



**Al-Mustaqbal University**  
**College of Technical Engineering**  
**Department of Electrical Engineering**  
**Techniques**



**Computer Application - UOMU0000032**

**Lecture 5 – Data Plotting**

By

**Dr. Mohammed Fadhil**

**PhD in Computer Networks**

**Email: [mohammed.fadhil1@uomus.edu.iq](mailto:mohammed.fadhil1@uomus.edu.iq)**





# Introduction to Data Plotting

- MATLAB provides powerful functions to visualize data effectively.
- Different types of plots: Line plots, scatter plots, bar charts, histograms, and 3D plots.
- Customizing plots for better readability and presentation.



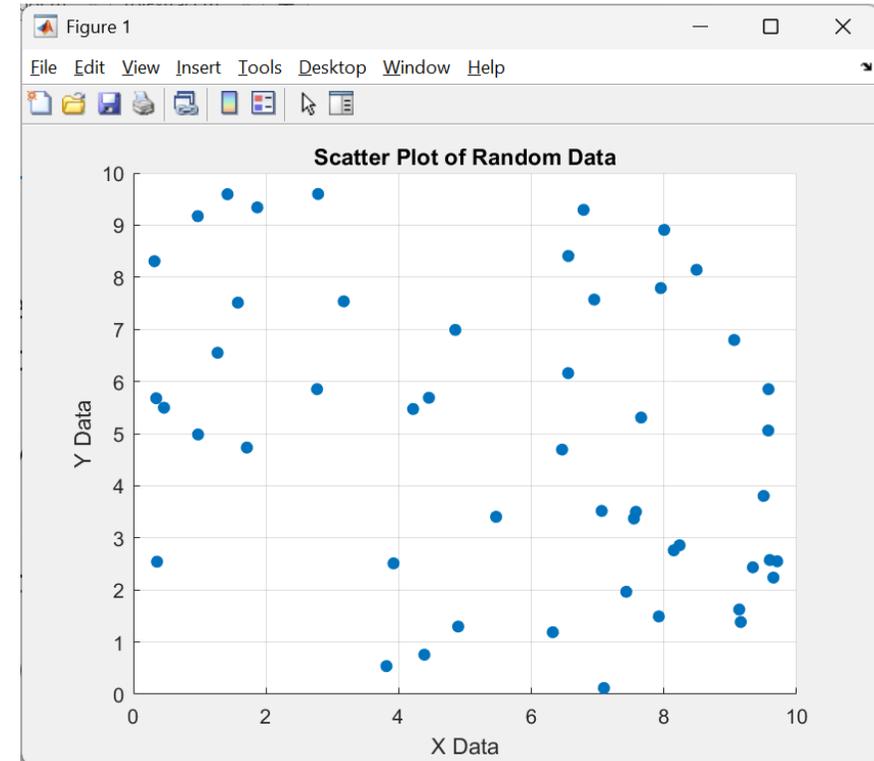
# Scatter Plots for Data Visualisation

- Code Example:

```
x = rand(1, 50) * 10;  
y = rand(1, 50) * 10;  
scatter(x, y, 'filled');  
xlabel('X Data');  
ylabel('Y Data');  
title('Scatter Plot of Random Data');  
grid on;
```

- Notes:

- `scatter(x, y, 'filled')` creates a scatter plot with filled markers.
- Useful for analyzing relationships between variables.



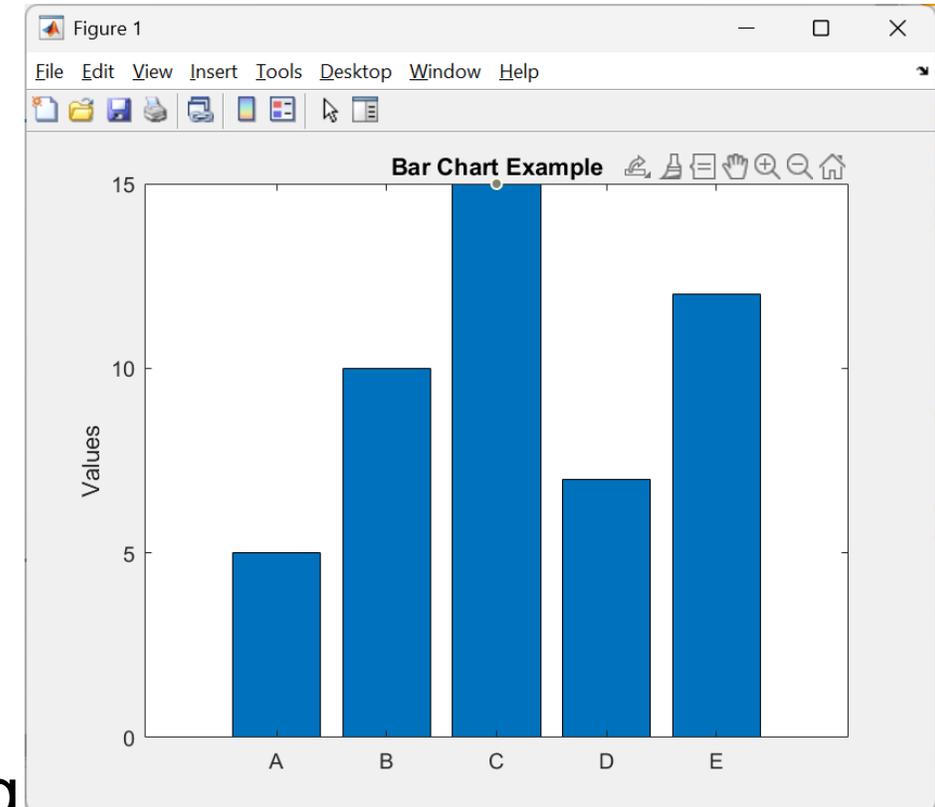
# Bar Charts for Categorical Data

- Code Example:

```
data = [5, 10, 15, 7, 12];  
categories = {'A', 'B', 'C', 'D', 'E'};  
bar(data);  
set(gca, 'XTickLabel', categories);  
ylabel('Values');title('Bar Chart Example');
```

- Notes:

- `bar(data)` creates a bar chart.
- `set(gca, 'XTickLabel', categories)` assigns axis.
- Ideal for representing categorical data.



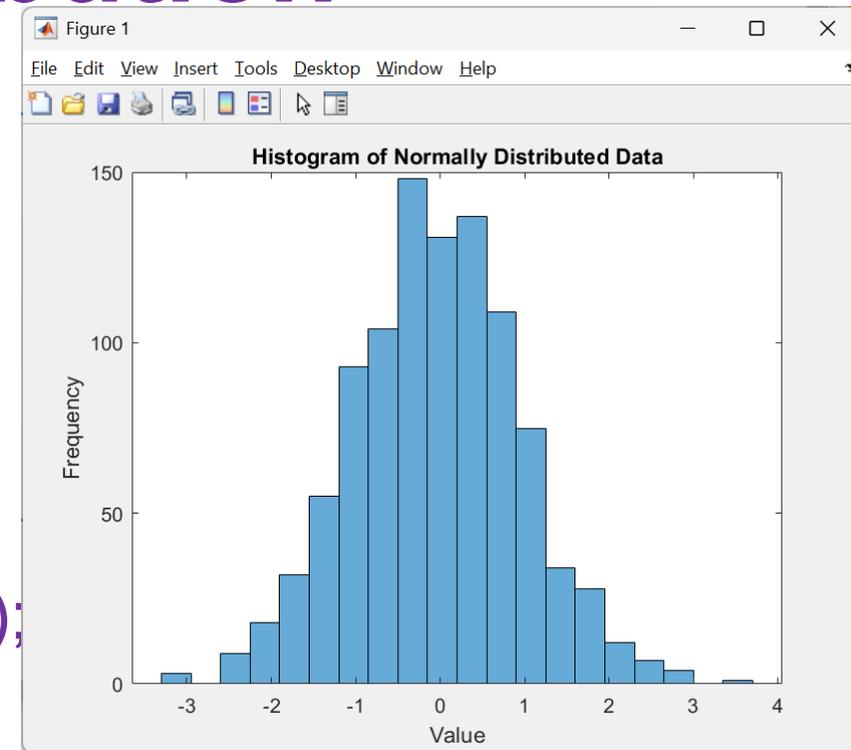
# Histograms for Data Distribution

- Code Example:

```
data = randn(1, 1000);  
histogram(data, 20);  
xlabel('Value');  
ylabel('Frequency');  
title('Histogram of Normally Distributed Data');
```

- Notes:

- histogram(data, 20) creates a histogram with 20 bins.
- Useful for understanding data distribution and patterns.





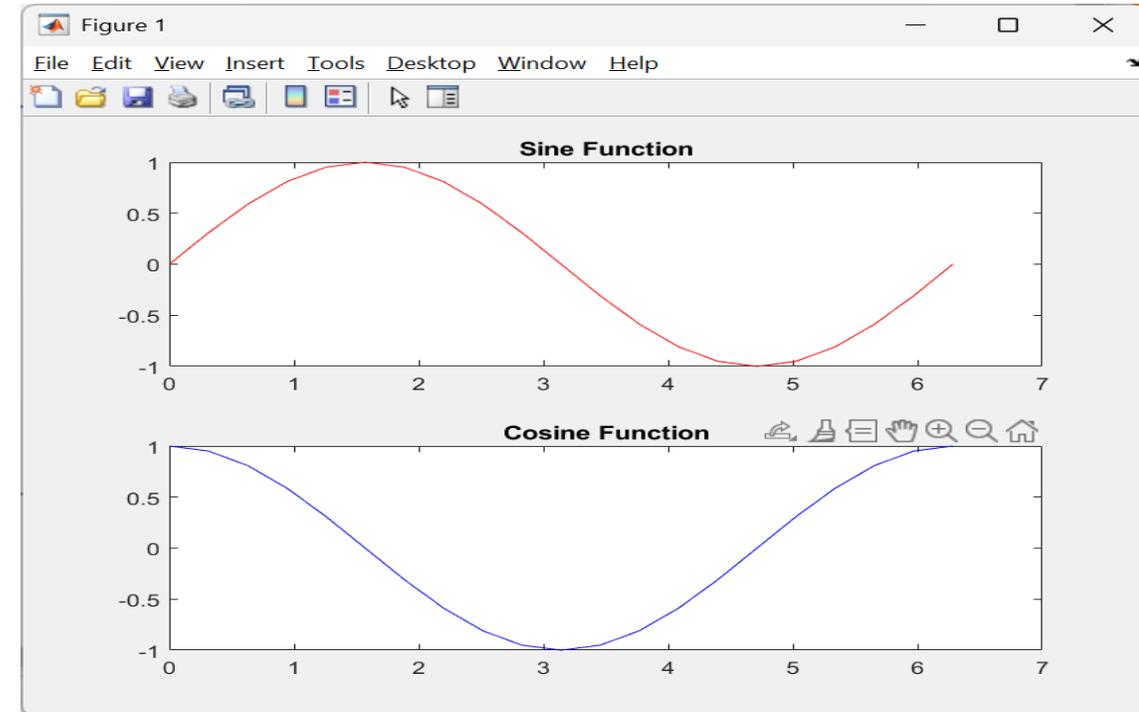
# Multiple Plots in One Figure

- Code Example:

```
t = 0:pi/10:2*pi;  
y1 = sin(t);  
y2 = cos(t);  
subplot(2,1,1);plot(t, y1, 'r');  
title('Sine Function');  
subplot(2,1,2);plot(t, y2, 'b');  
title('Cosine Function');
```

- Notes:

- subplot(m, n, p) divides the figure into m rows and n columns.
- The p value selects which subplot to activate.
- Useful for comparing multiple datasets.





# 3D Plotting Basics

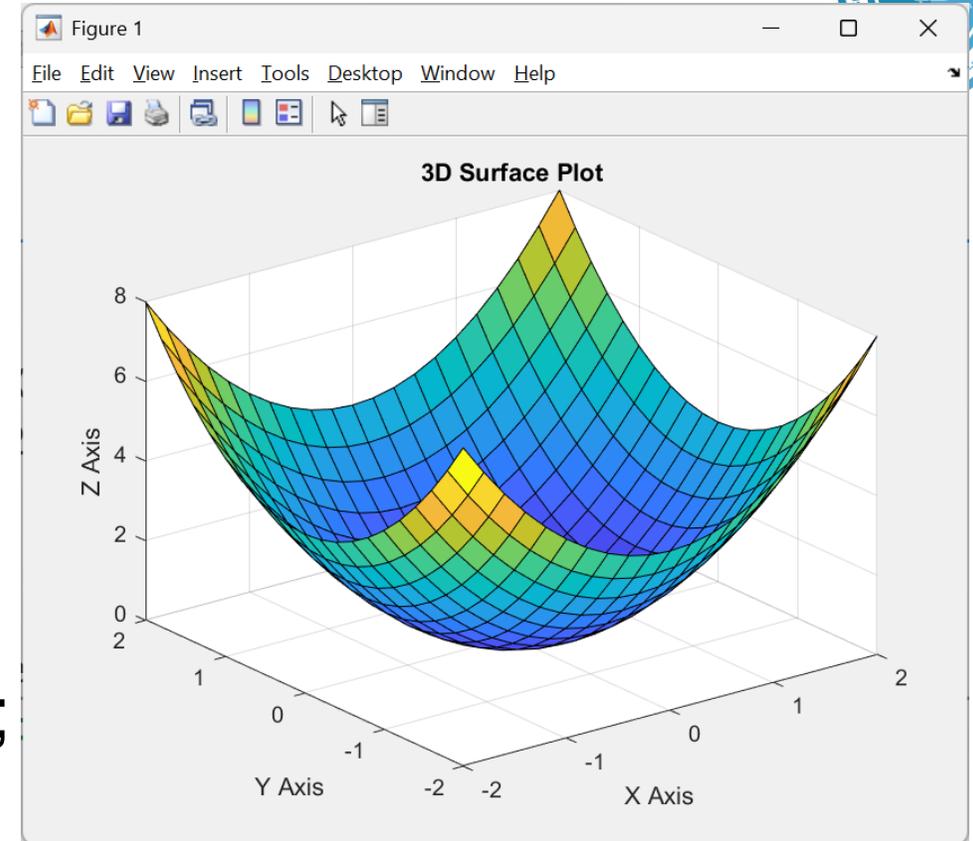


- Code Example:

```
[x, y] = meshgrid(-2:0.2:2, -2:0.2:2);  
z = x.^2 + y.^2;  
surf(x, y, z);  
xlabel('X Axis');ylabel('Y Axis');  
zlabel('Z Axis');title('3D Surface Plot');
```

- Notes:

- meshgrid generates a grid of x and y values.
- surf(x, y, z) creates a 3D surface plot.
- Useful for visualizing mathematical functions in 3D.





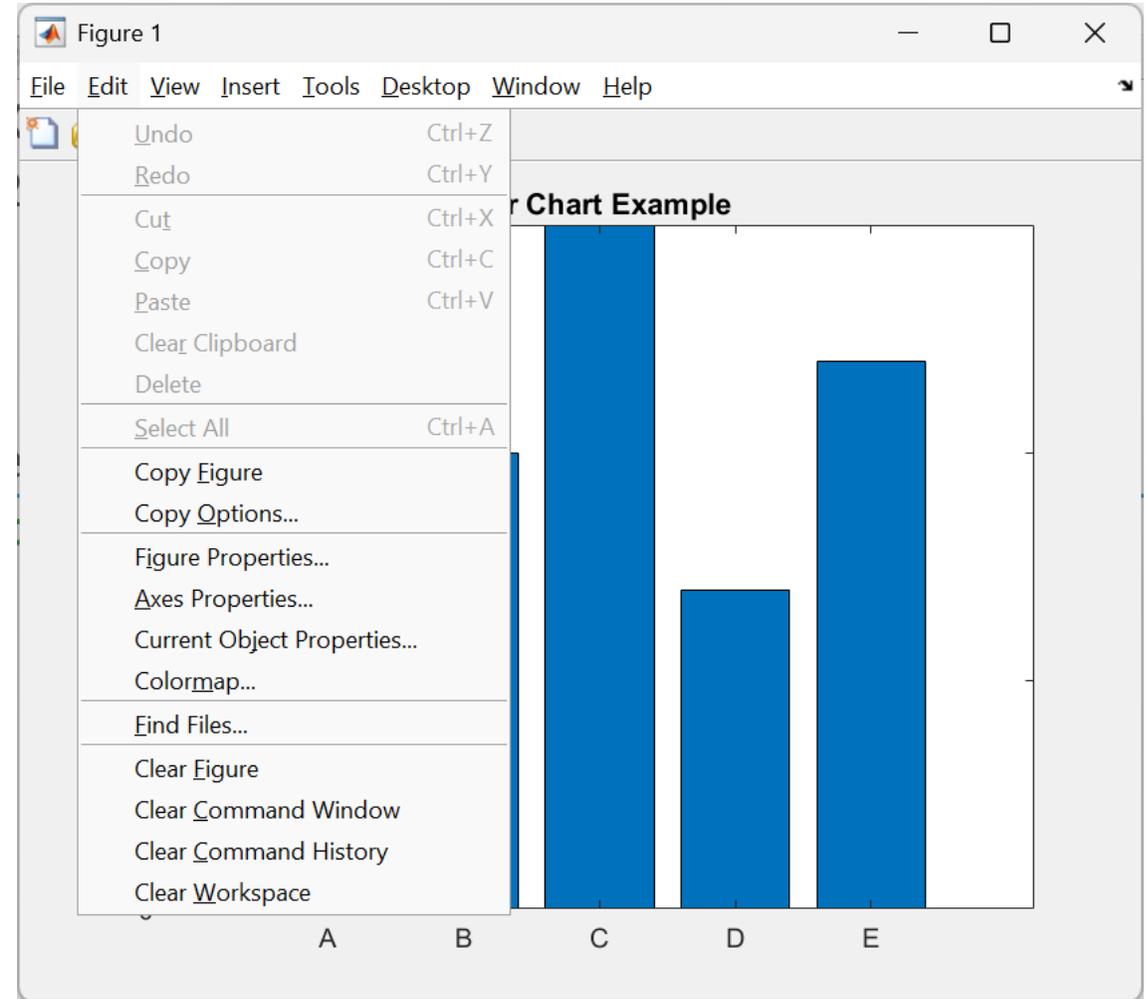
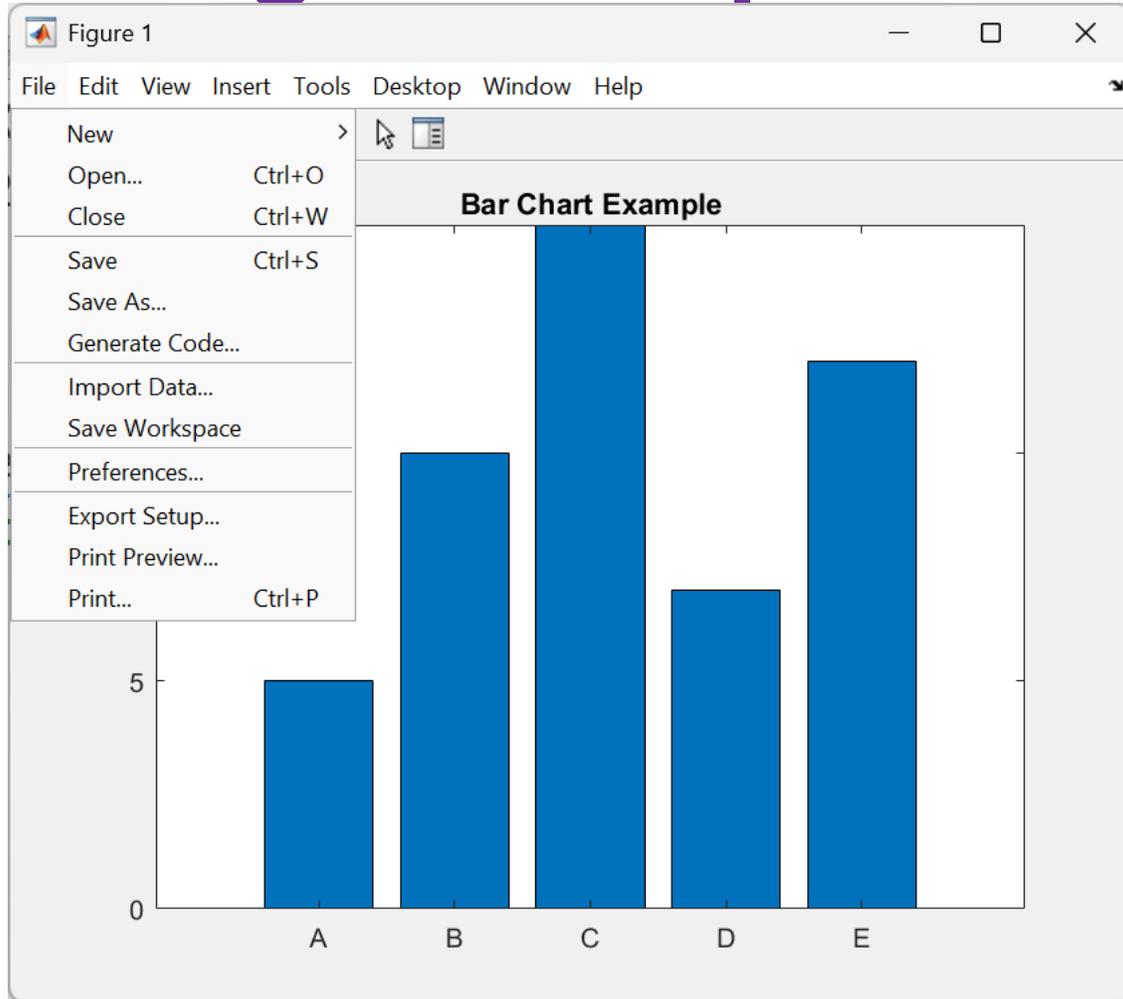
# Customizing Plot Styles



- Key Functions:
  - `plot(x, y, 'r--')` (Red dashed line)
  - `scatter(x, y, 'bo')` (Blue circle markers)
  - `bar(data, 'g')` (Green bar chart)
  - `grid on/off` (Enable/Disable grid)
  - `legend('Line 1', 'Line 2')` (Adds a legend)

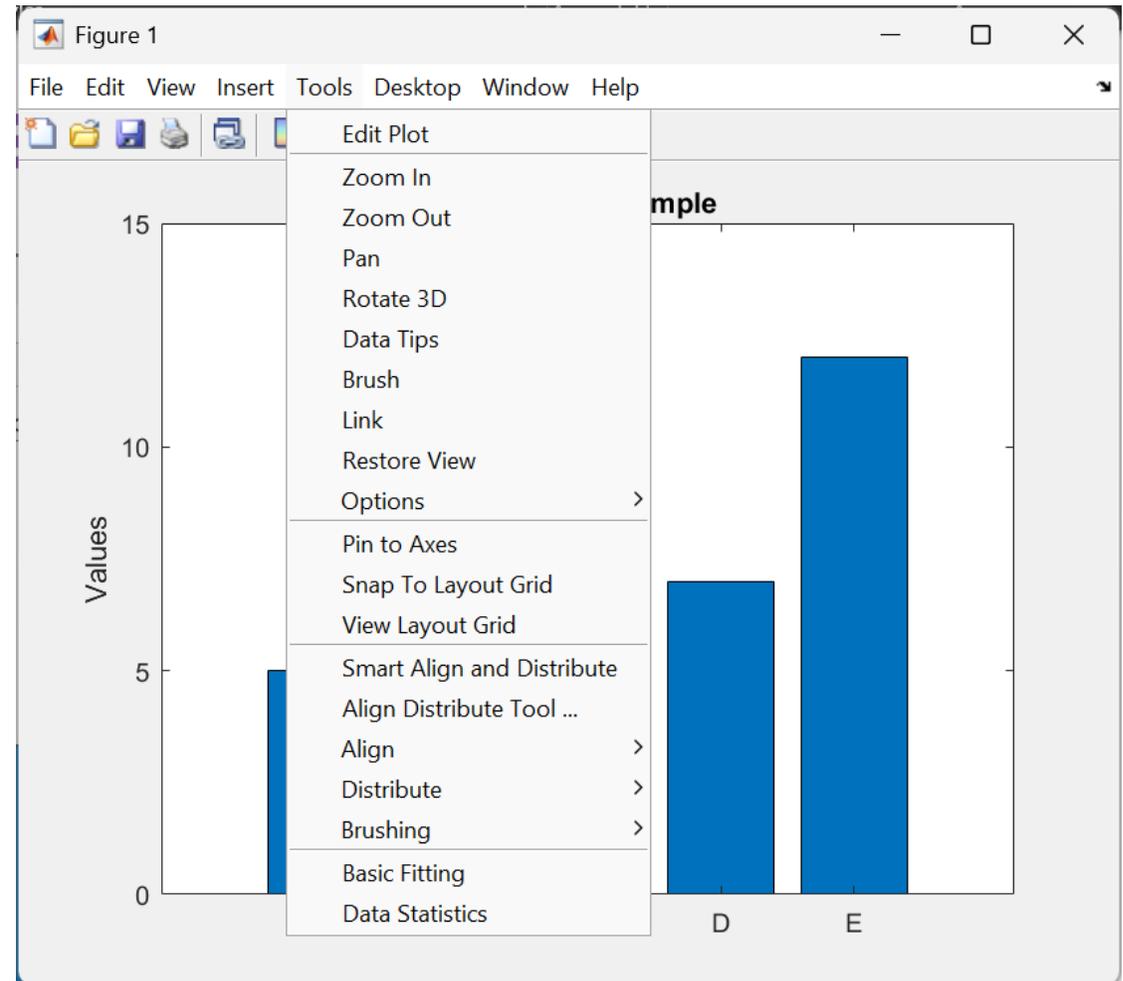
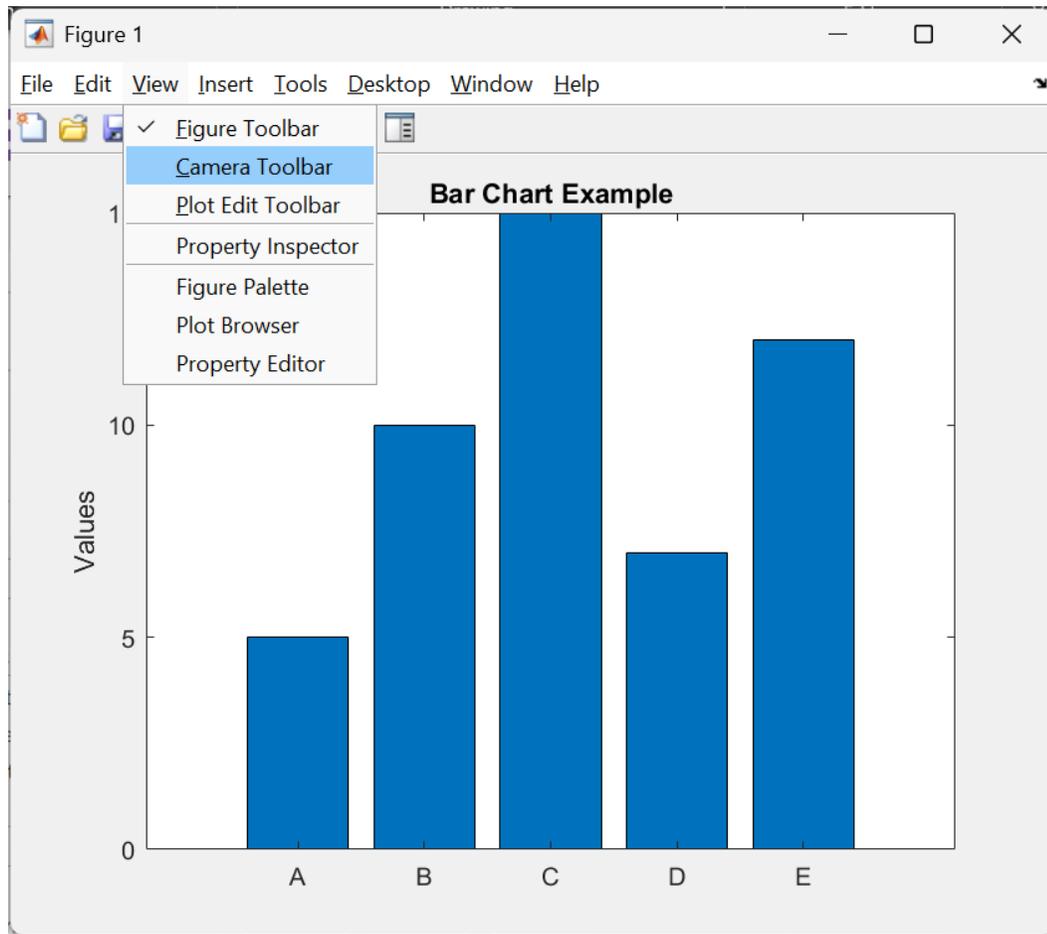


# Figure Props



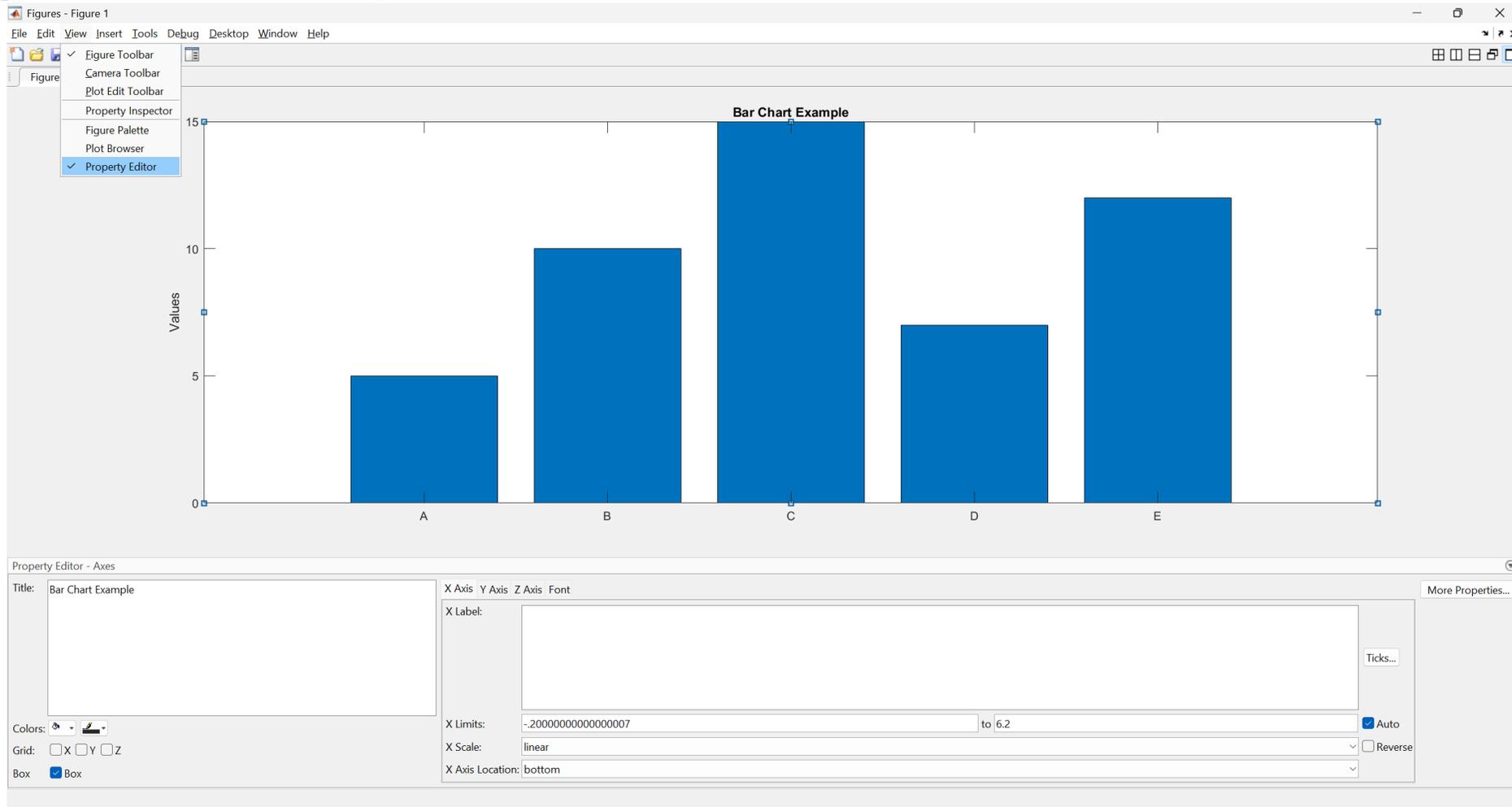


# Figure Props



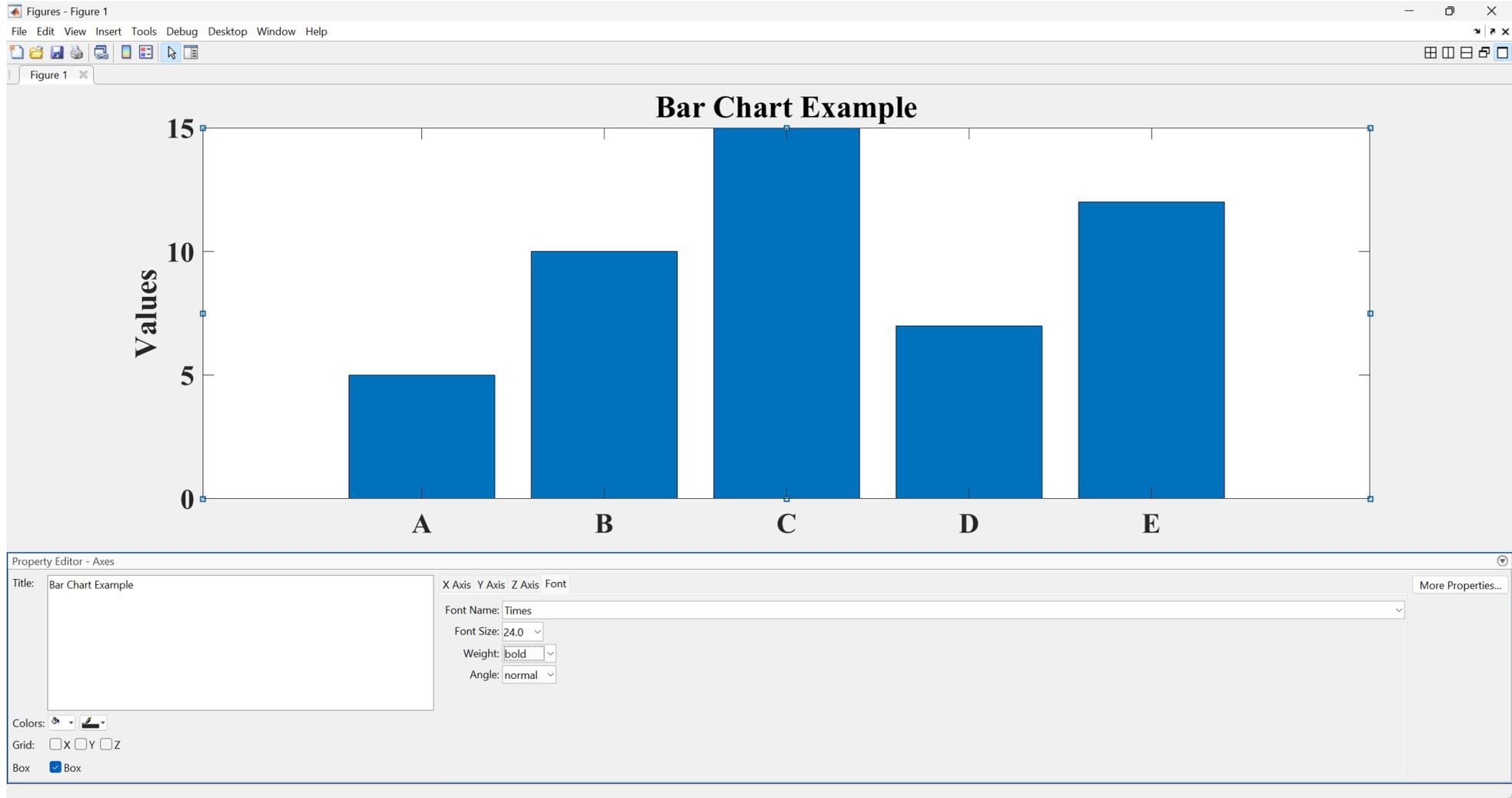


# Figure Props





# Figure Props





# Summary



- Creating line plots, scatter plots, bar charts, histograms, and 3D plots.
- Using subplot for multiple plots.
- Customizing plots with styles, colours, and legends.

