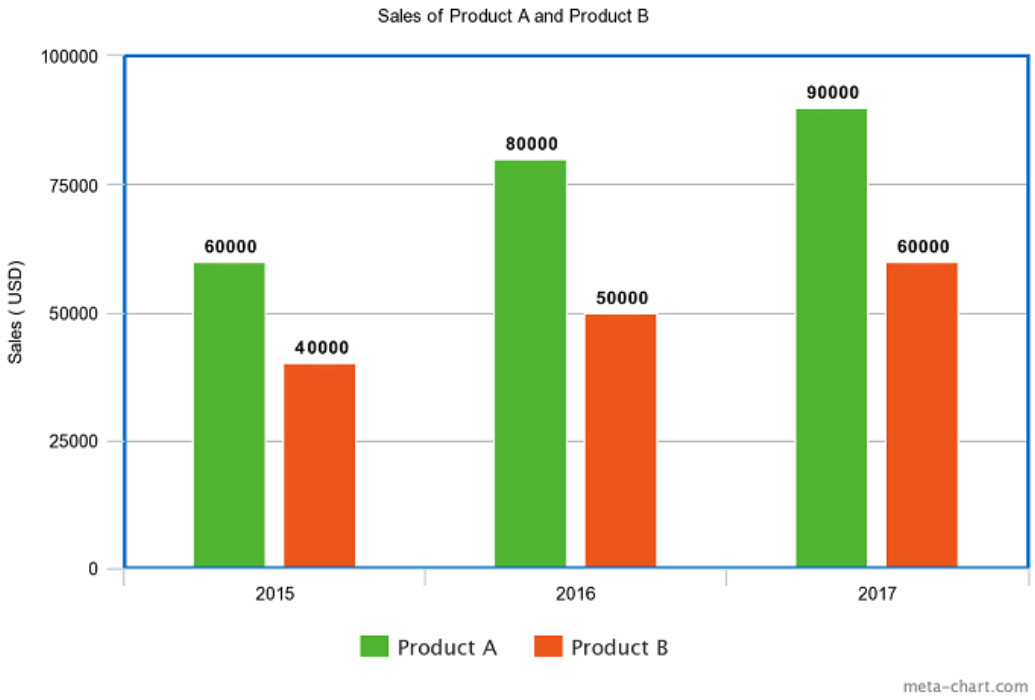
Assignment 1

1. Different types of Charts and Graphs used in Data Visualization

Bar Graph:

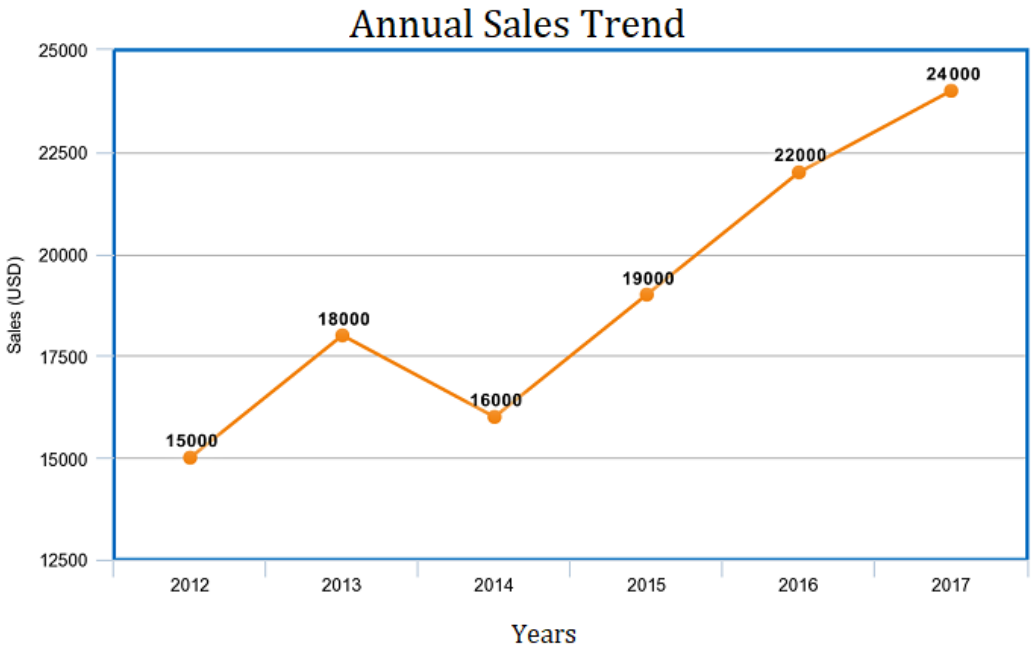
Bar Graph represents categorical data with rectangular bars. They are among the most popular types of graph and charts.

Bar graph is plotted on the X and Y axis where each of the axis represents a particular data type and the rectangular bars represent the values of those data types.



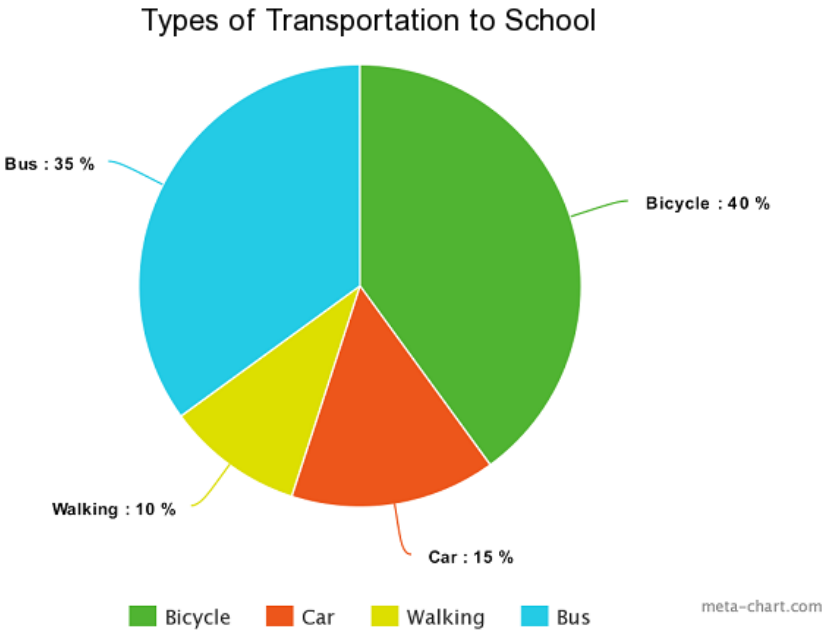
Line Graph:

A line graph is a type of data visualization that displays information as a series of data points connected by lines. It is commonly used to show the relationship between two variables, typically one independent variable and one dependent variable along the two axes.



Pie Chart:

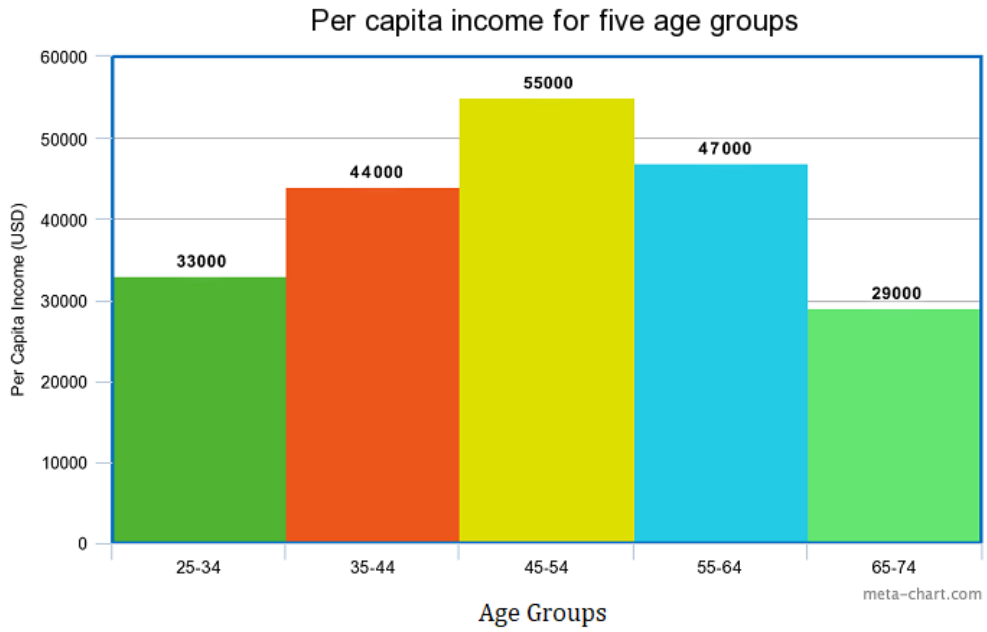
Pie charts are primarily used to show the distribution of a categorical variable and how each category contributes to the whole. They are ideal for illustrating the relative sizes or proportions of different parts within a whole.



Histogram:

A histogram shows continuous data in ordered rectangular columns. Usually, there are no gaps between the columns.

The histogram displays a frequency distribution of a data set. At first glance, histograms look alike to bar graphs. However, there is a key difference between them. Bar Chart represents categorical data and histogram represents continuous data.

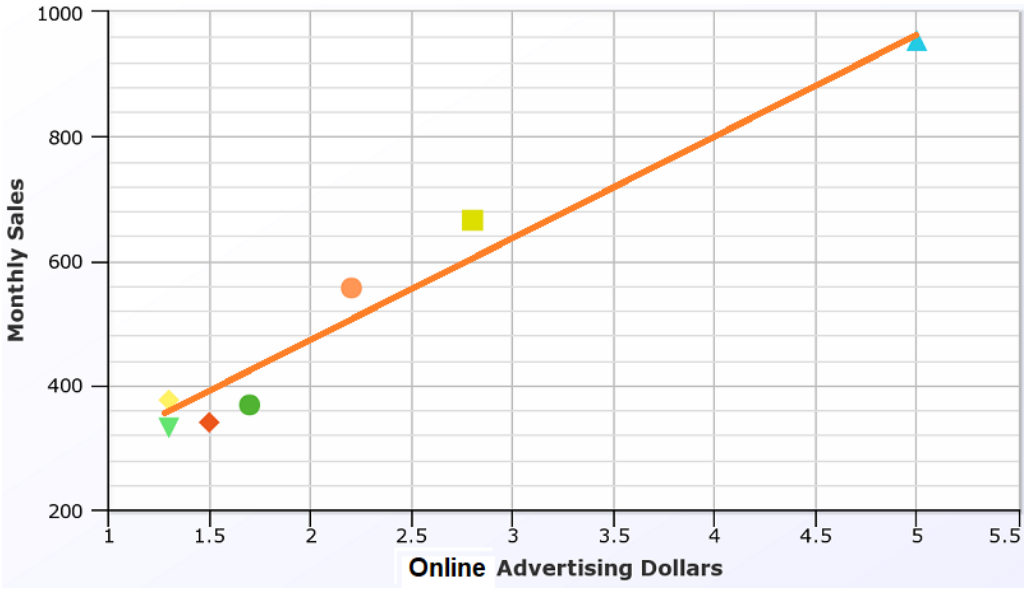


Scatter Plot:

Scatter Plot is used to display the relationship between two variables in a dataset. It consists of individual data points, each representing a unique observation, and it helps visualize the correlation, pattern, or lack thereof between the two variables

Scatter plots are primarily used to visually examine the relationship between two continuous or numerical variables. They help identify patterns, trends, or correlations between these variables.

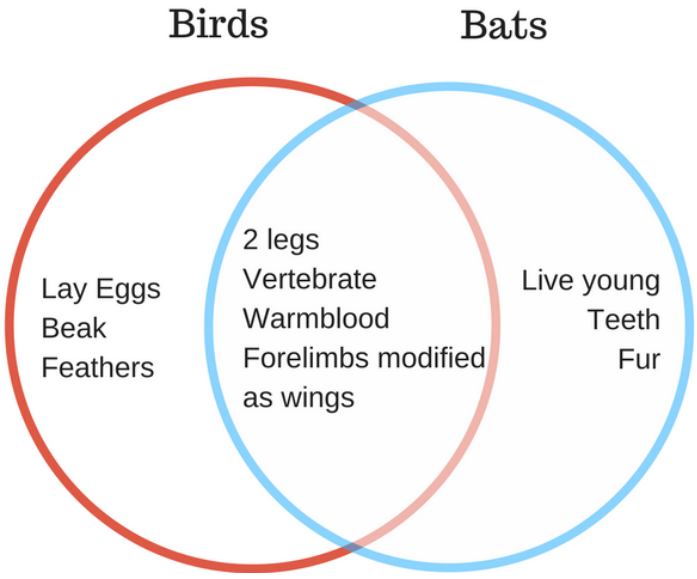
We can also easily visualize any outliers in the data set with a Scatter plot.



Venn Chart:

Venn Diagram (also called primary diagram, set diagram or logic diagrams) uses overlapping circles to visualize the logical relationships between two or more groups of items.

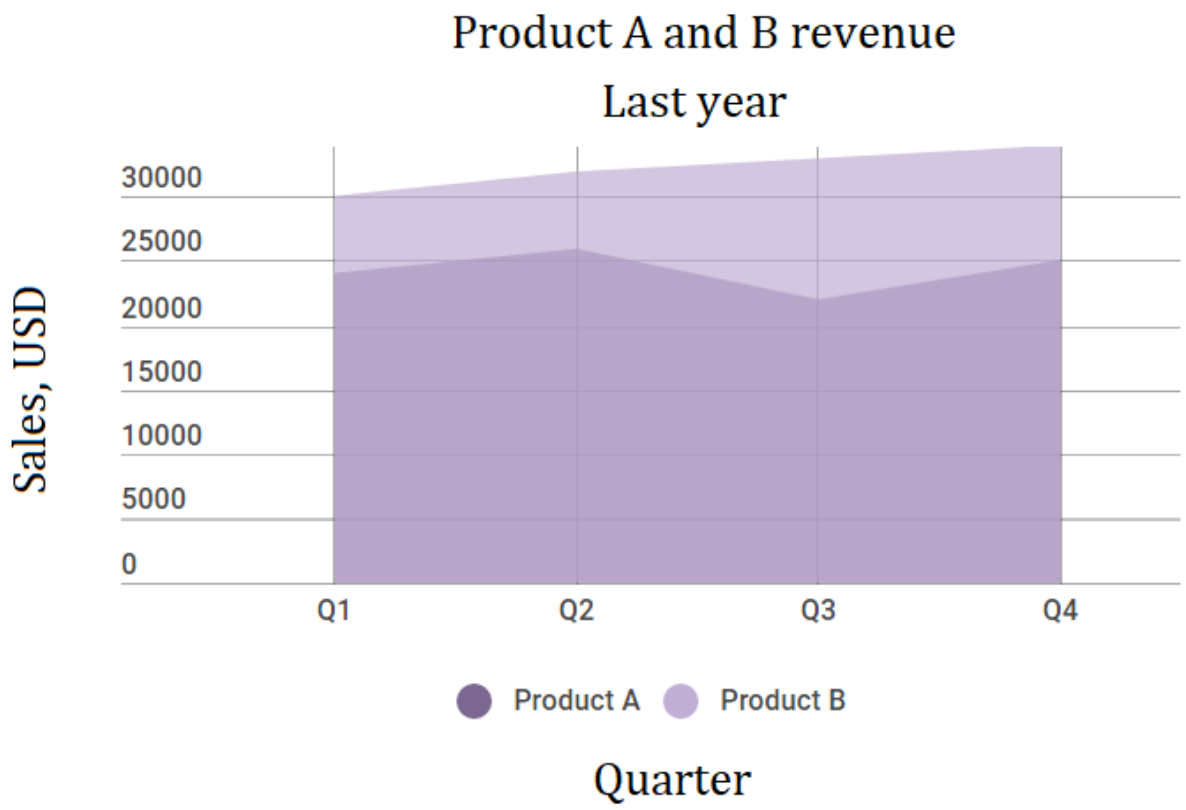
The basic structure of the Venn diagram is usually overlapping circles. The items in the overlapping section have specific common characteristics. Items in the outer portions of the circles do not have common traits



Area Chart:

An area chart is a type of data visualization that is similar to a line chart but with the area beneath the line filled with color or shading. It is used to represent data over time or other ordered categories and is particularly effective for illustrating how different categories contribute to a whole.

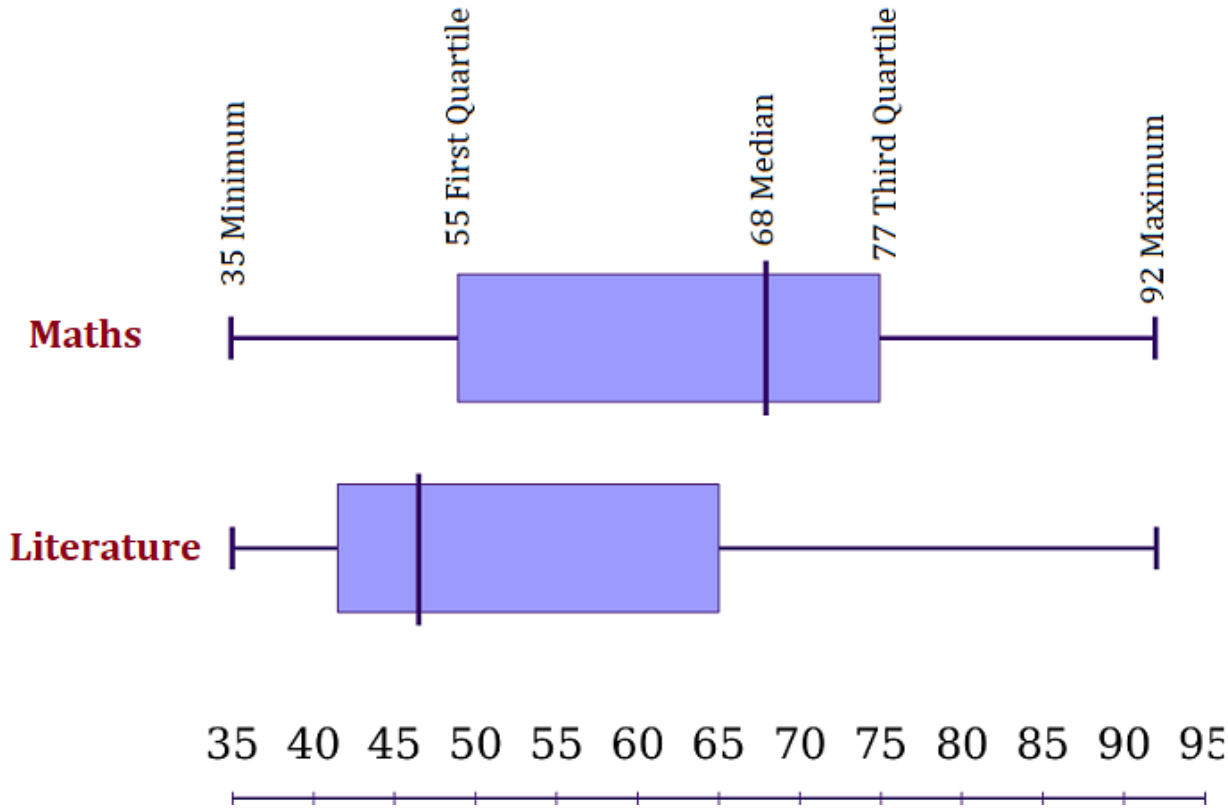
Area charts are used to compare the contributions or proportions of different categories or series to a total or each other. They make it easy to visualize how individual components add up to the whole.



Box Plot:

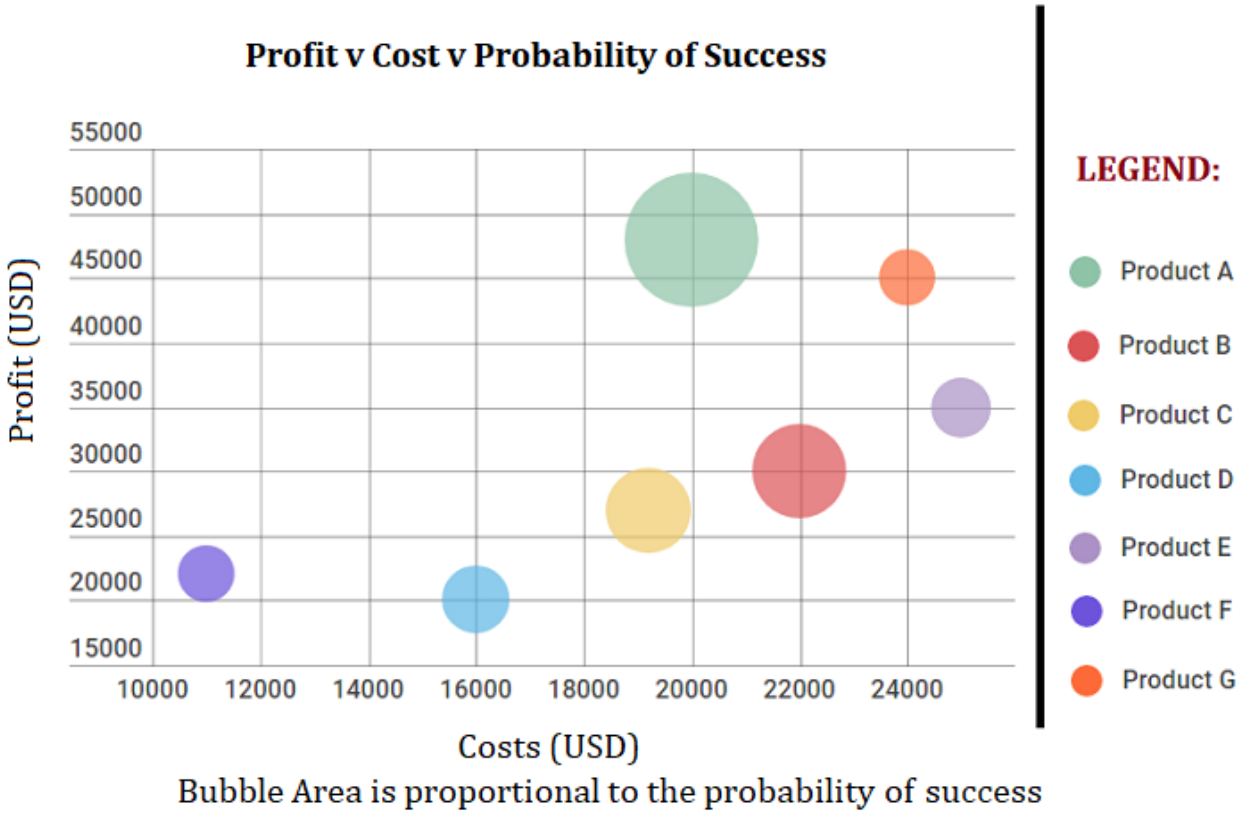
A box plot, also known as a box-and-whisker plot, is a graphical representation of the distribution and summary statistics of a dataset. It is particularly useful for visualizing the spread and central tendencies of the data.

Box plots are used to display the distribution of a dataset, including its central tendency, spread, skewness, and the presence of outliers.



Bubble Chart:

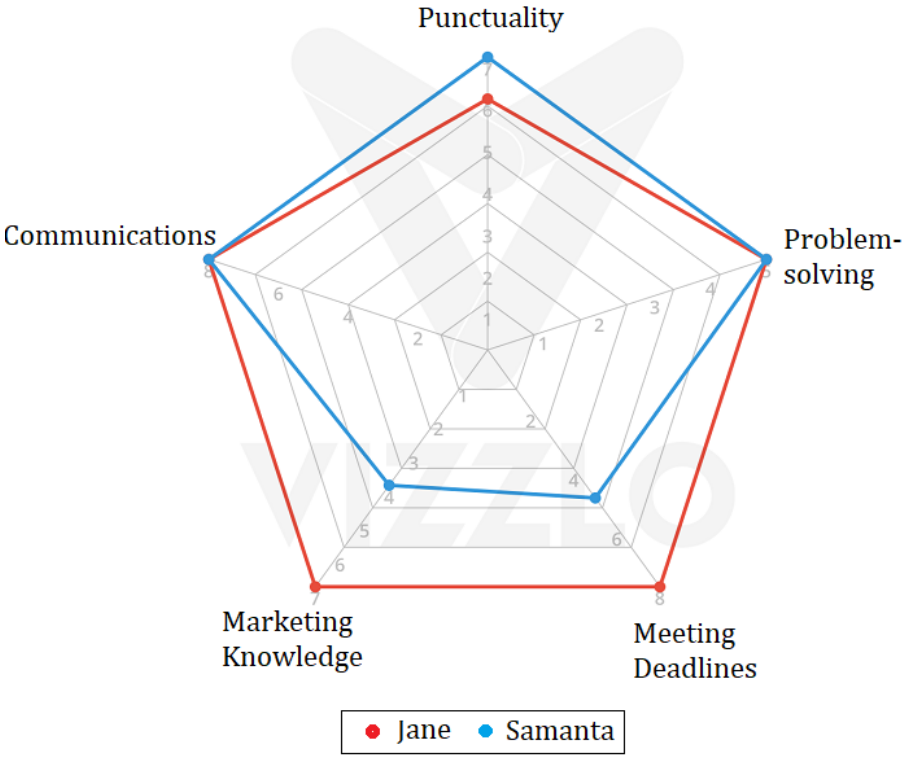
A bubble chart is a data visualization that displays three sets of data in a two-dimensional chart. It is similar to a scatter plot but introduces the additional dimension of the size of the data points, represented by circles or bubbles.



Radar Chart:

A radar chart, also known as a spider chart or web chart, is a data visualization tool used to display multivariate data in a two-dimensional chart with multiple axes. It is particularly useful for comparing the performance or attributes of different entities across multiple variables.

Employee Skills Analysis
(scale 1-8, being the highest)

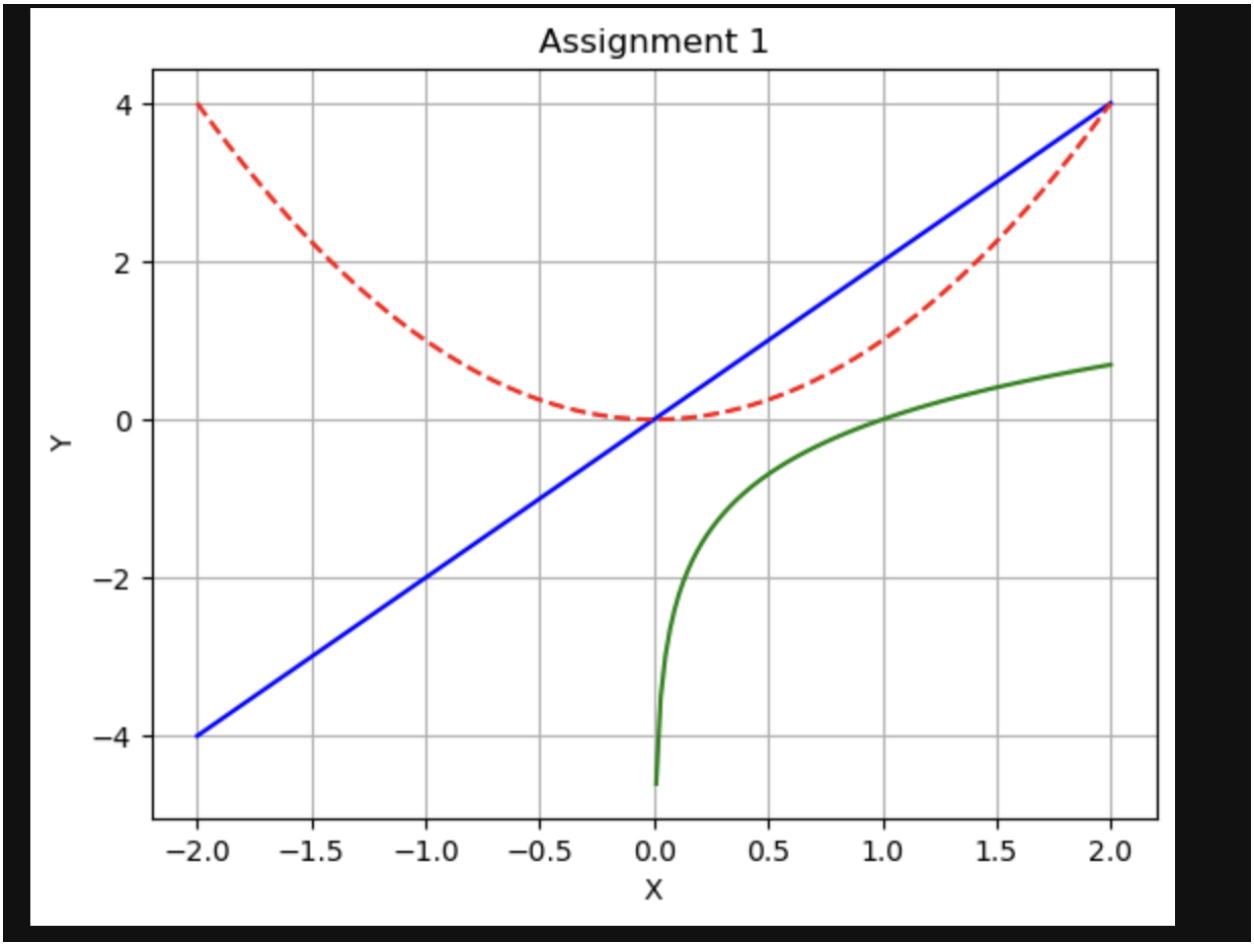


2. Identify at least three common mistakes in data visualization

- Misleading scales:
    1. Using non-zero baselines: Starting a y-axis at a value greater than zero can exaggerate differences and make small changes seem significant.
    2. Inconsistent scaling: Using different scales on multiple axes in the same visualization can lead to misinterpretation of relationships between variables.
- Ignoring context:
    1. Lack of context: Failing to provide context or background information can leave viewers puzzled about the significance and implications of the data.
    2. Omitting labels and titles: Not labeling axes, data points, or providing a clear title can make it difficult for viewers to understand what they are looking at.
- Overloading with information:
    1. Too much data: Overcrowding your visualization with too many data points or too many variables can overwhelm viewers and hinder comprehension.
    2. Unnecessary embellishments: Adding excessive decorations like 3D effects, flashy colors, or complex chart types can distract from the data and make it harder to interpret.
- Misleading visual representations:
    1. Inaccurate chart types: Choosing inappropriate chart types, like using a pie chart for data with too many categories, can distort the data and make it harder to compare values.
    2. Truncated axes: Cutting off portions of an axis or using a truncated y-axis can exaggerate differences or hide important information.
- Cherry-picking data:
    1. Selective data presentation: Presenting only a portion of the data or choosing a specific time frame to support a particular narrative can be misleading and biased.
    2. Data omission: Leaving out relevant data points or excluding outliers without explanation can distort the overall picture.

3. Using matplotlib, plot the following functions on a single figure: (20 points).
   - y=2x (where x ranges from -2 to 2)
   - y=x2 (where x ranges from -2 to 2)
   - y = log(x) (where x ranges from 0.01 to 2)

```python
import matplotlib.pyplot as plt
import numpy as np


x = np.linspace(-2, 2, 100)
x1 = np.linspace(0.01, 2, 100)

# 2x
y1 = x * 2

# x^2
y2 = x ** 2

# log(x)
y3 = np.log(x1)


plt.plot(x, y1, label='x*2', color="blue", linestyle="solid")

plt.plot(x, y2, label='x^2', color="red", linestyle="dashed")

plt.plot(x1, y3, label='log(x)', color="green", linestyle="solid")

plt.title("Assignment 1")
plt.xlabel("X")
plt.ylabel("Y")

plt.grid(True)

plt.show()
```


Assignment 1
4
2
0
−2
−4
Y
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
X