GOVERNMENT COLLEGE KASARAGOD

Open Data Lab

GCK

Micro-course in
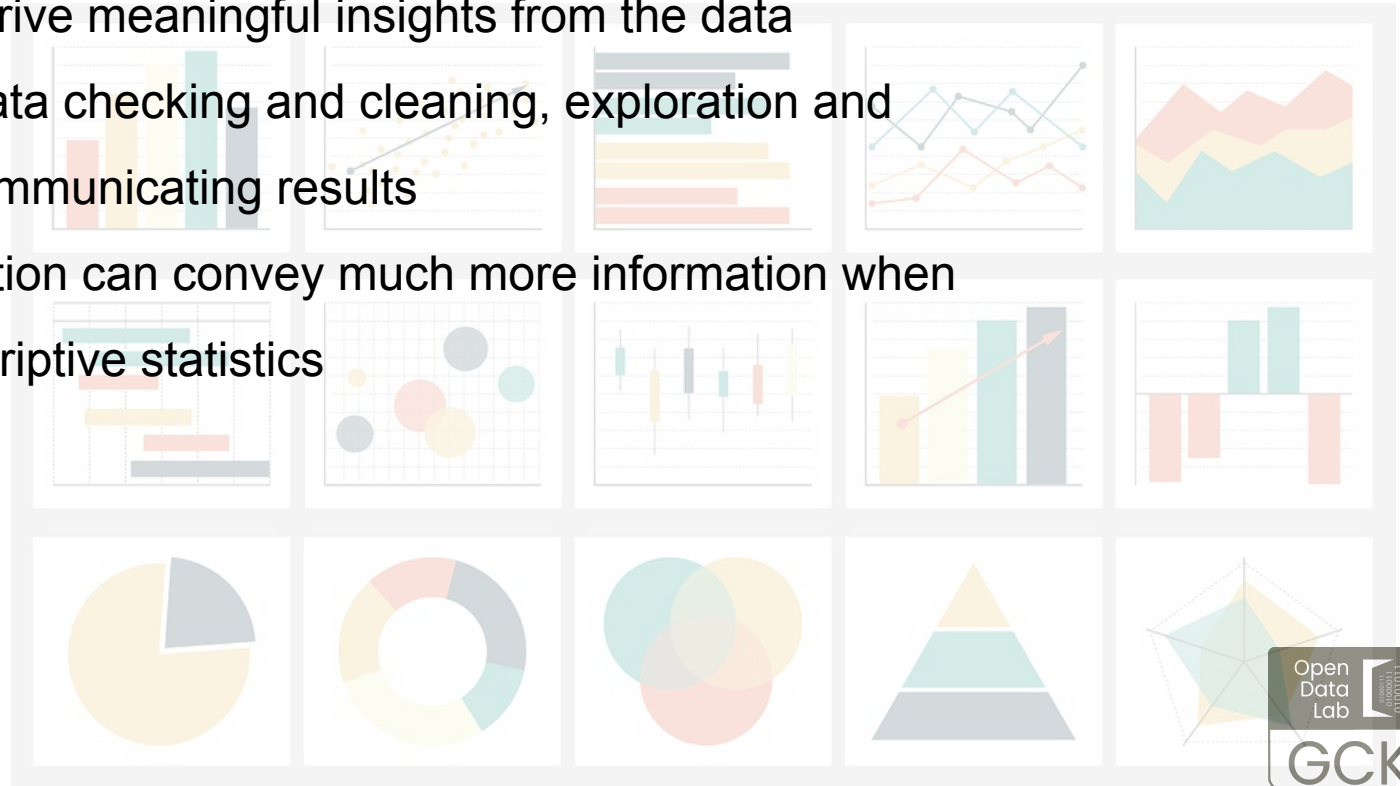
DATA ANALYSIS

Session – 4
Introduction to Popular Data Visualisation Tools

HIRAN DEVA K

Open Data Lab

GCK

Introduction to Data Visualisation

- Process of creating interactive visuals to understand trends, variations, and derive meaningful insights from the data
- Mainly used for data checking and cleaning, exploration and discovery, and communicating results
- Visual representation can convey much more information when compared to descriptive statistics

Different Types of Charts used in Data Analysis

Chart	Visual	X axis	Y axis	Analysis	Example
Scatter plot/Line Plot		Continuous	Continuous	- Understanding linear, non-linear relationship between two variables - Trend analysis, change in KPI over time	- How does heart rate change with age? - How sales of a company varied over a period of time?
Bar Graph		Categorical /Discrete Continuous	Continuous	- How Y (can be any performance indicator) varies across different categories?	- How sales in 2019 varied for different mobile phone brands? i.e. mobile phone brand is the category and sales is the KPI
Stack Bar Graph		Categorical	Continuous	- Relative comparison of multiple categories within a category	- Comparison of revenue generated by Apple, Samsung & Xiaomi across different products like mobile phone, laptops, television, and headsets
Box Plot		Continuous		- Outlier detection - Analysing data distribution across Median and Inter Quartile Range	- How different sales figures across a year is distributed?
Pie Chart		Categorical & Continuous		- Relative comparison of different categories for one single entity in terms of proportion/percentages	- What percentage of Sales in 2019 is constituted by different products under Apple?
Histogram Plot		Continuous	-	- How distribution of values of x varies across different range buckets?	- Distribution of income across income buckets for developing countries

Open Data Lab GCK
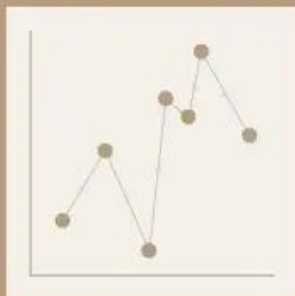
Chart types

scatterplot

The standard way to show the relationship between two variables, each of which has its own axis

line-column

A good way of showing the relationship between an amount (columns) and a rate (line)
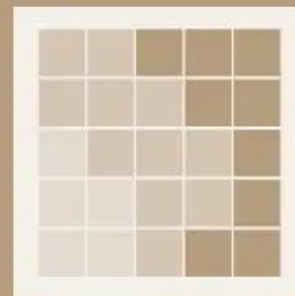
scatterplot-connected

Usually used to show how the relationship between 2 variables has changed over time

Bubble

Like a scatterplot, but adds additional detail by sizing the circles according to a third variable

XY-heatmap

A good way of showing the patterns between 2 categories of data, less good at showing fine differences in amounts

Popular Data Visualisation Libraries in Python

1. Matplotlib

2. Seaborn

3. Pandas Visualisation:

    ○ Built on top of Matplotlib library

    ○ Allows us to create visual representations of DataFrames and Series much quicker and easier way

4. Plotly:

    ○ Used for creating interactive and multidimensional plots

    ○ Advanced and high-level interface

- A multi-platform data visualisation library built on top of NumPy arrays

- Allows us to create high-quality graphics with a range of graphs

- In matplotlib, **pyplot** is used to create figures and change the characteristics of figures

- matplotlib.pyplot is a collection of functions that make matplotlib work like MATLAB

- Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

Anatomy of a figure
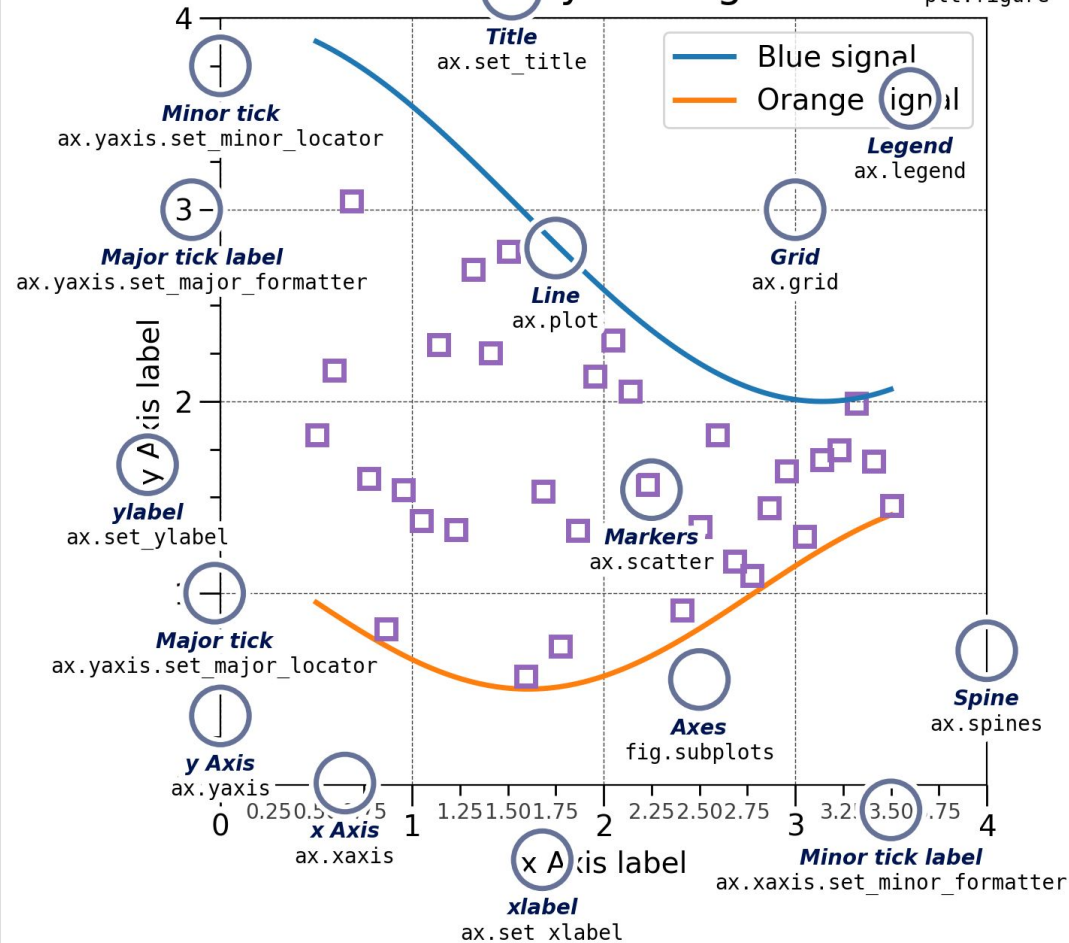
Understanding basics of Matplotlib:

- **Figure**: The entire area where everything is being drawn. It can contain multiple plots with axes, legends, a range of axes, grid, plot-title, etc.

- **Axes**: The area under the figure where the plot is being constructed (or the area your plot appears in) is known as axes. There can be multiple axes in a single figure.

- **Axis**: Number line representing the range of values for the plot. More than two axis can be seen in a multi-dimensional graph.

- **Plot title**: The title is positioned in the center above the axes, giving an overview of the plot.

- Library for statistical graphics plotting in Python

- Provides many default styles and colour palettes

- Built on top of Matplotlib library

- Closely integrated to data structures from Pandas

- The lines of code required are reduced to a very great extent
  (as compared to matplotlib).

Univariate Analysis

- Analysis of only one variable.

- The most common univariate analysis is checking the central tendency (mean, median and mode), the range, the maximum and minimum values, and standard deviation of a variable

- Common visual technique used for univariate analysis is a histogram, which is a frequency distribution graph

- You could also use a box plot or violin plot to compare the spread of the variables and provides an insight into outliers

Bivariate Analysis

- Comparing two variables to study their relationships. These variables could be dependent or independent to each other

- The most common visual technique for bivariate analysis is a scatter plot

Multivariate Analysis

- Multivariate analysis looks at more than two variables and their relationship

- For three variables, you can create a 3-D model to study the relationship

print('Thank You')

Thank You