**Expt. No.:04 Date: 05/10/2020**

**DATA VISUALIZATION USING PYTHON**

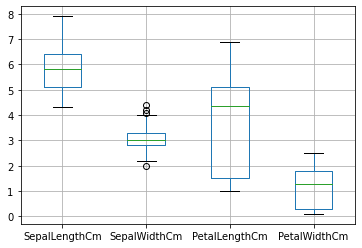
**AIM:**

1. **Data Visualization using Pandas and Seaborn:**
   1. Plot a histogram
   2. Plot a scatterplot and identify redundant attributes
   3. Plot a box plot, bar chart, pie chart for all the attributes

**THEORY:** We use Visualization tools to get a graphical summary of the dataset.

***Box Plot:***

Box Plot is the visual representation of the depicting groups of numerical data through their quartiles. Boxplot is also used for detect the outlier in data set. It captures the summary of the data efficiently with a simple box and whiskers and allows us to compare easily across groups. Boxplot summarizes a sample data using 25th, 50th and 75th percentiles.



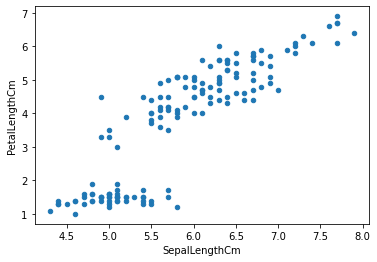
*Figure 1: Box Plot for IRIS dataset*

A box plot consist of 5 things.

* Minimum
* First Quartile or 25%
* Median (Second Quartile) or 50%
* Third Quartile or 75%
* Maximum

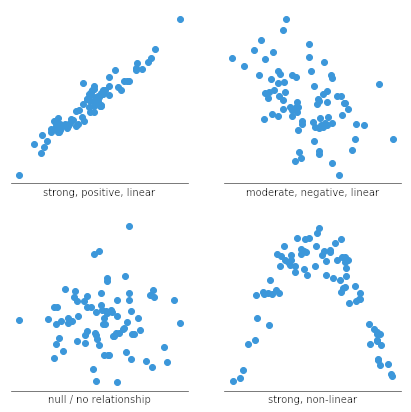
**Scatter Plot**

A scatter plot (aka scatter chart, scatter graph) uses dots to represent values for two different numeric variables. The position of each dot on the horizontal and vertical axis indicates values for an individual data point. Scatter plots are used to observe relationships between variables.



*Figure 2: Scatter plot for SepalLenthCm and PetalLengthCm*

Scatter plots’ primary uses are to observe and show relationships between two numeric variables. The dots in a scatter plot not only report the values of individual data points, but also patterns when the data are taken as a whole.

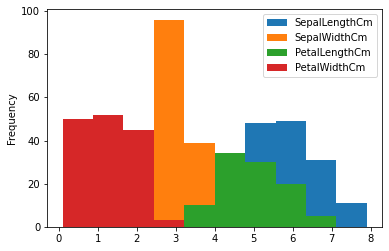


Identification of correlational relationships are common with scatter plots. In these cases, we want to know, if we were given a particular horizontal value, what a good prediction would be for the vertical value. Relationships between variables can be described in many ways: positive or negative, strong or weak, linear or nonlinear.

**Histogram:**

A histogram is an approximate representation of the [distribution](https://en.wikipedia.org/wiki/Frequency_distribution) of numerical data. It was first introduced by [Karl Pearson](https://en.wikipedia.org/wiki/Karl_Pearson). To construct a histogram, the first step is to "[bin](https://en.wikipedia.org/wiki/Data_binning)" (or "[bucket](https://en.wikipedia.org/wiki/Data_binning)") the range of values—that is, divide the entire range of values into a series of intervals—and then count how many values fall into each interval. The bins are usually specified as consecutive, non-overlapping [intervals](https://en.wikipedia.org/wiki/Interval_(mathematics)) of a variable. The bins (intervals) must be adjacent and are often (but not required to be) of equal size.

If the bins are of equal size, a rectangle is erected over the bin with height proportional to the [frequency](https://en.wikipedia.org/wiki/Frequency_(statistics))—the number of cases in each bin. A histogram may also be [normalized](https://en.wikipedia.org/wiki/Normalization_(statistics)) to display "relative" frequencies. It then shows the proportion of cases that fall into each of several [categories](https://en.wikipedia.org/wiki/Categorization), with the sum of the heights equaling 1.



*Figure 3: Histogram for IRIS Dataset*

However, bins need not be of equal width; in that case, the erected rectangle is defined to have its *area* proportional to the frequency of cases in the bin. The vertical axis is then not the frequency but *frequency density*—the number of cases per unit of the variable on the horizontal axis. Examples of variable bin width are displayed on Census bureau data below.

As the adjacent bins leave no gaps, the rectangles of a histogram touch each other to indicate that the original variable is continuous.

Histograms give a rough sense of the density of the underlying distribution of the data, and often for [density estimation](https://en.wikipedia.org/wiki/Density_estimation): estimating the [probability density function](https://en.wikipedia.org/wiki/Probability_density_function) of the underlying variable. The total area of a histogram used for probability density is always normalized to 1. If the length of the intervals on the *x*-axis are all 1, then a histogram is identical to a [relative frequency](https://en.wikipedia.org/wiki/Relative_frequency) plot.

**PROGRAM:**

%matplotlib inline

import seaborn as sns

**#histogram**

cereal\_data.plot(kind="hist")

cereal\_data\_numeric = cereal\_data.drop(columns=['name','mfr','type'])

cereal\_data\_categorical = cereal\_data[['name','mfr','type']]

**#scatter plots**

sns.pairplot(cereal\_data\_numeric)

**#correlation matrix to analyse which attributes have high linear relationship**

corr = cereal\_data.corr()

**#heatmap**

sns.heatmap(corr,linewidths=.8)

**#box-plot**

cereal\_data.plot.box(figsize= (15,5))

**#pie-charts**

cereal\_data\_categorical["mfr"].value\_counts().plot(kind="pie")

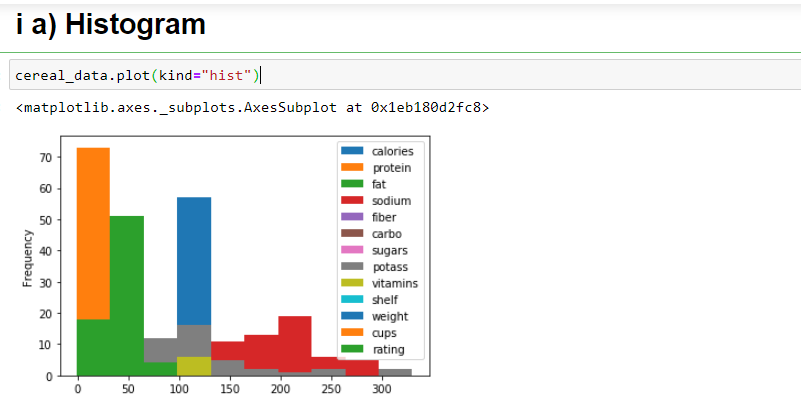
cereal\_data\_categorical["type"].value\_counts().plot(kind="pie")

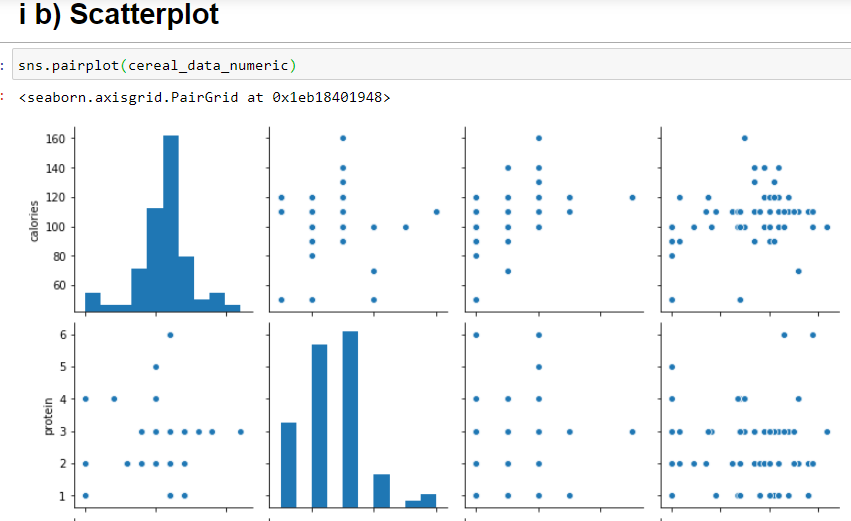
**#bar-charts**

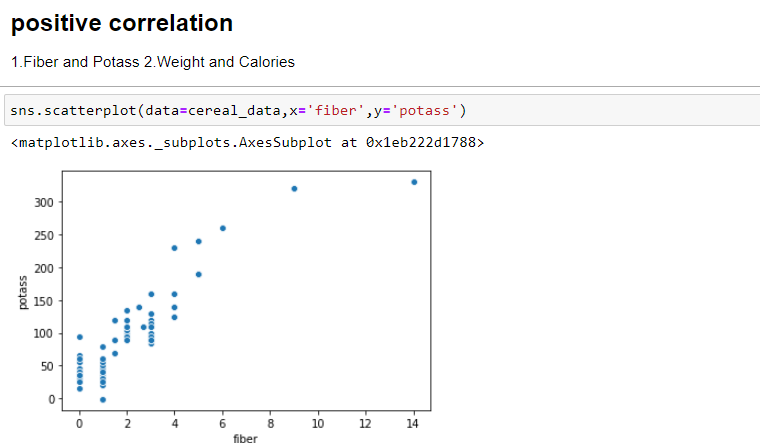
sns.catplot(x="mfr",kind="count",color="orange",data=cereal\_data\_categorical,order=cereal\_data\_categorical['mfr'].value\_counts().index)

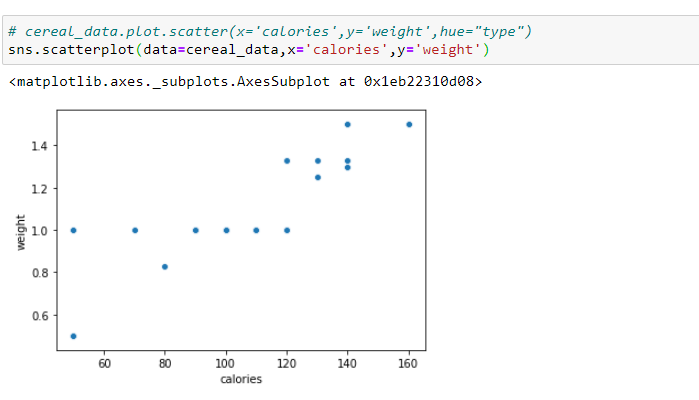
sns.catplot(x="type",kind="count",color="orange",data=cereal\_data\_categorical,order=cereal\_data\_categorical['type'].value\_counts().index)

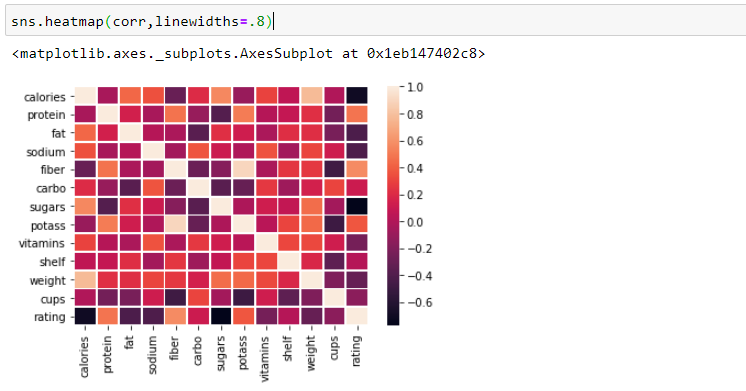
**OUTPUT:**

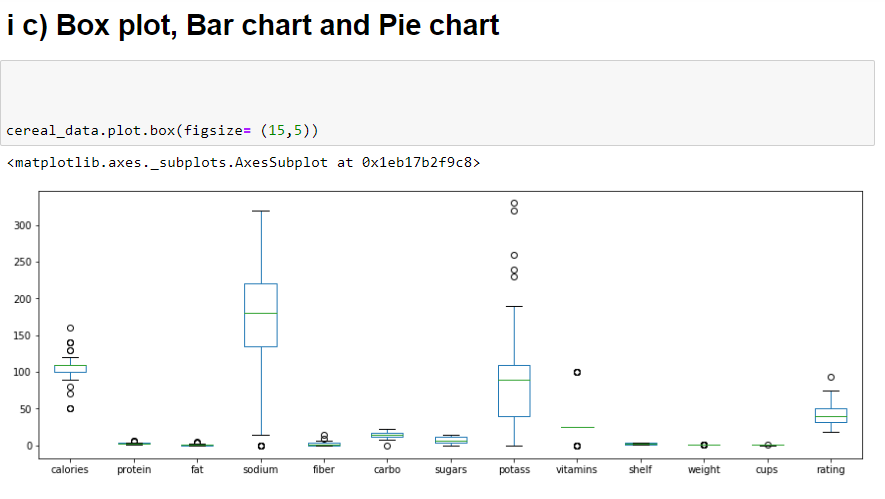
****

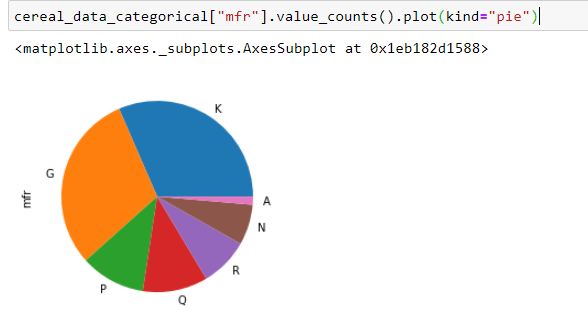
****

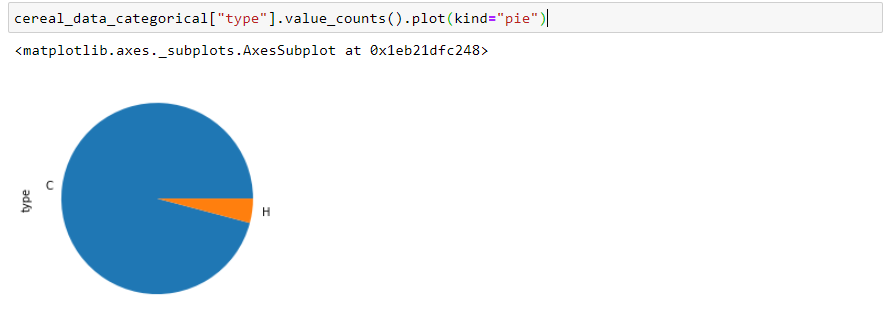
****

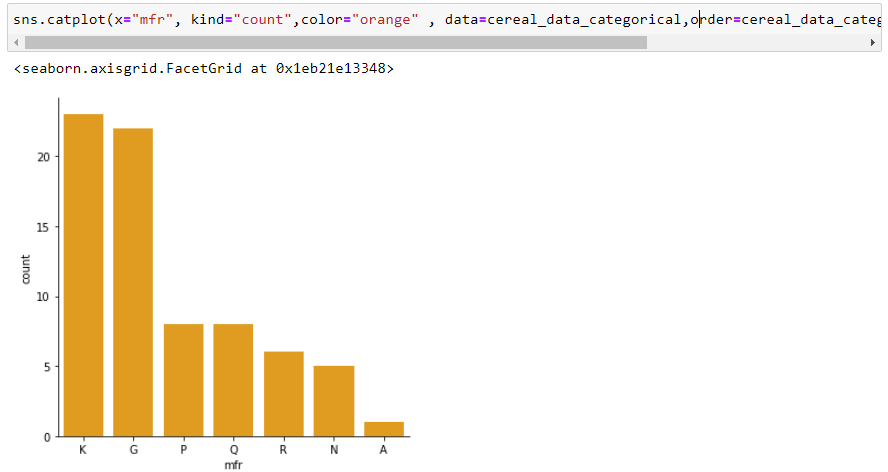
****

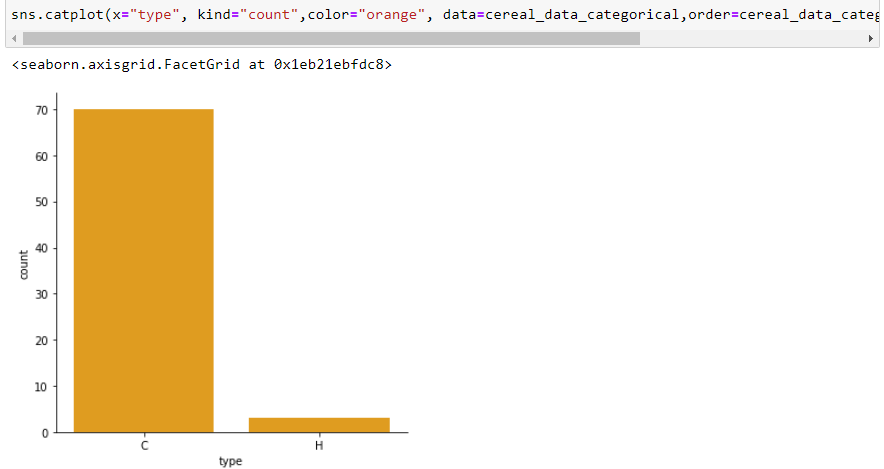
****

****

****

****

****

****

**CONCLUSION:**

Data Visualisation of Cereal Dataset was completed using pandas and seaborn.