Task 3: Report

About automobile dataset

Data Preparation

Today, a report is written about ‘automobile’ dataset which has been loaded using these given functions, automobile_filename = "automobile.csv" and

automobile = pd.read_csv(automobile_filename, sep='#', decimal='.', header = None, \names=['Symboling', 'Normalised-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'])

automobile

Here, all the variables’ names are included with the above dataset.

Task 1(Step:2: Checking the data types)

In this data types step, the ‘automobile’ dataset has checked using ‘automobile.dtypes’ and ‘automobile.info()’ functions where 238 observations and 26 columns have obtained. Apart from this, float64(11), int64(5), object (10) data types are found from the datset. Here, ‘astype()’ function is used to convert few variables from ‘object’ to ‘category’ datatypes. During this conversion process, ‘str.strip()’ and ‘replace()’ functions are applied to complete the conversion process adequately. Apart from this, `dict()` with `zip()` function have been used to convert ‘Num-of-cylinders’ variable from ‘object’ to ‘interger’ format. Finally, when the datatypes are checked again using `info()` function, I have found that category(6), float64(11), int64(6), object(3) datatypes are enlightened in the output panel.

Task 1(Step:3: Typos):

In this step, some typing error has been fixed using the mask condition where a specific range works for getting the perfect and desirable condition. By applying this mask, some data have been selected which are greater than 3 in the `Symboling` variable and this are known as impossible value and also indicated the typing error. That’s why, this function `mask_automobile=(automobile['Symboling'] > 3)` has been used to fix the error. After that, this function is also used to fix the error completely, `automobile = automobile.loc[automobile['Symboling'] < 4]`. To check the dropping result of impossible value '4', value_counts() function has been used and found that no impossible values are staying in the output pane. Apart from this manipulation, `unique()` and `replace()` function have been used to fix the other variables such as `Make`, `Fuel-type`, `Body-style`, `Drive-wheels`, `Engine-type` and `Aspiration` errors.

Task 1(Step:4: Extra-whitespaces):

In this step, the `value_counts()`, `unique()` and `str.strip()` functions are applied to fi x the extra-whitespaces error of the `Make`, `Fuel-type`, `Body-style`, `Drive-wheels` , `Engine-type`, `Num-of-doors` and `Aspiration` variable.

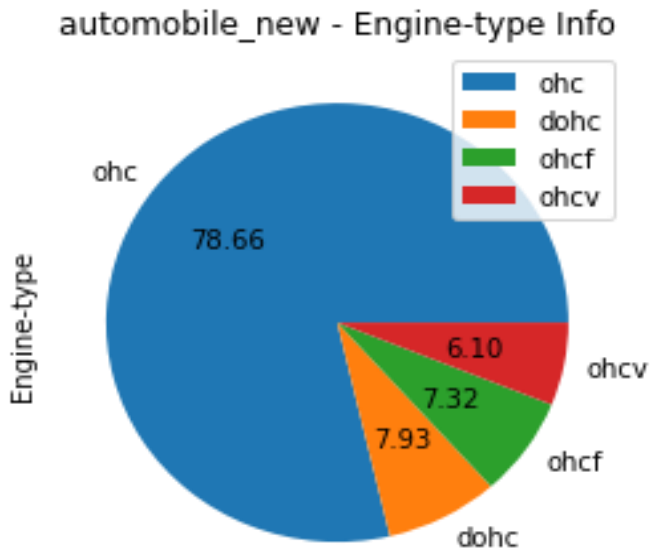
Task 1(Step:5: Upper/Lower-case):
In this step, it is decided to keep all values' alphabet into the lower-case. Therefore, t o increase the consistency and readability, the `value_counts()`, `unique()` and `repl ace()`functions are used to fix the upper-case error of the `Make`, `Fuel-type`, `Body-style`, `Drive-wheels`, `Engine-type`, and `Aspiration` variables' values.

Task 1(Step:6: Sanity checks):
To check the sanity of the `Symboling` variable's, the `unique()` function has been us ed here. To make this 'automobile' data more validable and acceptable, this function `automobile = automobile[automobile.Symboling < 3]` has been used, As the auto co ndition, +3 indicates high-risk and -3 indicates safe. After that, this `automobile = aut omobile[automobile.Price > 0]` function is used to remove the redundant value from t he `Price` variable which has extended the sanity of the `automobile` dataset. To che ck both variables' values numbers, the `value_counts()`function is applied.

Task 1(Step:7: Missing values):
To check the missing values, `head()` and `isnull()` functions have been used and fo und that few missing values which are known as `NaN(Not a Number)` are highlighti ng in the output panel. To drop these missing values, used `dropna()` function and th e result again for all variables and the entire data frame using `.loc[:,:].isnull().sum()` and `isnull()` function.

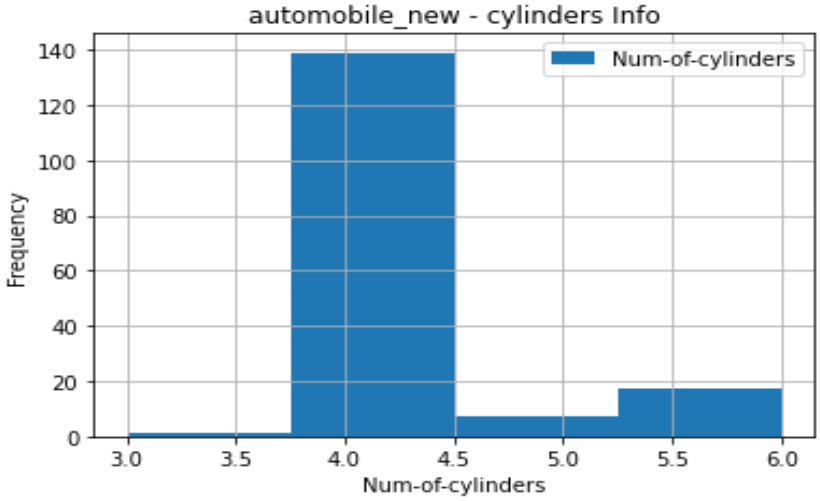
Task 2(Data Exploration):

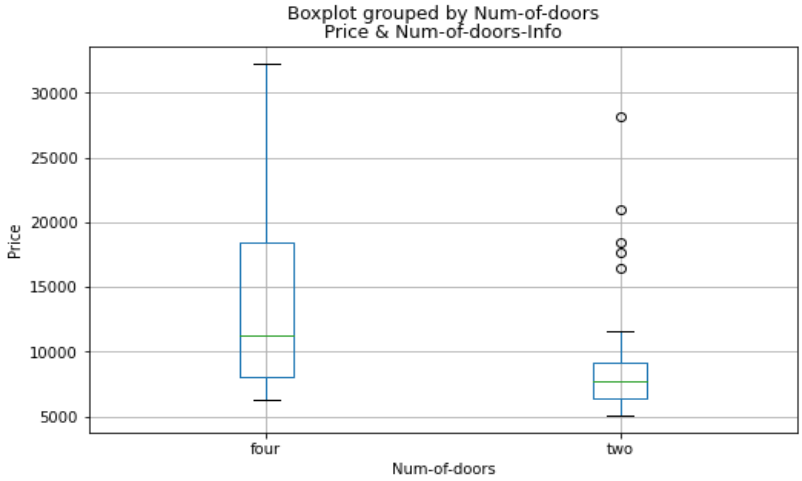
Task2(Subtask 1): Chosen 3 columns
Pie Chart:



Here, I have chosen this pie chart graph to explore the nominal variable's statistical a nd graphical condition with proportionally which named as `Engine-type`. The pie ch art has given above. The above step's explanation is given in the jupyter notebook

Histogram: Here, I have chosen this histogram chart to explore the numerical variable's statistical and graphical condition which named as `Num-of-cylinders`. The histogram chart is given as below.

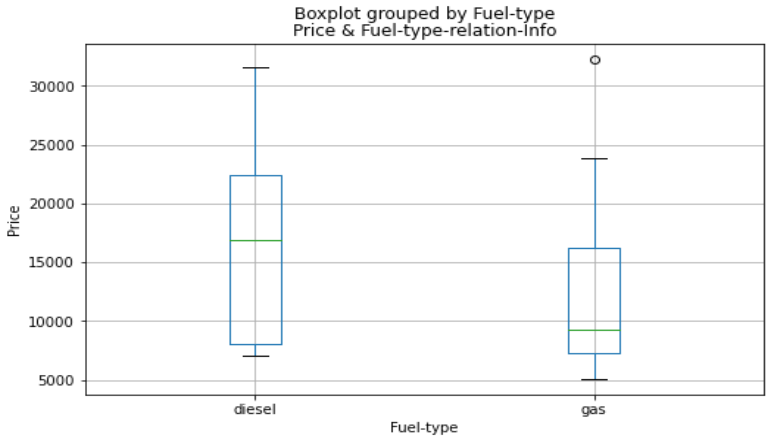




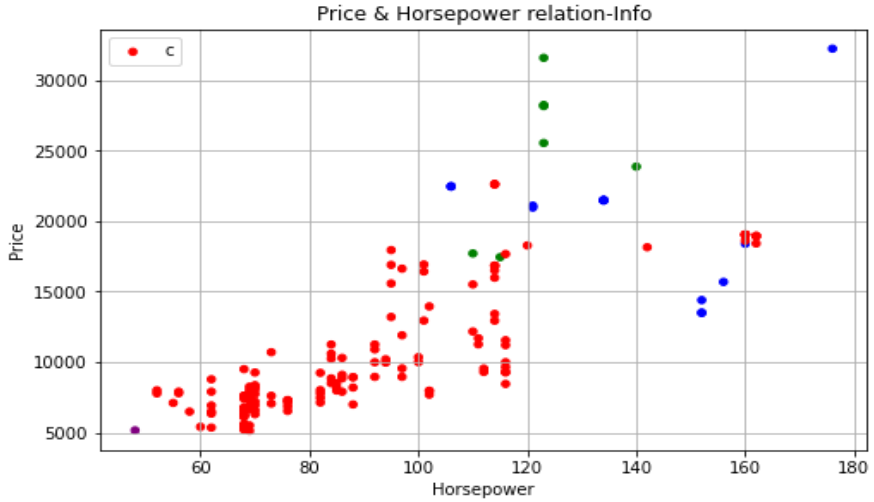
Boxplot:

Here, the boxplot has been chosen to explore the numerical and ordinal variable's key figures within the distribution which can also help to detect the outliers. As the boxplot can show the data by group, `Price` and `Num-of-doors` variables have been added by making group. The boxplot figure is given above.

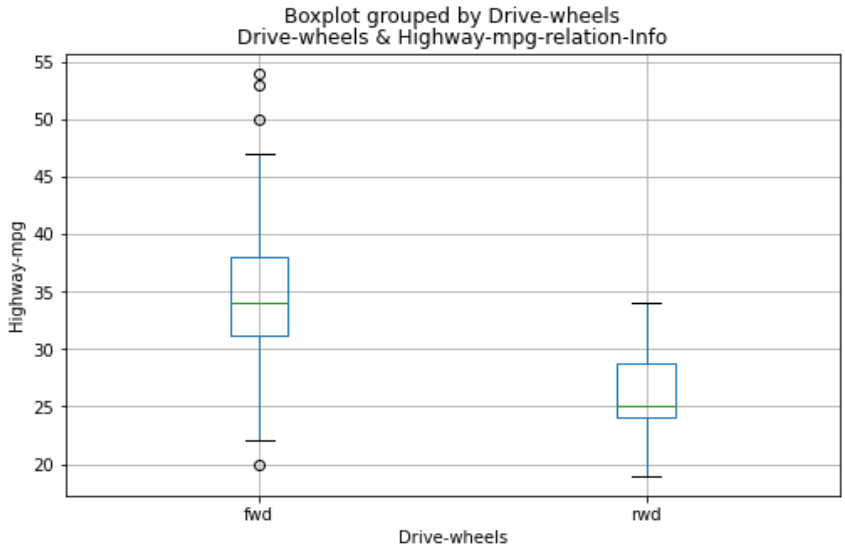
Task2(Subtask 2): Relationships between columns



After investigating the boxplot, it is predictable that the Fuel-type of the automobiles are demonstrating different price where diesel is showing higher price than the gas. However, according to the median explanation, diesel holder autos are more expensive than the gas holder auto which is highlighting in the above boxplot.



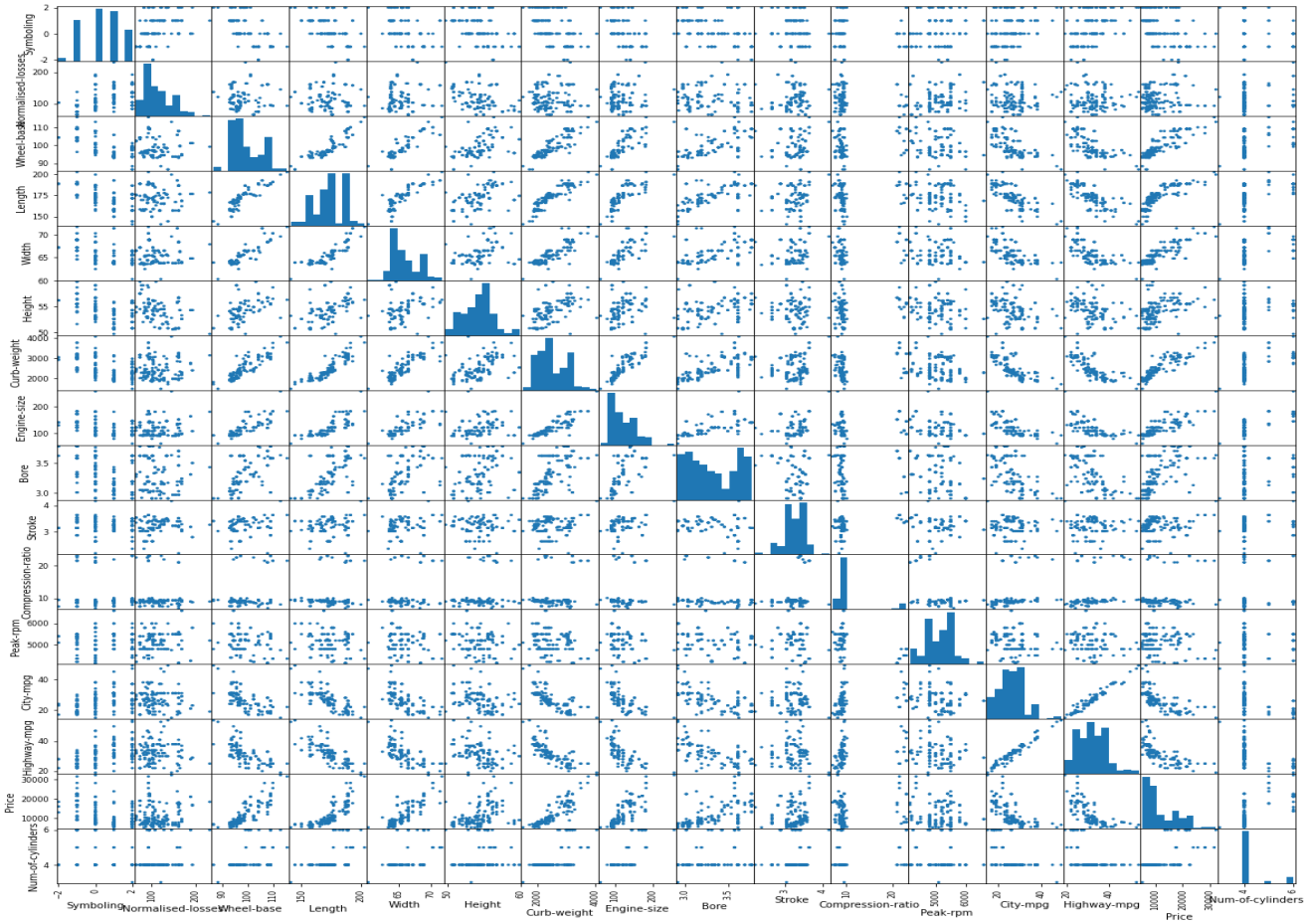
After investigating the scatter plot, it is predictable that 4 cylinders automobiles' which are known as red dots in the above scatter plot which is having comparatively more horsepower than the other number of cylinders or other dots colors. Besides this, it is also hypothetically believable that 4 cylinders automobiles are cheaper than the other cylinders despite of having it's more Horsepowers. Therefore, as the more horsepower produces its better acceleration, which is a strong factor in its overall performance, 4 cylinders automobiles will be more usable and acceptable than the other cylinders automobiles to the respected users.



After investigating the boxplot, it is predictable that the `Drive-wheels` of the automobiles are demonstrating the different Highway-mpg where fwd is showing greater highway-mpg than the `rwd`. However, according to the median exploration, 'fwd(front wheel drive)'autos are more powerful to make highway-mpg(miles per gallon) drive than the 'rwd(rear wheel drive)' autos which is highlighting from the above boxplot.

Task 2(Subtask 3): Scatter matrix for all numerical variables:

As this scatter matrix plot is generated with more than one variable, hence, all different features are plotted against all other features(variables). Here, scatter matrix diagram represents the histogram chart of the `Symboling` variable's which is distributed with it's all values by showing almost normalised position. But, comparing with the `Symboling` feature's distribution, the `Num-of-cylinders` feature's distribution is less-normalised which is plotted in the bottom right corner of the scatter matrix plot. Here, all numeric variables are highlighting their histogram chart by following the `Symboling` variable's figure in diagonally where these features are making strong correlation with each other. Apart from this, the weakest correlation with the features also have been demonstrated because of their scattered values in the top right corner of the scatter matrix plot which has given in the following scatter matrix plot.


Symboling
Normalised-losses
Wheel-base
Length
Width
Height
Curb-weight
Engine-size
Bore
Stroke
Compression-ratio
Peak-rpm
City-mpg
Highway-mpg
Price
Num-of-cylinders