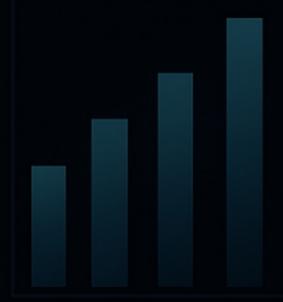
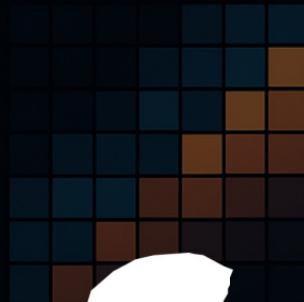


# Data Visualization in the Age of AI

Empowering data points to tell a story

Lamerck Kavuma



```
import matplotlib.pyplot as plt
import seaborn as sns
x = [1, 2, 3, 4]
y = [10, 15, 13, 17]
plt.plot(x, y)
plt.xlabel("x")
fig, axs = plt.subplots(1, 2)
sns.heatmap(data, ax=axs[0])
plt.bar(x, y)
```

```
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 plt.plot(x, y)
5 plt.bar(x, y)
```

# Lameck Kavuma

Data Scientist & ML Engineer | Founder | Cofounder | COO



## Cofounder, COO, Team Lead: AI-Studio Uganda

**Cofounded AISU** - A collective of engineers building AI that matters  
**COO at AISU** - Overseeing the day-to-day company operations  
**Team Lead at AISU** - For the data collection and aggregation team

## Organizer, Mentor: PyData Kampala

Coordinating events and mentorship programs, including personally mentoring others, to foster skill development, knowledge sharing, and career growth in the data science space

## Project Manager: Delta Rising Foundation

Manage product lifecycles through ensuring effective resource allocation, consistent stakeholder communication, and cross-functional team alignment with Scrum

## Founder: Mellon Digital Partners

Lead the development and delivery of innovative technology solutions, focusing on digital transformation and AI-driven services to empower businesses and communities in Uganda.

# The Importance of Data Visualization

Understanding why effective visualization is crucial

## Big Data Era

As the volume of data increases, data visualisation offers a quick and efficient way to summarize the data story.



## Attention spans are dwindling

There's an information overload, data visualization offers a way to capture attention better than individual data points.



## The brain loves graphs

Visuals activate neural pathways better than data points, increasing the chances of memorization.



## Visuals contextualize

Well-constructed visuals show relationships between data points, giving more value to data points beyond their individual value



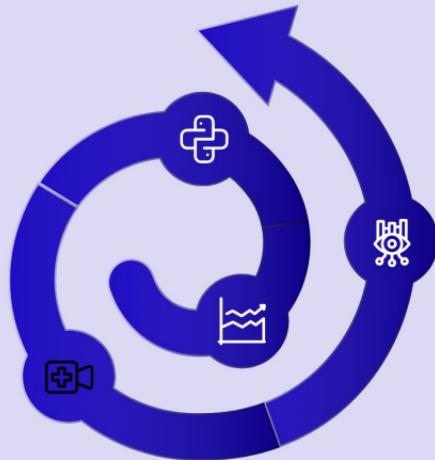
## Visualization for EDA

Visualization serves as a cognitive aid that helps us understand and analyze data to identify gaps, alignment, positioning, and outliers.



# The logic in every visualization

Bridging raw data and meaningful insights through effective storytelling.



## Data points create events.

Every data point tells a part of the story, representing specific events that contribute to the larger narrative.

## Transform raw data into plots.

Through different techniques, raw data points are converted into plots that make data easier to understand.

## Telling the story.

Raw data points have a coherent story, highlighting trends and patterns that could otherwise go unnoticed.

## Identify gaps and outliers.

Easier to spot gaps and outliers in the data which reveals more in the story such as

# Visual Toolbox in Python

Matplotlib vs Seaborn for Data Visualization



## Matplotlib for all generic plots

A powerful library for creating static, animated, and interactive visualizations in



## Seaborn for aesthetic customizations

Built on Matplotlib, Seaborn offers a higher-level interface for drawing attractive statistical graphics.



## Plotly for interactive visuals

An open-source graphing library that allows you to create interactive plots and

# Matplotlib for Data Visualization

Crafts customizable charts, but it can be tricky for beginners and less interactive than the alternatives.

1



2



3



## A customizable library

Matplotlib allows users to tailor their plots according to specific preferences and needs, making it versatile for various applications.

## Figures and axes

The foundational elements of any plot, figures are the overall container, while axes define the area for data representation.

## Adding labels and legends

Labels help in identifying axes and data points, while legends provide clarity on what different plot elements represent.

# Seaborn: Your Go-To for Stunning Plots

A user-friendly but less flexible tool that creates beautiful visualizations with minimal code

1

## **Built on Matplotlib's foundation**

Utilizes Matplotlib, enabling smoother integration.

2

## **High-level plotting interface**

Less code, more beauty.

3

## **Elegant themes and palettes**

Numerous pre-set themes and color palettes that make visualizations stand out effortlessly.

4

## **Ideal for statistical visuals**

Designed specifically for statistical visualizations.

# Choosing the Right Visualization Tool

Understanding when to use Matplotlib or Seaborn for your data visualizations.

## Custom control: Matplotlib



When you need **extensive customization**, Matplotlib shines with its flexibility, while Seaborn offers a more limited set of options.

## Quick plots: Seaborn



For **fast and efficient** plotting, Seaborn allows you to create visualizations rapidly without much boilerplate code, unlike the more verbose Matplotlib.

## Aesthetic appeal: Seaborn



If you want **great-looking** plots right out of the box, Seaborn is the way to go; Matplotlib can achieve the same but requires more effort.

# Essential Plot Types for Visualization

Not every plot suits every dataset or context. Choosing the right visualization depends on the data type, audience, and the story you want to tell.

## ● Line Plot: Trends over time

A **line plot** is perfect for visualizing trends over time, showing how values change sequentially. It's a go-to for both Matplotlib and Seaborn users who want to track progress or changes.

## ● Bar Plot: Categorical comparison

**Bar plots** allow for easy comparison of different categories, making it simple to see differences in size or amount. This type works well in both libraries for presenting categorical data effectively.

## ● Histogram: Distribution insights

A **histogram** displays the distribution of data points across different bins, revealing patterns in data frequency. It's particularly useful in Seaborn for understanding how data is distributed.

## ● Box Plot: Spread & outliers

The **box plot** is a fantastic tool for visualizing the spread of data and identifying outliers. It provides a clear summary of data variability, which is easily achievable with Seaborn.

## ● Heatmap: Matrix relationships

**Heatmaps** visualize relationships in matrix data, highlighting areas of interest through color gradients. Seaborn excels at creating these informative visualizations to uncover correlations.

## ● Scatter Plot: Two-variable relationships

A **scatter plot** helps illustrate relationships between two variables, showcasing potential correlations. It's a versatile option available in both Matplotlib and Seaborn.

# Key considerations

Creating effective visuals with contrast and accessibility in mind

## 1 Use contrast wisely

Effective visuals leverage contrast to enhance readability and draw attention to key data points, making them stand out clearly.

## 2 Label clearly

Clear labeling ensures that your audience understands the data presented, minimizing confusion and enhancing communication.

## 3 Remove unnecessary gridlines

Eliminating clutter like gridlines helps focus on the data itself, making the visualization cleaner and easier to interpret.

## 4 Choose colors for accessibility

Select colors that are accessible to all viewers, including those with color vision deficiencies, to ensure everyone can interpret the data accurately.

# What do you want to visualize?

