**Plot a Pie chart for 5 features of Diabetes detection with at least 5 attributes of Pie chart.**

import matplotlib.pyplot as plt

# Data for the pie chart

features = ['Glucose', 'BMI', 'Age', 'Blood Pressure', 'Insulin']

values = [30, 25, 15, 20, 10] # These could represent feature importance or mean values

colors = ['red', 'blue', 'green', 'orange', 'purple']

explode = (0.1, 0, 0, 0, 0) # "explode" the first slice (Glucose) for emphasis

# Create the pie chart

plt.figure(figsize=(8, 6))

plt.pie(values, labels=features, colors=colors, explode=explode,

autopct='%1.1f%%', shadow=True, startangle=140)

# Add title and display the chart

plt.title("Feature Distribution for Diabetes Detection")

plt.axis('equal') # Equal aspect ratio ensures the pie is circular

plt.show()

**Give implementation of Histogram using matplotlib.**

import matplotlib.pyplot as plt

import numpy as np

# Generate some random data (e.g., exam scores)

data = np.random.normal(loc=70, scale=10, size=100)

# Create the histogram

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

plt.hist(data, bins=10, color='skyblue', edgecolor='black')

# Add titles and labels

plt.title("Histogram of Exam Scores")

plt.xlabel("Score")

plt.ylabel("Number of Students")

# Show the plot

plt.grid(True)

plt.show()

**Give implementation of Pie chart using matplotlib with 5 attributes.**

import matplotlib.pyplot as plt

# Data for the pie chart

labels = ['Apples', 'Bananas', 'Cherries', 'Dates', 'Elderberries']

sizes = [20, 25, 15, 30, 10] # Percentages or proportions

colors = ['red', 'yellow', 'pink', 'brown', 'purple']

explode = (0, 0.1, 0, 0, 0) # "explode" the 2nd slice (Bananas)

# Create the pie chart

plt.figure(figsize=(7, 7))

plt.pie(sizes, labels=labels, colors=colors, explode=explode,

autopct='%1.1f%%', shadow=True, startangle=140)

# Title and display

plt.title("Fruit Distribution")

plt.axis('equal') # Equal aspect ratio ensures the pie is circular.

plt.show()

**Display Bar chart with at least 5 different widths and colours.**

import matplotlib.pyplot as plt

# Sample data

categories = ['A', 'B', 'C', 'D', 'E']

values = [10, 15, 7, 12, 9]

colors = ['red', 'green', 'blue', 'orange', 'purple']

widths = [0.3, 0.6, 0.4, 0.7, 0.5]

# X-axis positions for bars

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

# Plotting the bars with individual widths and colors

for i in range(len(categories)):

plt.bar(positions[i], values[i], width=widths[i], color=colors[i], label=categories[i])

# Labels and title

plt.title("Bar Chart with Different Widths and Colors")

plt.xlabel("Category")

plt.ylabel("Value")

plt.xticks(positions, categories)

plt.legend(title="Categories")

plt.grid(axis='y', linestyle='--', alpha=0.7)

# Display the chart

plt.show()

**Implement 5 types of markers and 2 markeredgecolor using matplotlib.**

import matplotlib.pyplot as plt

# Sample x and y data points

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

y = [10, 15, 7, 12, 9]

# Marker styles and edge colors

markers = ['o', 's', '^', 'D', '\*'] # Circle, Square, Triangle, Diamond, Star

colors = ['blue', 'green', 'red', 'purple', 'orange']

edge\_colors = ['black', 'gray'] # Two edge colors

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

# Plot each point with a different marker and edge color (alternating)

for i in range(len(x)):

plt.plot(x[i], y[i],

marker=markers[i],

color=colors[i],

markersize=12,

markeredgewidth=2,

markeredgecolor=edge\_colors[i % 2],

label=f'Marker {markers[i]}')

# Add legend and labels

plt.title("Markers with Different Shapes and Edge Colors")

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

plt.legend()

plt.grid(True)

plt.show()

**Display 3 Pie charts as follows:**

**1. 4 features with startangle= 80**

**2. 5 features with startangle= 160**

**3. 4 features with startangle=120**

import matplotlib.pyplot as plt

# Data for the pie charts

labels1 = ['A', 'B', 'C', 'D']

sizes1 = [15, 25, 35, 25]

labels2 = ['Apple', 'Banana', 'Cherry', 'Date', 'Elderberry']

sizes2 = [20, 30, 10, 25, 15]

labels3 = ['North', 'South', 'East', 'West']

sizes3 = [40, 20, 30, 10]

# Pie Chart 1: 4 features, startangle=80

plt.figure(figsize=(10, 8))

plt.subplot(131) # 1st pie chart

plt.pie(sizes1, labels=labels1, startangle=80, autopct='%1.1f%%', shadow=True, colors=['red', 'blue', 'green', 'yellow'])

plt.title("Pie Chart 1 - Startangle 80")

# Pie Chart 2: 5 features, startangle=160

plt.subplot(132) # 2nd pie chart

plt.pie(sizes2, labels=labels2, startangle=160, autopct='%1.1f%%', shadow=True, colors=['pink', 'purple', 'orange', 'brown', 'cyan'])

plt.title("Pie Chart 2 - Startangle 160")

# Pie Chart 3: 4 features, startangle=120

plt.subplot(133) # 3rd pie chart

plt.pie(sizes3, labels=labels3, startangle=120, autopct='%1.1f%%', shadow=True, colors=['magenta', 'lightgreen', 'lightblue', 'gray'])

plt.title("Pie Chart 3 - Startangle 120")

# Display all pie charts

plt.tight\_layout()

plt.show()

**Plot a Scatter plot for Employee’s age and Salary. (Minim 7 records)**

import matplotlib.pyplot as plt

# Sample data for Employees (Age and Salary)

ages = [25, 30, 35, 40, 45, 50, 55]

salaries = [30000, 35000, 40000, 45000, 50000, 55000, 60000]

# Plotting the scatter plot

plt.figure(figsize=(8, 6))

plt.scatter(ages, salaries, color='blue', marker='o')

# Adding labels and title

plt.title("Employee Age vs Salary")

plt.xlabel("Age (Years)")

plt.ylabel("Salary ($)")

# Show the plot

plt.grid(True)

plt.show()

**Give implementation of Scatter plot using matplotlib.**

import matplotlib.pyplot as plt

# Sample data for X and Y axes

x = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

y = [10, 12, 13, 15, 18, 20, 22, 25, 28, 30]

# Creating the scatter plot

plt.figure(figsize=(8, 6))

plt.scatter(x, y, color='blue', marker='o') # Blue color, circle markers

# Adding title and labels

plt.title("Scatter Plot: X vs Y")

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

# Display the plot

plt.grid(True)

plt.show()

**Give implementation of Histogram and Scatter plot using matplotlib.**

import matplotlib.pyplot as plt

import numpy as np

# --- 1. Histogram Implementation ---

# Sample data: Normally distributed data (e.g., exam scores)

data = np.random.normal(loc=75, scale=10, size=100)

# Create the histogram

plt.figure(figsize=(12, 6))

# Subplot 1: Histogram

plt.subplot(121) # 1st plot (1 row, 2 columns, 1st plot)

plt.hist(data, bins=10, color='skyblue', edgecolor='black')

plt.title("Histogram of Exam Scores")

plt.xlabel("Score")

plt.ylabel("Frequency")

plt.grid(True)

# --- 2. Scatter Plot Implementation ---

# Sample data for Scatter plot (e.g., Age vs Salary)

x = np.array([25, 30, 35, 40, 45, 50, 55, 60, 65, 70])

y = np.array([30000, 35000, 40000, 45000, 50000, 55000, 60000, 65000, 70000, 75000])

# Subplot 2: Scatter plot

plt.subplot(122) # 2nd plot (1 row, 2 columns, 2nd plot)

plt.scatter(x, y, color='orange', marker='o')

plt.title("Scatter Plot: Age vs Salary")

plt.xlabel("Age (Years)")

plt.ylabel("Salary ($)")

plt.grid(True)

# Display both plots

plt.tight\_layout() # Adjust subplots to fit in the figure area

plt.show()

**Display Students roll number and marks as X axis(roll number) , Y axis (marks) using matplotlib (bar).**

import matplotlib.pyplot as plt

# Sample data for students' roll numbers and marks

roll\_numbers = [101, 102, 103, 104, 105]

marks = [85, 92, 78, 88, 95]

# Creating the bar chart

plt.figure(figsize=(8, 6))

plt.bar(roll\_numbers, marks, color='blue')

# Adding title and labels

plt.title("Students' Marks")

plt.xlabel("Roll Number")

plt.ylabel("Marks")

# Display the plot

plt.show()

**Give implementation of Bar chart using matplotlib.**

import matplotlib.pyplot as plt

# Sample data for the bar chart

categories = ['A', 'B', 'C', 'D', 'E']

values = [23, 45, 56, 78, 33]

# Create the bar chart

plt.figure(figsize=(8, 6))

plt.bar(categories, values, color='green')

# Adding title and labels

plt.title("Bar Chart Example")

plt.xlabel("Categories")

plt.ylabel("Values")

# Display the plot

plt.show()

**Plot a Histogram with 0 to 150 bins for age of 20 people.**

import matplotlib.pyplot as plt

import numpy as np

# Sample data: Ages of 20 people

ages = [23, 45, 34, 23, 25, 33, 41, 56, 60, 20, 18, 28, 38, 40, 47, 50, 26, 31, 42, 54]

# Create the histogram

plt.figure(figsize=(8, 6))

plt.hist(ages, bins=np.arange(0, 151, 10), color='skyblue', edgecolor='black')

# Adding title and labels

plt.title("Histogram of Ages (0 to 150 bins)")

plt.xlabel("Age")

plt.ylabel("Frequency")

# Display the plot

plt.grid(True)

plt.show()