**Minor in AI – Batch 04**

**03 March 2025**

**Title: Mastering Data Visualization with Matplotlib: From Basics to Stunning Plots**

Topics:

* Customize axes, titles, and labels to enhance readability
* Work with Different Plot Types
* Create bar charts (plt.bar()), scatter plots (plt.scatter()), and histograms (plt.hist()).
* Differentiate between various chart types and their appropriate use cases.

**Note:**

Below are codes that we types together. For others, refer to the collab link.

Why that is for daily\_steps CSV file, a line plot not a good idea?

import pandas as pd

import matplotlib.pyplot as plt

# Read CSV file

df = pd.read\_csv("daily\_steps.csv")

# Create the line plot

plt.plot(df['Person\_ID'], df['Daily\_Steps'])

# Add labels and title

plt.xlabel("Person ID")

plt.ylabel("Daily Steps")

plt.title("Daily Steps of Person")

# Display the plot

plt.show()

Instead we can go for a histogram.

import pandas as pd

import matplotlib.pyplot as plt

# Read CSV file

df = pd.read\_csv("daily\_steps.csv")

# Plot histogram

plt.hist(df["Daily\_Steps"], bins=10, edgecolor="black")

plt.xlabel("Daily Steps")

plt.ylabel("Frequency")

plt.title("Distribution of Daily Steps Among People")

plt.show()

Understanding the customizations

import matplotlib.pyplot as plt

# Sample data

x = [1, 2, 3, 4, 5]

y1 = [2, 4, 6, 8, 10]  # Line 1

y2 = [1, 3, 5, 7, 9]   # Line 2

# Set figure size

plt.figure(figsize=(8, 5))

# Plot with different styles

plt.plot(x, y1, color="blue", marker="o", markersize=8, linestyle="--", linewidth=2, label="Dashed Blue")

plt.plot(x, y2, color="red", marker="s", markersize=10, linestyle="-.", linewidth=2, label="Dash-dot Red")

# Labels and title

plt.xlabel("X-axis Label", fontsize=12, color="darkblue")

plt.ylabel("Y-axis Label", fontsize=12, color="darkred")

plt.title("Matplotlib Customization Demo", fontsize=14, fontweight="bold")

# Grid and legend

plt.grid(True, linestyle="--", alpha=0.5)

plt.legend()

# Show plot

plt.show()

Scatter Plots:

import pandas as pd

import matplotlib.pyplot as plt

# Sample Data

df = pd.read\_csv("daily\_steps.csv")

df["Calories\_Burned"] = df["Daily\_Steps"] \* 0.04  # Assuming 0.04 calories per step

# Scatter Plot

plt.scatter(df["Daily\_Steps"], df["Calories\_Burned"])

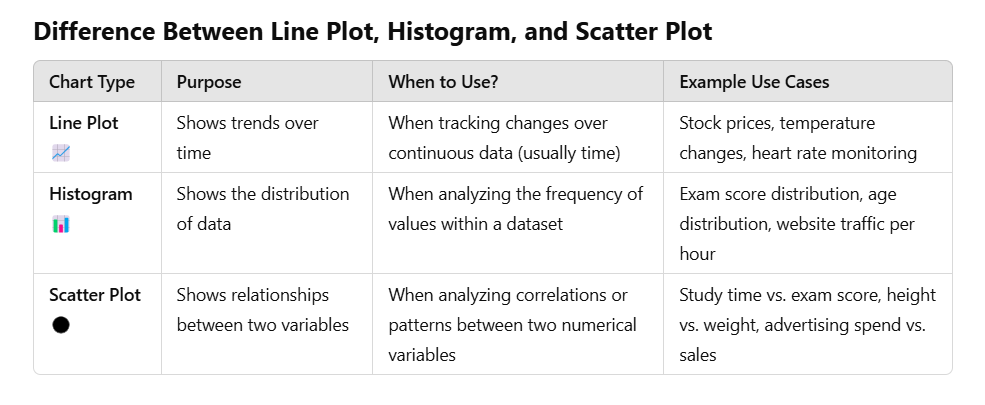
plt.xlabel("Daily Steps")

plt.ylabel("Calories Burned")

plt.title("Daily Steps vs. Calories Burned")

plt.show()

Difference Table: (Source: ChatGPT)



Other Case Study used: (Interpreting the Mumbai local train lines)

