Data Visualization Laboratory BCS358D

Experiment No. 8

1BY22IS087- MEGHANA PURUMU

1BY22IS088- MIT D PATWA

1BY22IS089 - AAMIR

QUESTION

Write a python program to explain working with bokeh line graph using Annotations and Legends.

a) Write a python program for plotting different types of plots using bokeh.

Bokeh Library

Bokeh : Bokeh is a data visualization library in Python that provides high-performance interactive charts and plots.

Annotations: Annotations are used to highlight or label specific features on a plot. Some common types of annotations in Bokeh include :

- Text Annotations
- Arrow Annotations
- Line Annotations

Legends

Legends are essential when you have multiple data series on a single plot. They help users identify and differentiate between different elements in the plot. Bokeh allows you to easily add legends to your plots.

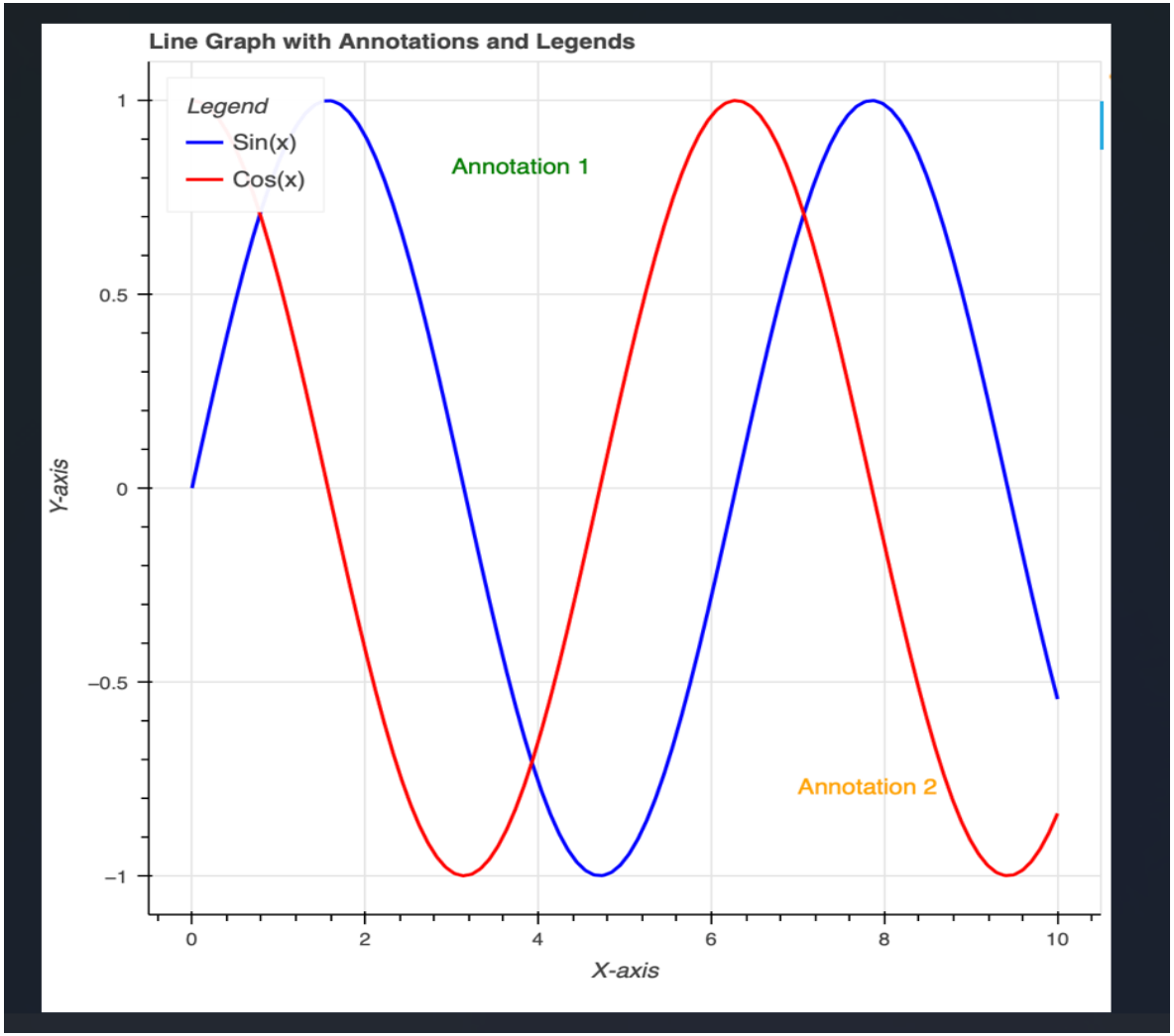
CODE

```
a) from bokeh.plotting import figure, show
from bokeh.models import Label
# Importing NumPy for generating random data
import numpy as np
# Generate some random data
x_values = np.linspace(0, 10, 100)
y_values1 = np.sin(x_values)
y_values2 = np.cos(x_values)
# Create a Bokeh figure
p = figure(title='Line Graph with Annotations and Legends',
x_axis_label='X-axis',          y_axis_label='Y-axis')
# Plot the first line
line1 = p.line(x_values, y_values1, line_width=2, line_color='blue',
legend_label='Sin(x)')
```

CODE

```
a) # Plot the second line

line2 = p.line(x_values, y_values2, line_width=2, line_color='red', legend_label='Cos(x)’)

# Add annotations

annotation1 = Label(x=3, y=0.8, text='Annotation 1', text_color='green', text_font_size='10pt’)

annotation2 = Label(x=7, y=-0.8, text='Annotation 2', text_color='orange', text_font_size='10pt’)

p.add_layout(annotation1)

p.add_layout(annotation2)

# Add a legend

p.legend.location = 'top_left’

p.legend.title = 'Legend’

p.legend.label_text_font_size = '10pt’

# Show the plot     show(p)
```

//Output


Line Graph with Annotations and Legends
Legend
Sin(x)
Cos(x)
Annotation 1
Annotation 2
Y-axis
X-axis
1
0.5
0
−0.5
−1
0
2
4
6
8
10

CODE

```
from bokeh.plotting import figure, show
from bokeh.io import output_file
from bokeh.models import Label
from bokeh.layouts import gridplot
import numpy as np
# Create a Bokeh output fileoutput_file("bokeh_plots.html")
# Generate some sample data
x_values = np.linspace(0, 10, 100)
y_values = np.sin(x_values)
# Create a Bokeh figure for Scatter Plot
scatter_plot = figure(title='Scatter Plot', x_axis_label='X-axis', y_axis_label='Y-axis')
scatter_plot.circle(x_values, y_values, size=8, color='green', alpha=0.6)
```

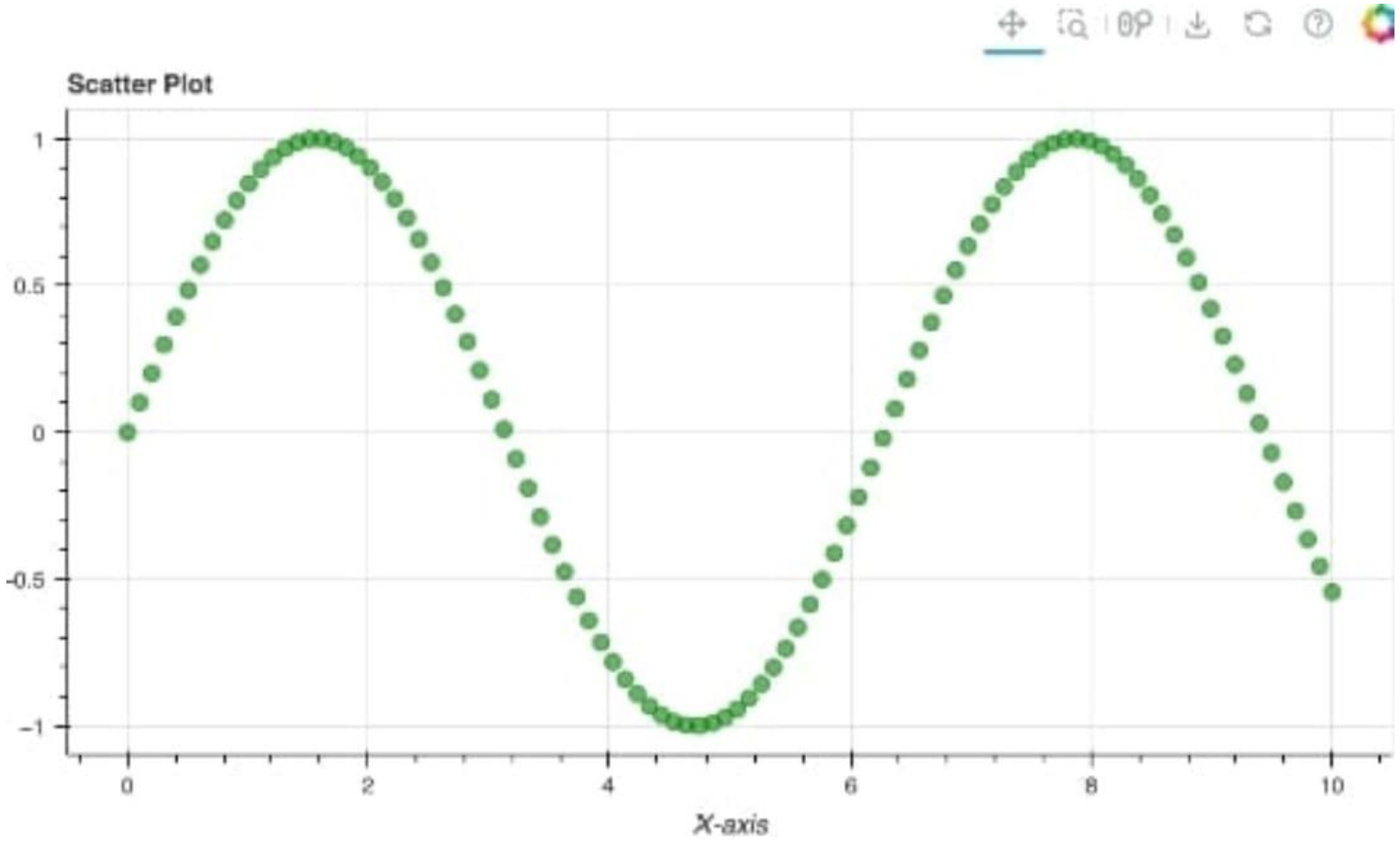
CODE

```
# Create a Bokeh figure for Bar Plot
categories = ['Category A', 'Category B', 'Category C’]
bar_plot = figure(x_range=categories, title='Bar Plot',
x_axis_label='Categories',y_axis_label='Values’)
bar_plot.vbar(x=categories, top=[3, 5, 2], width=0.8, color='orange’)


# Create a Bokeh figure for Histogram
histogram_plot = figure(title='Histogram', x_axis_label='Values',
y_axis_label='Frequency’)
histogram_plot.quad(top=np.histogram(y_values, bins=20)[0], bottom=0,
left=np.histogram(y_values, bins=20)[1][:-1], right=np.histogram(y_values,
bins=20)[1][1:], fill_color="purple", line_color="black")
```

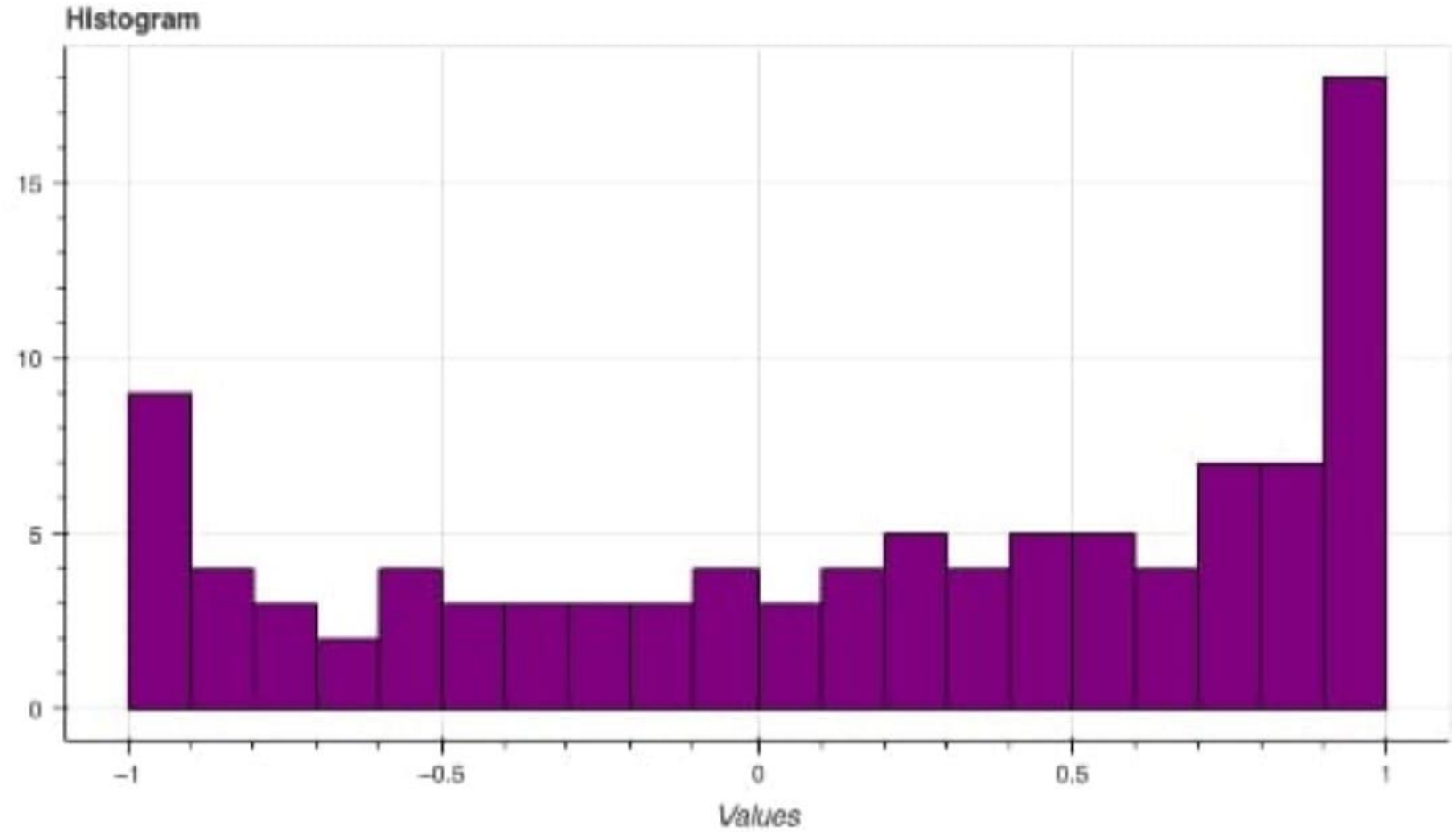
CODE

```
# Add annotations
annotation = Label(x=7, y=0.8, text='Annotation', text_color = 'red', text_font_size = '10pt')line_plot.add_layout(annotation)
# Combine plots into a grid layout
plots = [[scatter_plot], [bar_plot, histogram_plot]]
grid = gridplot(plots, sizing_mode='stretch_both’)
# Show the grid layout
show(grid)
```

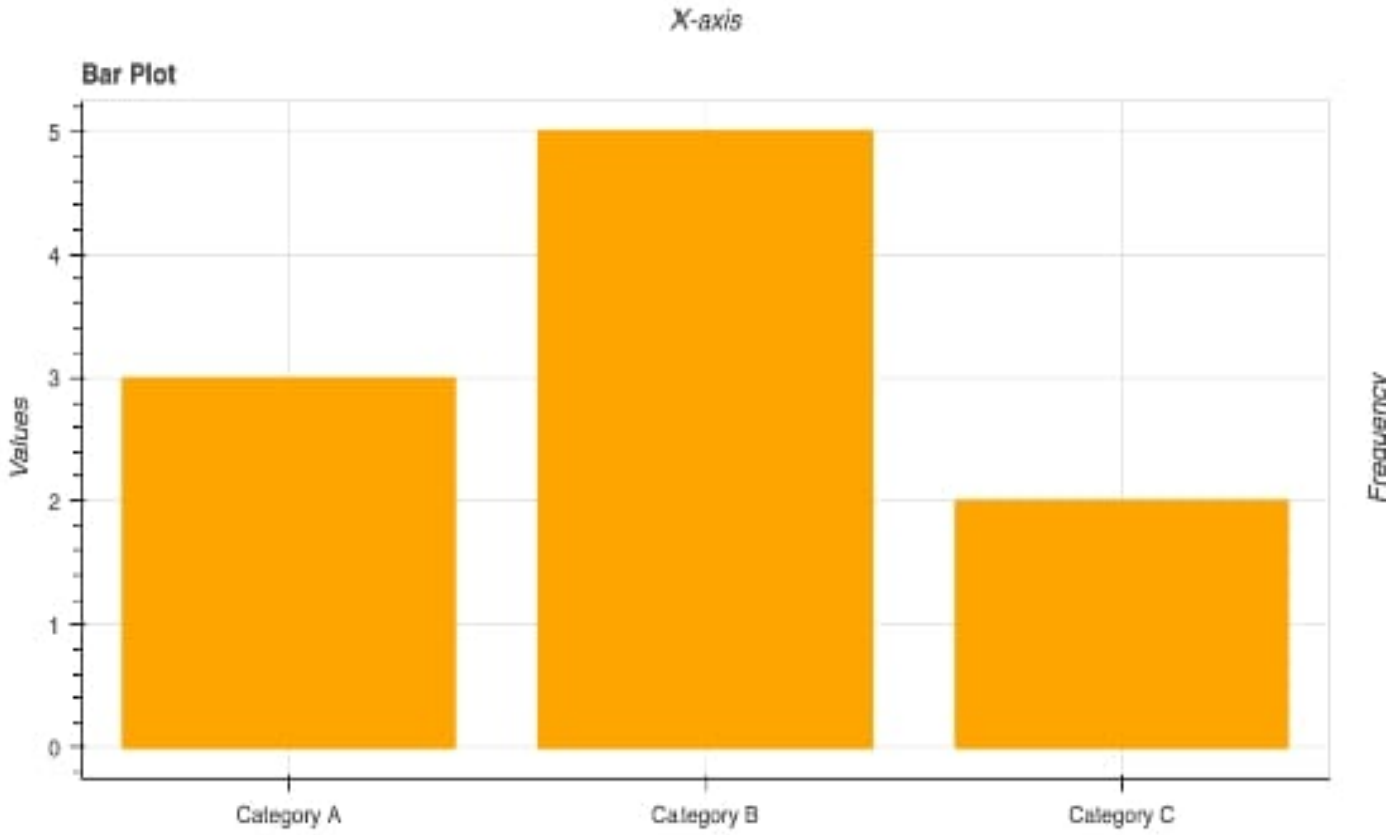
//output



//output



//output


X-axis
Bar Plot
Values
Frequency
5
4
3
2
1
0
Category A
Category B
Category C

APPLICATIONS

- Data Visualization:
- Custom Web Applications
- Network Graphs
- Machine Learning Model Visualization

THANK YOU