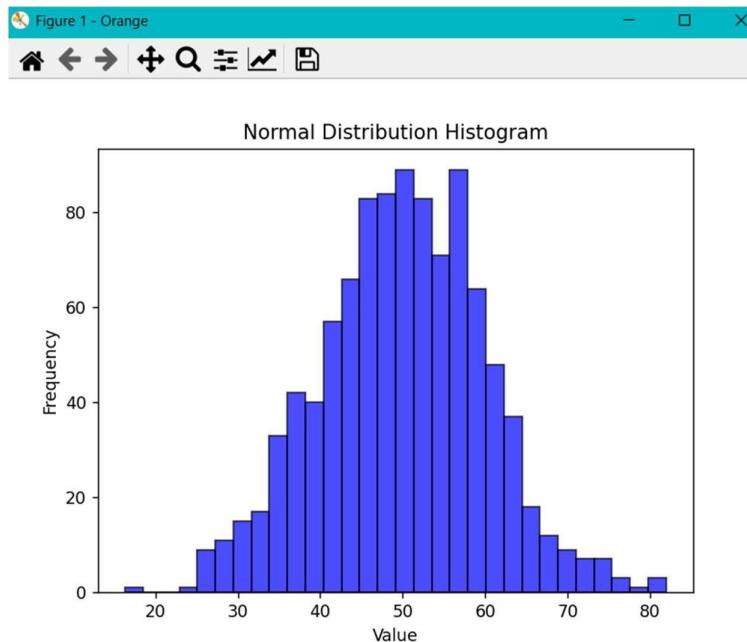


NORMAL DISTRIBUTION HISTOGRAM:

```
import numpy as np
import matplotlib.pyplot as plt
data = np.random.normal(loc=50, scale=10, size=1000)
plt.hist(data, bins=30, color='blue', alpha=0.7,
edgecolor='black') plt.title('Normal Distribution Histogram')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:

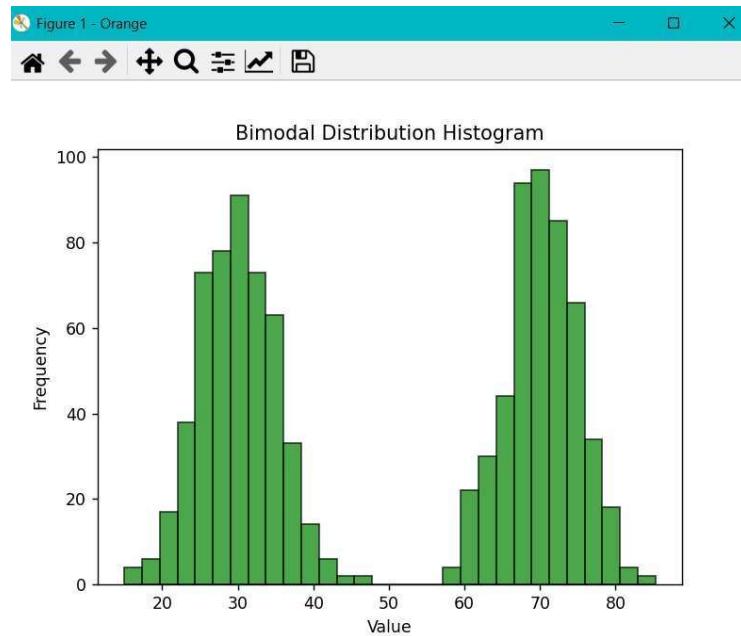


BIMODAL DISTRIBUTION HISTOGRAM:

```
import numpy as np
import matplotlib.pyplot as plt
data1 = np.random.normal(loc=30, scale=5,
size=500) data2 = np.random.normal(loc=70,
scale=5, size=500) data = np.concatenate([data1,
data2])

plt.hist(data, bins=30, color='green', alpha=0.7,
edgecolor='black') plt.title('Bimodal Distribution Histogram')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:

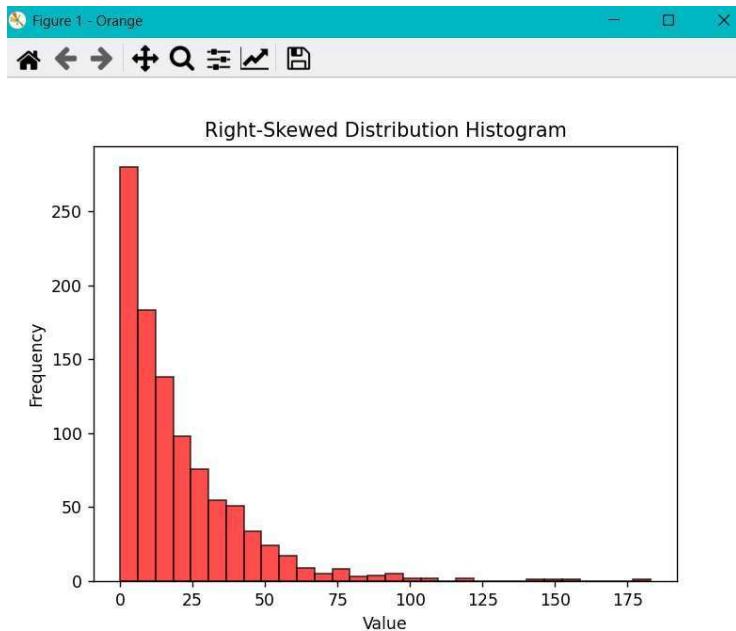


RIGHT-SKewed DISTRIBUTION HISTOGRAM:

```
import numpy as np
import matplotlib.pyplot as plt

data = np.random.exponential(scale=20, size=1000)
plt.hist(data, bins=30, color='red', alpha=0.7,
edgecolor='black') plt.title('Right-Skewed Distribution
Histogram') plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:

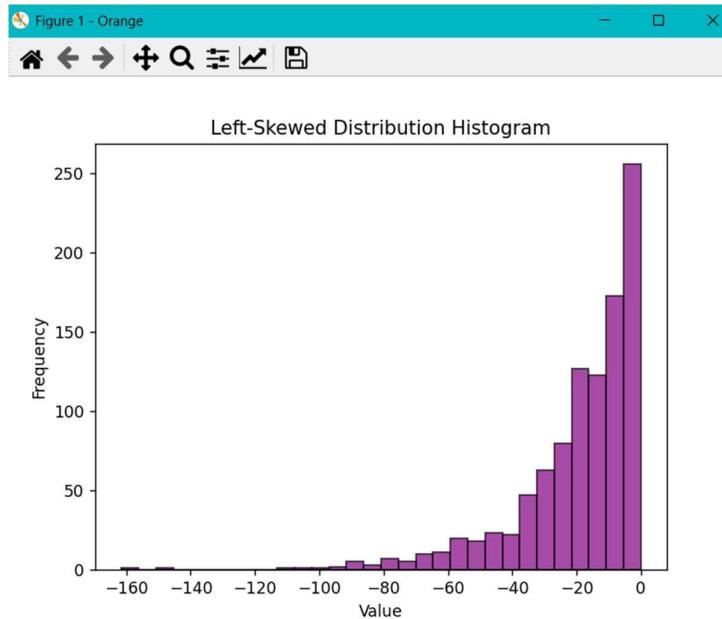


LEFT-SKewed DISTRIBUTION HISTOGRAM:

```
import numpy as np
import matplotlib.pyplot as plt

data = -np.random.exponential(scale=20, size=1000)
plt.hist(data, bins=30, color='purple', alpha=0.7,
edgecolor='black') plt.title('Left-Skewed Distribution
Histogram')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:



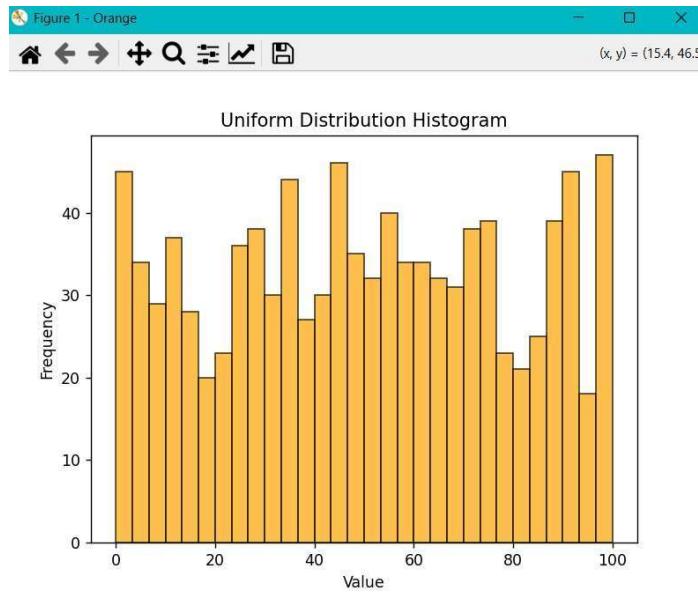
UNIFORM DISTRIBUTION HISTOGRAM:

```
import numpy as np
import matplotlib.pyplot as plt

data = np.random.uniform(low=0, high=100, size=1000)

plt.hist(data, bins=30, color='orange', alpha=0.7,
edgecolor='black') plt.title('Uniform Distribution Histogram')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.show()
```

OUTPUT:



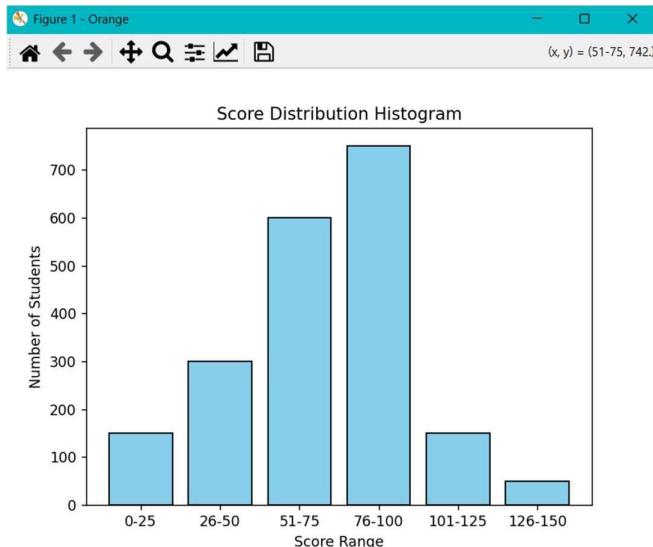
CUSTOM HISTOGRAM OF SCORE FREQUENCIES:

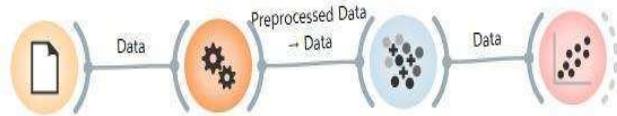
```
import matplotlib.pyplot as plt

score_ranges = ['0-25', '26-50', '51-75', '76-100', '101-125', '126-150']
frequencies = [150, 300, 600, 750, 150, 50]

plt.bar(score_ranges, frequencies, color='skyblue', edgecolor='black')
plt.title('Score Distribution Histogram')
plt.xlabel('Score Range')
plt.ylabel('Number of Students')
plt.show()
```

OUTPUT:





File

Preprocess

k-Means

Scatter Plot

