of
Co

In [1]:
```
import pandas as pd
import numpy as np
```

Data Acquisit

There are various formats for a dataset: .csv, .json, .xlsx etc. The dataset can be stored in different places, on your local machine or sometimes online.
In this section, you will learn how to load a dataset into our Jupyter Notebook.
In our case, the Automobile Dataset is an online source, and it is in a CSV (comma separated value) format.
Let's use this dataset as an example to practice data reading.

- Data source: used-cars.csv
- Data type: csv

The Pandas Library is a useful tool that enables us to read various datasets into a dataframe; our Jupyter notebook platforms have a built-in **Pandas Library** so that all we need to do is import Pandas without installing.

Read Data

We use `pandas.read_` function to read the csv file. In the brackets, we put the file path along with a quotation mark so that pandas will read the file into a dataframe from that address. The file path can be either an URL or your local file address.
Because the data does not include headers, we can add an argument `headers = None` inside the `read_csv()` method so that pandas will not automatically set the first row as a header.
You can also assign the dataset to any variable you create.

In [2]:
```
# Read in the file by specifying its directory, and assign it to variable "df"
path = 'C:/Users/saryjamie/Desktop/Data_Science/Module3/used-cars.csv'
df = pd.read_csv(path, header=None)
```

After reading the dataset, we can use the `dataframe.hea` method to check the top n rows of the dataframe, where n is an integer. Contrary to `dataframe.hea` `dataframe.tai` will show you the bottom n rows of the dataframe.

In [3]:
```
# show the first 5 rows using dataframe.head() method
print("The first 5 rows of the dataframe")
df.head(5)
```

Out[3]:
The first 5 rows of the dataframe

	0	1	2	3	4	5	6	7	8	9	...	16	17	18	19	20	21	22	23	24	25
0	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111.0	5000.0	21	27	13495.0
1	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111.0	5000.0	21	27	16500.0
2	1	NaN	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	154.0	5000.0	19	26	16500.0
3	2	164.0	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	102.0	5500.0	24	30	13950.0
4	2	164.0	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	115.0	5500.0	18	22	17450.0

5 rows × 26 columns

Questi
#1:

Check the bottom 10 rows of data frame "df".

In [4]:
```
# Write your code below and press Shift+Enter to execute
print("The bottom 10 rows of the dataframe")
df.tail(10)
```

Out[4]:
The bottom 10 rows of the dataframe

	0	1	2	3	4	5	6	7	8	9	...	16	17	18	19	20	21	22	23	24	25
195	-1	74.0	volvo	gas	std	four	wagon	rwd	front	104.3	...	141	mpfi	3.78	3.15	9.5	114.0	5400.0	23	28	13415.0
196	-2	103.0	volvo	gas	std	four	sedan	rwd	front	104.3	...	141	mpfi	3.78	3.15	9.5	114.0	5400.0	24	28	15985.0
197	-1	74.0	volvo	gas	std	four	wagon	rwd	front	104.3	...	141	mpfi	3.78	3.15	9.5	114.0	5400.0	24	28	16515.0
198	-2	103.0	volvo	gas	turbo	four	sedan	rwd	front	104.3	...	130	mpfi	3.62	3.15	7.5	162.0	5100.0	17	22	18420.0
199	-1	74.0	volvo	gas	turbo	four	wagon	rwd	front	104.3	...	130	mpfi	3.62	3.15	7.5	162.0	5100.0	17	22	18950.0
200	-1	95.0	volvo	gas	std	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	114.0	5400.0	23	28	16845.0
201	-1	95.0	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	8.7	160.0	5300.0	19	25	19045.0
202	-1	95.0	volvo	gas	std	four	sedan	rwd	front	109.1	...	173	mpfi	3.58	2.87	8.8	134.0	5500.0	18	23	21485.0
203	-1	95.0	volvo	diesel	turbo	four	sedan	rwd	front	109.1	...	145	idi	3.01	3.40	23.0	106.0	4800.0	26	27	22470.0
204	-1	95.0	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	114.0	5400.0	19	25	22625.0

10 rows × 26 columns

Add Headers

Take a look at our dataset. Pandas automatically set the header with an integer starting from 0.

To better describe our data, we can introduce a header. This information is available at:
https://archive.ics.uci.edu/ml/datasets/autor

Thus, we have to add headers manually.

First, we create a list "headers" that include all column names in order. Then, we use `dataframe.columns = headers` to replace the headers with the list we created.

In [5]:
```
# create headers list
headers = ["symboling","normalized-losses","make","fuel-type","aspiration", "num-of-doors","body-style",
           "drive-wheels","engine-location","wheel-base", "length","width","height","curb-weight","engine-type",
           "num-of-cylinders", "engine-size","fuel-system","bore","stroke","compression-ratio","horsepower",
           "peak-rpm","city-mpg","highway-mpg","price"]
```

We replace headers and recheck our data

In [6]:
```
df.columns = headers
df.head()
```

Out[6]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke	compression-ratio	horsepower	peak-rpm	city-mpg	highway-mpg	price
0	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111.0	5000.0	21	27	13495.0
1	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111.0	5000.0	21	27	16500.0
2	1	NaN	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	154.0	5000.0	19	26	16500.0
3	2	164.0	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	102.0	5500.0	24	30	13950.0
4	2	164.0	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	115.0	5500.0	18	22	17450.0

5 rows × 26 columns

W
c
dr
m
ve
al
th
cc
"c
ar
fo