Islamic University of Technology (IUT)

Organization of Islamic Cooperation (OIC)

Department of Electrical and Electronic Engineering (EEE)

COURSE NO	**:**	**EEE 4416**
LAB NO	**:**	**09 (Part – C)**
TOPIC	**:**	**Beyond Basic Plots**

Geographical Plots

Geographical plots are visualizations that display data on a map using geographic coordinates (latitude and longitude). They are useful for showing where events or measurements occur and for identifying spatial patterns.

- Data points are placed using latitude (Y) and longitude (X).
- Often layered over a map or basemap (streets, terrain, satellite imagery).
- Can display additional variables through color, size, or density.

Common Types of Geographical Plots

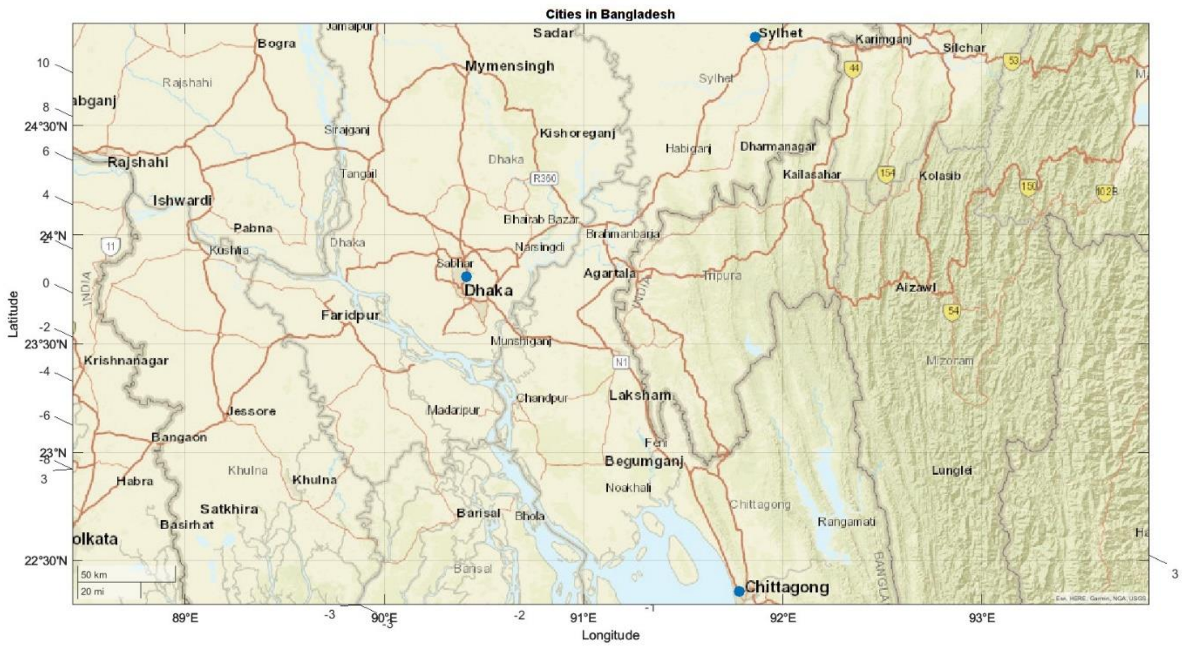
i. Point maps (**geoscatter**) – Show individual data points on a map.
ii. Path or route maps (**geoplot**) – Show connections or trajectories (e.g., travel routes).
iii. Heat/density maps (**geodensityplot**) – Highlight areas with high/low concentrations of points.
iv. Bubble maps (**geobubble**) – Points sized or colored according to a variable.
v. **Shapefile**/region maps – Show boundaries (countries, states, regions) with data overlays.

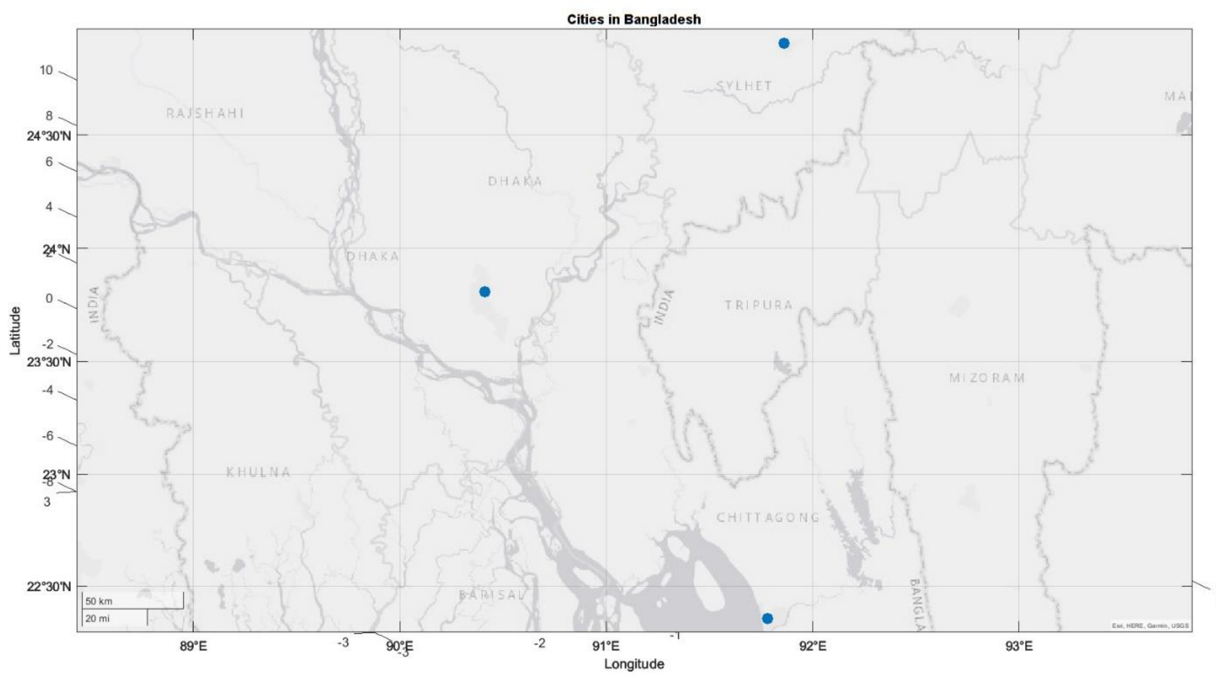
1. Geoscatter Plot

Instead of x and y values like the scatter plot, the geoscatter plot requires latitude and longitude values.

```
lat = [23.8103 22.3569 24.9045];   % Latitude of 3 cities
lon = [90.4125 91.7832 91.8611];   % Longitude

geoaxes;
geoscatter(lat, lon, 80, 'filled')
geobasemap streets
```

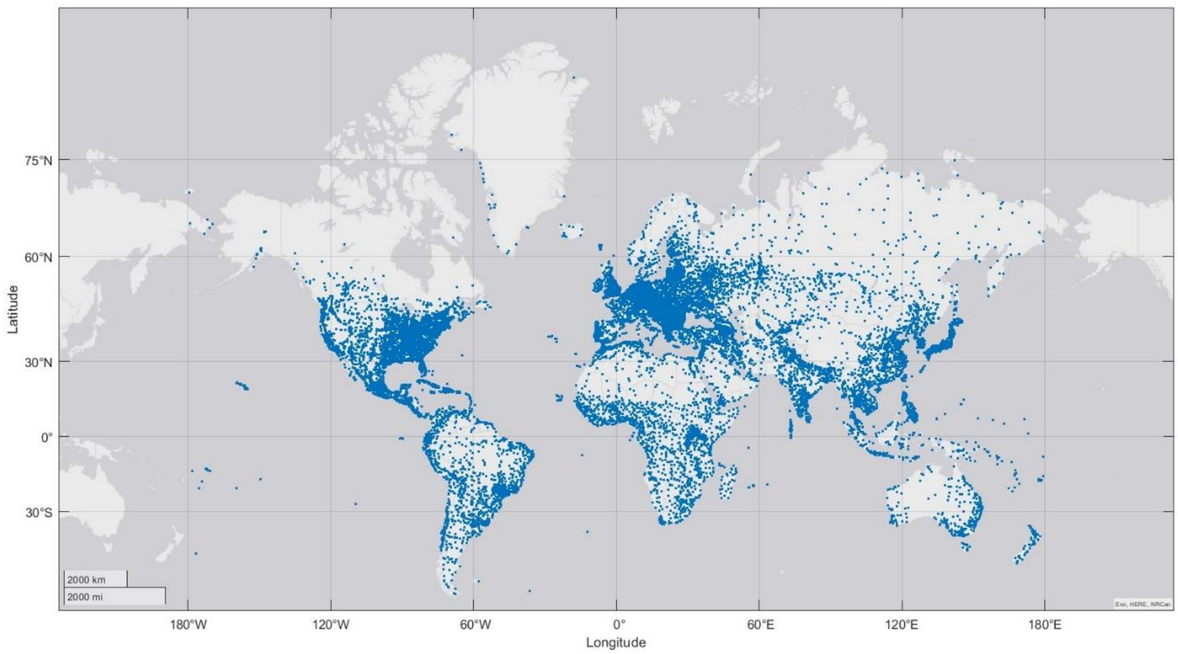
- The geobasemap offers several distinct visualizations such as ‘colorterrain’, 'landcover', 'satellite', etc. Check documentation for details.

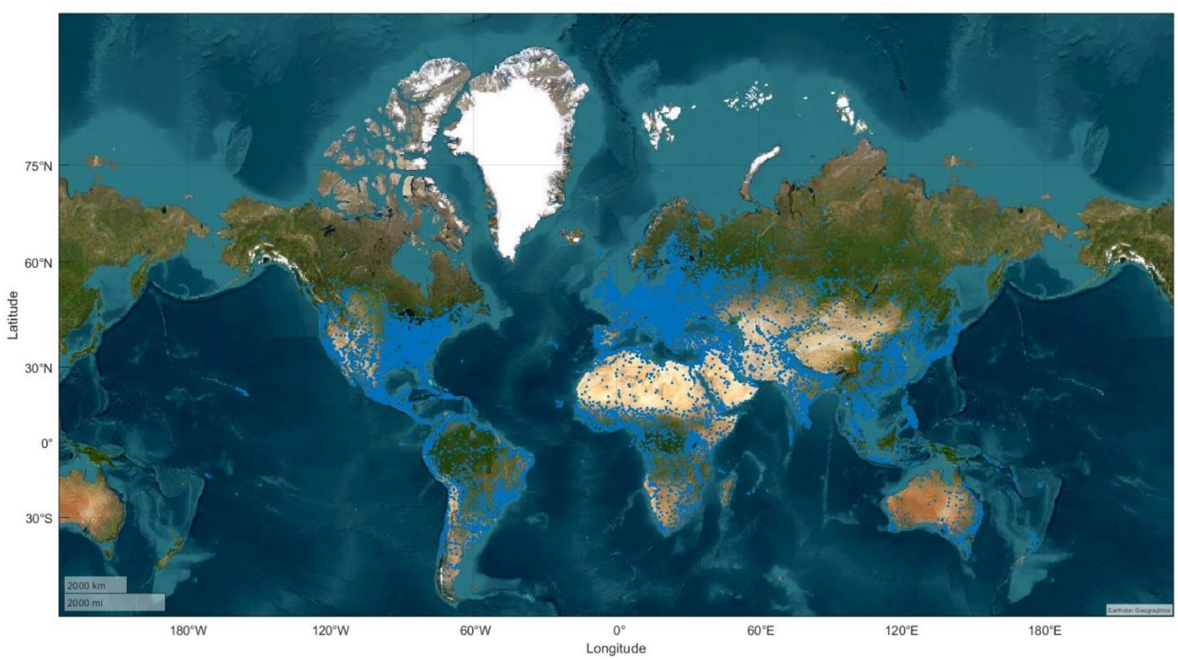




⇨ Import the **'worldcities'** dataset. Using the latitude and longitude columns, create a scatter plot of all the cities mentioned in the dataset.

```
geoscatter(worldcities.lat, worldcities.lng,  '.')
```

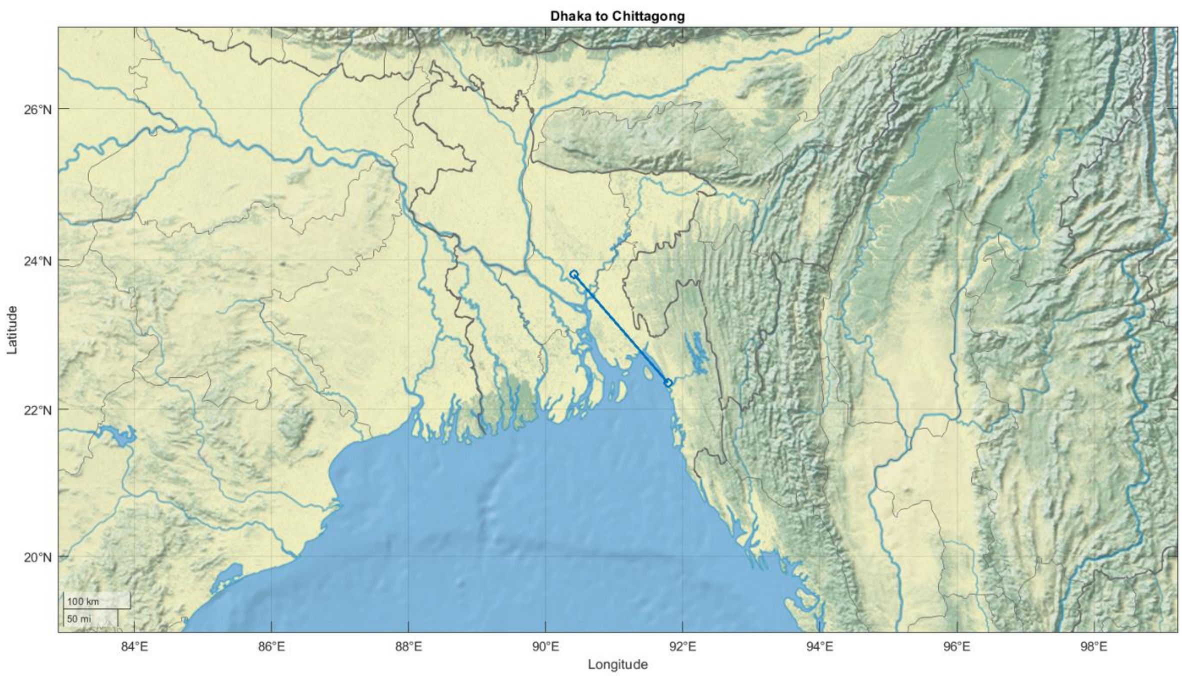




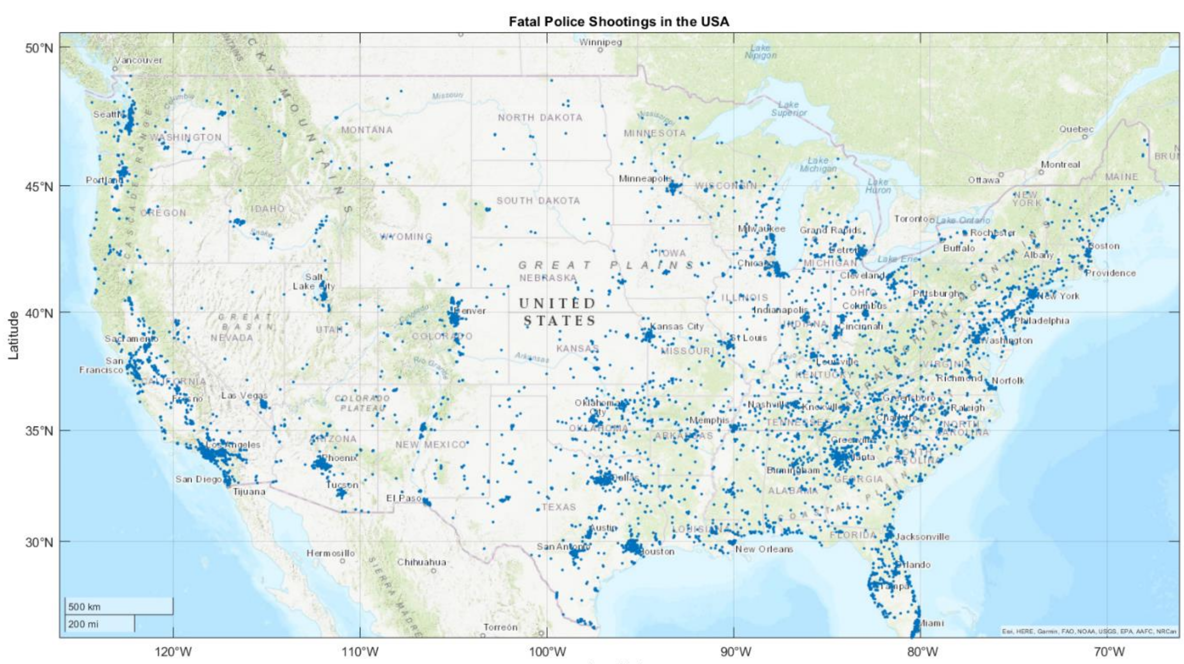
2. Geoplot

Map data points or routes on Earth's surface. geoplot(lat,lon) plots a line in a geographic axes with vertices at the latitude-longitude locations specified (in degrees) by the vectors lat and lon. The lat and lon inputs must be the same size. By default, geoplot draws a solid line, with no markers, using colors specified by the ColorOrder property of the Geographic axes object.

```
lat = [23.8103,22.3569];
lon = [90.4125,91.7832];

figure;
geoplot(lat,lon,'-o','LineWidth',2);
geobasemap landcover
title('Dhaka to Chittagong');
```



These types of geographical plots can effectively display geographical information when representing data.

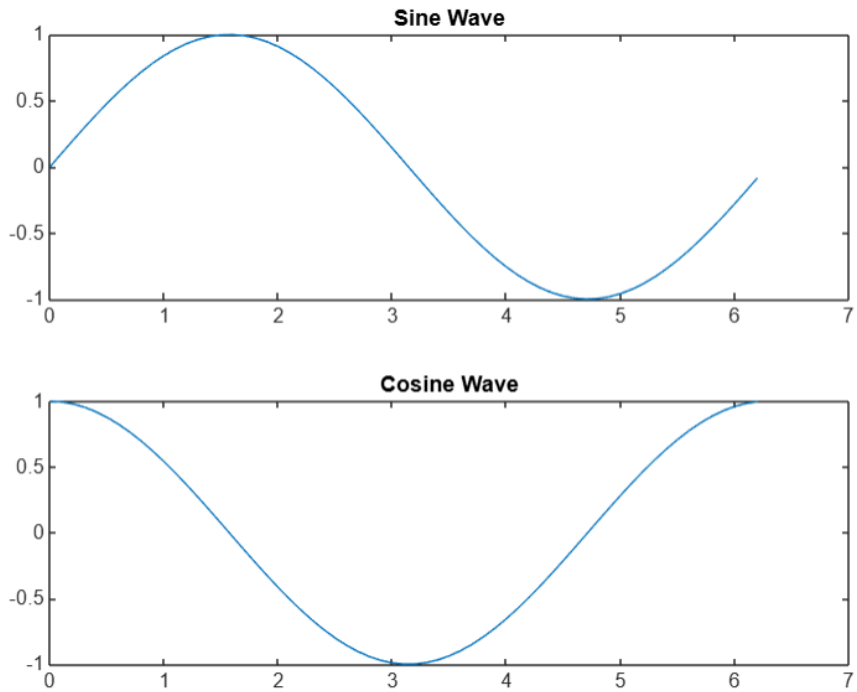


Subplots

A subplot is a way to display **multiple plots in a single figure window** by dividing the figure into a grid layout. Instead of creating multiple figure windows for each plot, we can organize several plots together for easy comparison using the subplot() function.

- ⇨ subplot(m, n, p) divides the figure into an m-by-n grid and activates the p-th plot for drawing. Here, p controls the position.

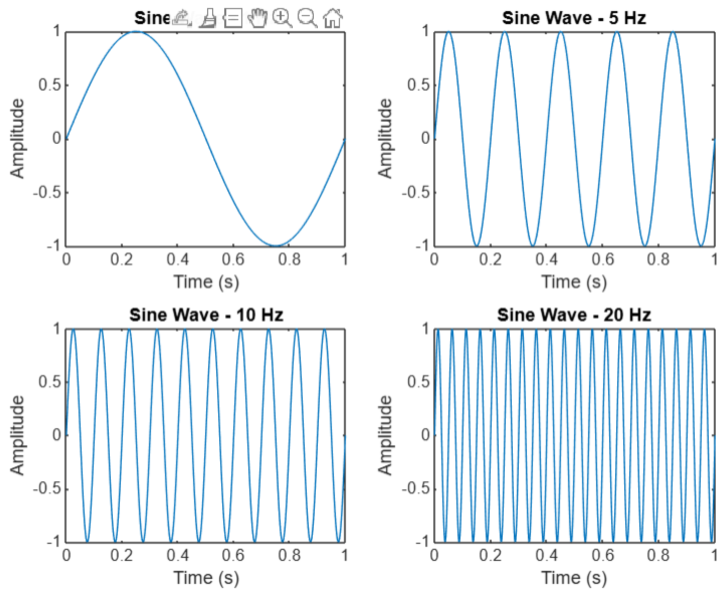
```
x = 0:0.1:2*pi;

y1 = sin(x);
y2 = cos(x);

subplot(2,1,1);    % 2 rows, 1 column, 1st plot
plot(x, y1)
title('Sine Wave')

subplot(2,1,2);    % 2nd plot
plot(x, y2)
title('Cosine Wave')
```



<u>Class Task</u>

Create the following subplot by creating a 2-by-2 grid.

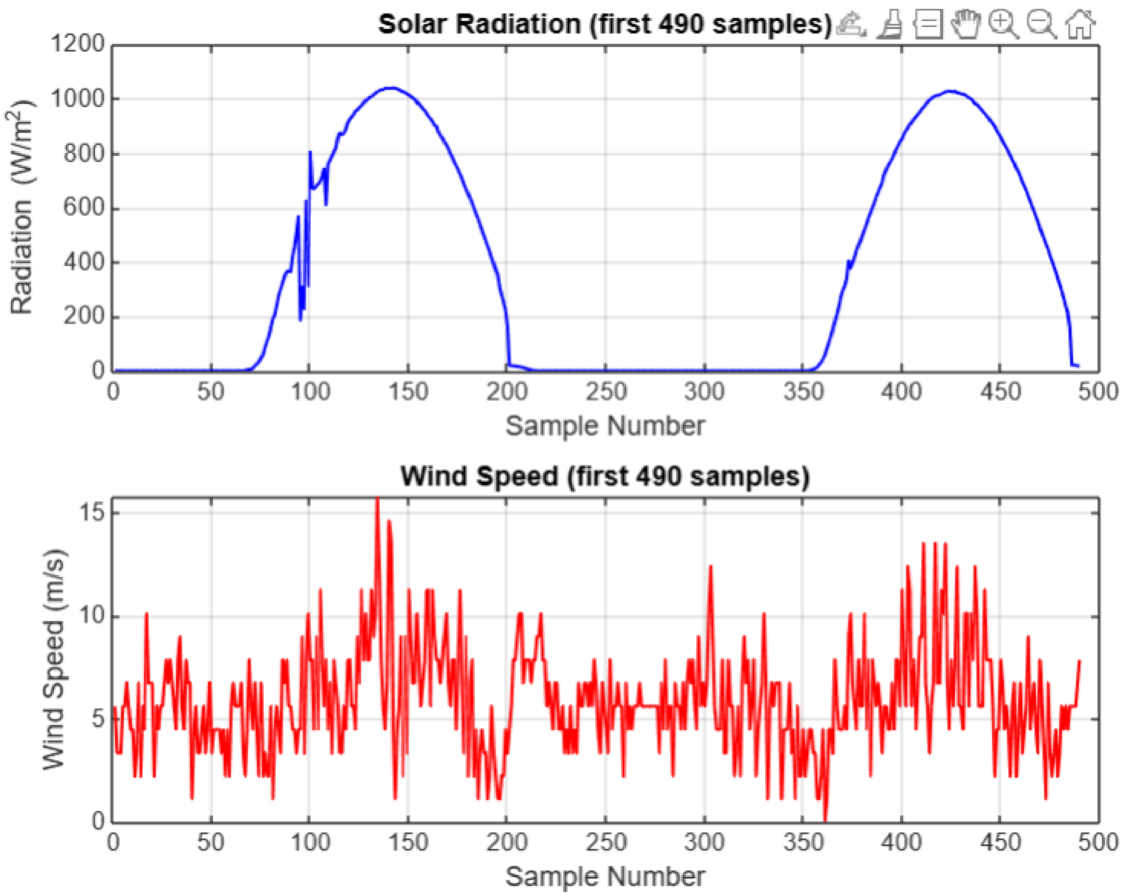
Asif Newaz, AKM Rakib, Mr. Salman Hossain
Lecturer, EEE, IUT



Plotting from Data

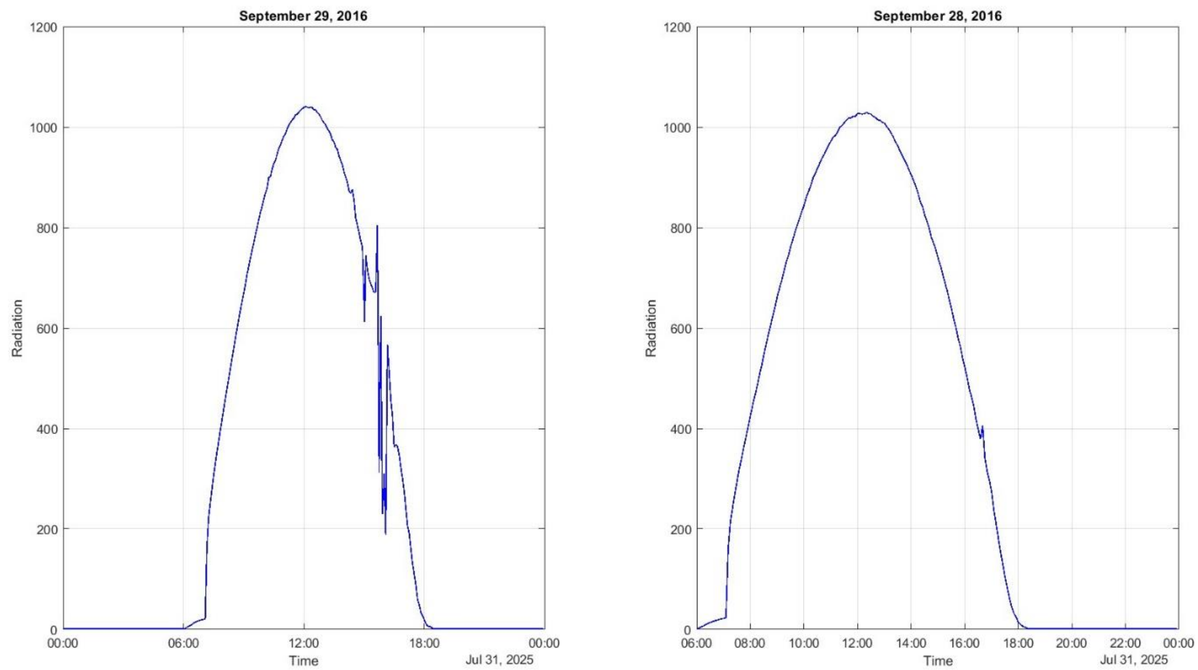
Import the **Weather data** file into your workspace. It's a time-series data. Let's plot some of the features to obtain insight into the data.

```
% Read the CSV file

data = readtable('weatherdata.csv');
head(Data)

% Number of rows to plot (change as needed)
numRows =490;
data = data(1:numRows, :);

figure;

% Plot Radiation (Irradiance) vs index
subplot(2,1,1);
plot(data.Radiation, '-b', 'LineWidth', 1.2);
xlabel('Sample Number');
ylabel('Radiation (W/m^2)');
title(['Solar Radiation (first ', num2str(numRows), ' samples)']);
grid on;

% Plot Wind Speed vs index
subplot(2,1,2);
plot(data.Speed, '-r', 'LineWidth', 1.2);
xlabel('Sample Number');
ylabel('Wind Speed (m/s)');
title(['Wind Speed (first ', num2str(numRows), ' samples)']);
grid on;
```



As you can see from the figure, when we plot solar radiation against the sample numbers, it does not really provide appropriate information. Instead, if we plot the feature information with respect to time, we might get a better idea of the variable.

```
% separating two days data manually. Can be done directly
using the datetime datatype

T29= T(1:285,:)
T28 = T(286:end,:)


subplot(1,2,1)
plot(T29.Time, T29.Radiation, 'b-', 'LineWidth', 1)
grid on
xlabel('Time');
ylabel('Radiation');
title('September 29, 2016');


subplot(1,2,2)
plot(T28.Time, T28.Radiation, 'b-', 'LineWidth', 1)
grid on
xlabel('Time');
ylabel('Radiation');
title('September 28, 2016');
```

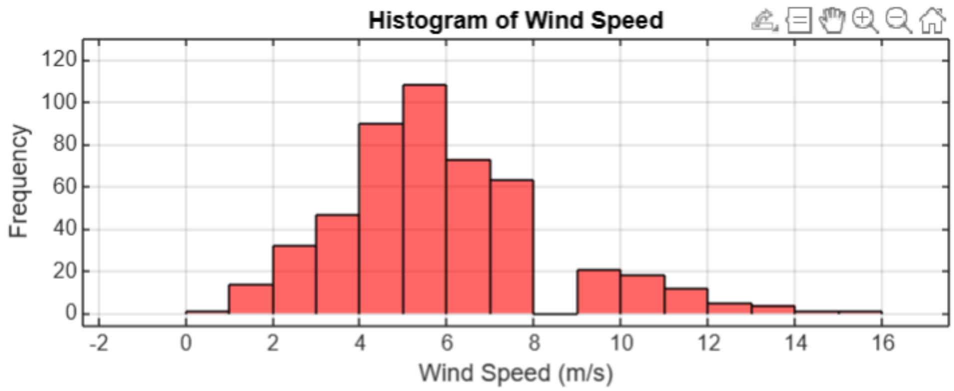


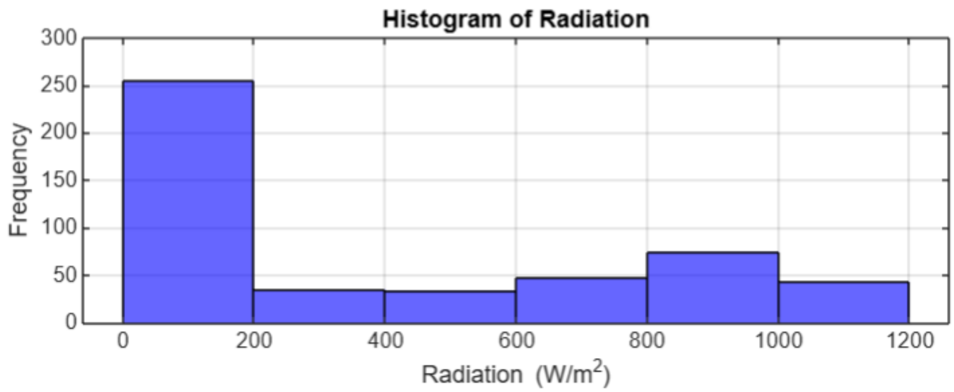
While this approach provides a good comparison and preserves the time information, the two subplots currently use different x-axis tick values, resulting in a **lack of synchronization**. How to fix that?

- What other plots can we use to represent the information in the data?

Histogram of Wind Speed and Solar Radiation

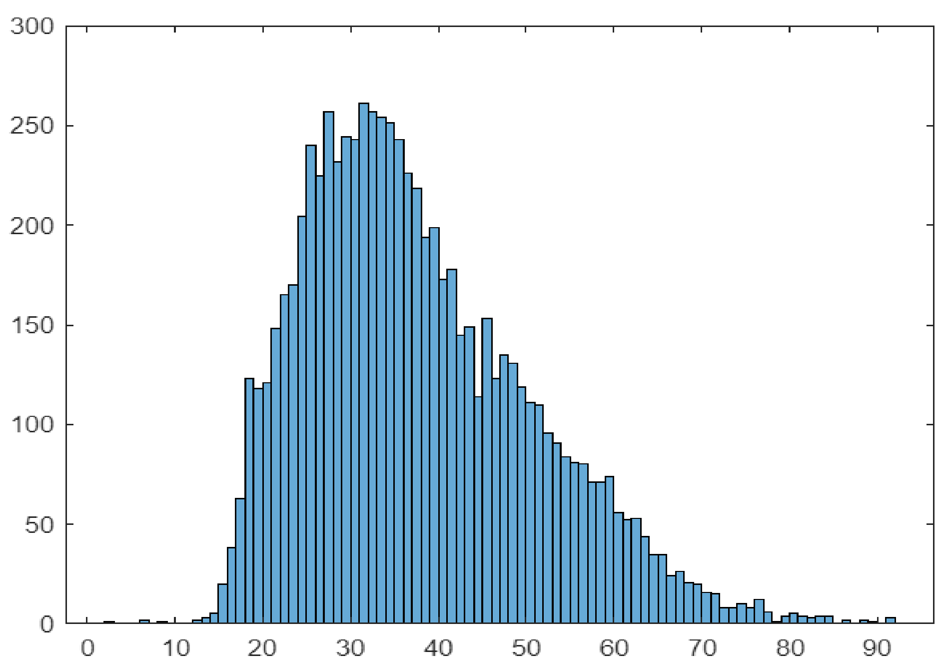
```
figure;
subplot(2,1,1);
histogram(data.Speed, 'FaceColor', 'r');
xlabel('Wind Speed (m/s)');
ylabel('Frequency');
title('Histogram of Wind Speed');
grid on;


% Histogram: Radiation
subplot(2,1,2);
histogram(data.Radiation, 'FaceColor', 'b');
xlabel('Radiation (W/m^2)');
ylabel('Frequency');
title('Histogram of Radiation');
grid on;
```





Import the ‘**fatal police shootings**’ dataset. The dataset contains information regarding fatal shootings in the USA. This includes the race of the deceased, the circumstances of the shooting, whether the person was armed, whether the person was experiencing a mental-health crisis, etc. The data has been recorded since 2015. Data up to 2024 is available.

Let’s explore some basic information in the data. Plot the histogram of the ‘Age’ column. What interesting insight can we obtain from the figure?



Creating Animation

Creating a GIF or video of plots is an effective way to visualize **how data evolves over time or across iterations**. Unlike static plots, animated plots capture transitions, making patterns, trends, and anomalies easier to understand. They are also highly engaging for presentations, publications, or social media, allowing complex concepts to be communicated clearly and dynamically.

Let's plot the data of fatal police shootings in the USA over the years. We want to visualize how the shootings (frequency, location) changed over time. From 2015 to 2024, we have around 10 years of data. It can be properly represented using a GIF/Video file, which contains a separate plot for each year.

Check the code below to create a GIF like this. You need to run a for loop over 10 years, while each year's plot needs to be saved in a file using the **imwrite** function. The **getframe** function allows you to capture the plot as an image frame. Then you just have to save the frames with a small delay to display each frame one after the other.

```
% excluding alaska for now

% Define the latitude and longitude boundaries for the continental USA (excluding Alaska)
lat_min = 24.396308;   % Southernmost point (Florida)
lat_max = 49.384358;   % Northernmost point (Minnesota)
lon_min = -125.0;      % Westernmost point (California)
lon_max = -66.93457;   % Easternmost point (Maine)


% Apply the filter for continental USA, excluding Alaska
in_usa = (data.latitude >= lat_min) & (data.latitude <= lat_max) & (data.longitude >=
lon_min) & (data.longitude <= lon_max);


% Extract the valid locations
df= data(in_usa,:);
```

```
  % Initialize GIF
   filename = 'police_killings.gif';
   years= 2015:2022;


   for i = 1:length(years)
       currentYear = years(i);
       subset = df(df.year == currentYear, :);

       figure;
       geoscatter(subset.latitude, subset.longitude,  '.');
       geobasemap topographic
       title(sprintf('Police Killings in %d', currentYear));


       % Capture the plot as an image
       frame = getframe(gcf);
       im = frame2im(frame);
       [imind, cm] = rgb2ind(im, 256);


       % Write to the GIF file
       if i == 1
           imwrite(imind, cm, filename, 'gif', 'Loopcount', inf, 'DelayTime', 1);
       else
           imwrite(imind, cm, filename, 'gif', 'WriteMode', 'append', 'DelayTime', 1);
       end


       pause(1);
   end
```

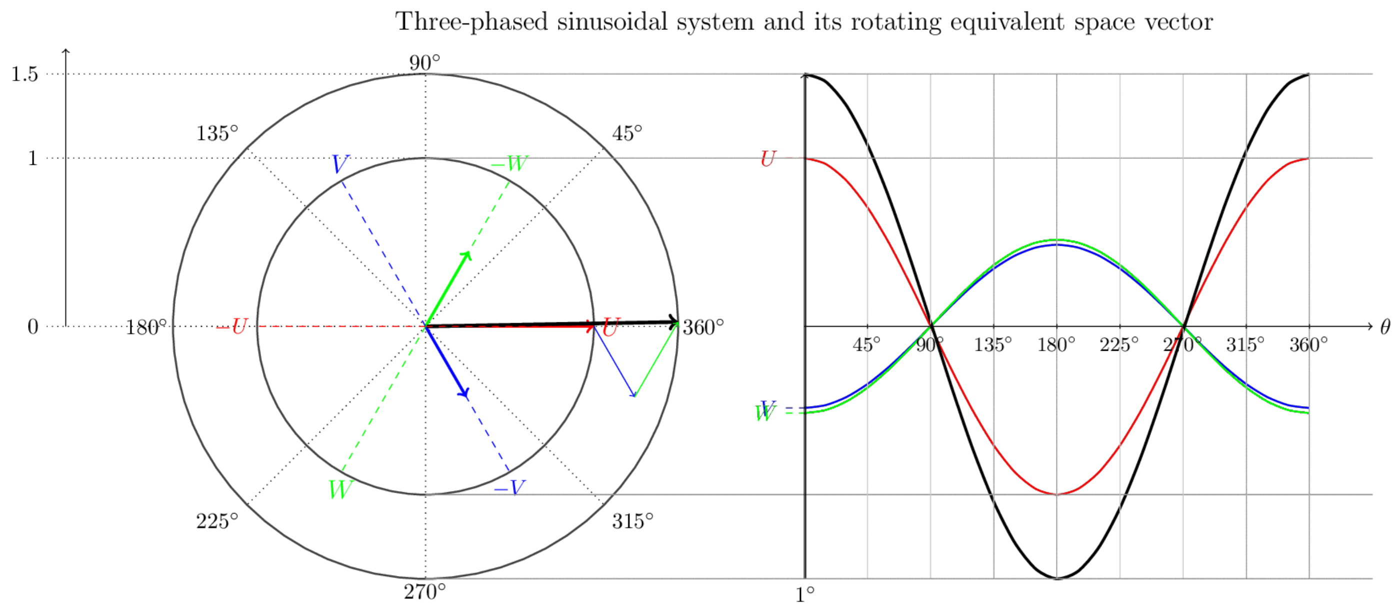
Let’s create another animation showing the 3-phase sinusoidal waves. The animation is saved as an .mp4 file.

```
%% Animated Video of Slow and Clear 3-Phase Oscillations


%% --- Parameters ---
Vm = 1;              % Peak voltage
f  = 1;              % Frequency (1 Hz: one cycle per second)
w  = 2*pi*f;
Fs = 200;            % Sampling frequency
T_total = 6;         % Total time: 6 seconds = 6 full cycles
t = 0:1/Fs:T_total;

%% --- Generate 3-phase signals ---
va = Vm * cos(w*t);
vb = Vm * cos(w*t - 2*pi/3);
vc = Vm * cos(w*t + 2*pi/3);

%% --- Video Writer Setup ---
video = VideoWriter('slow_visible_3phase_oscillation.mp4', 'MPEG-4');
video.FrameRate = 10;    % Slow playback
open(video);

%% --- Animate ---
figure('Color','w');
step = 4;  % frame step size

for k = 2:step:length(t)
    clf;
    plot(t(1:k), va(1:k), 'r', 'LineWidth', 2); hold on;
    plot(t(1:k), vb(1:k), 'g', 'LineWidth', 2);
    plot(t(1:k), vc(1:k), 'b', 'LineWidth', 2);

    ylim([-Vm Vm]*1.2);
    xlim([0 t(end)]);

    xlabel('Time (s)');
    ylabel('Amplitude');
    title('Slow 3-Phase Oscillations (1 Hz)');
    legend('v_a','v_b','v_c','Location','southoutside','Orientation','horizontal');
    grid on;

    frame = getframe(gcf);
    writeVideo(video, frame);
end

close(video);
disp('Video saved as slow_visible_3phase_oscillation.mp4');
```

Class Task

i. Create a similar GIF file of **'unarmed'** police shootings over the years. There is a column that contains the information on whether the victim was armed or unarmed. Extract the records of unarmed cases first, then plot the figures.

ii. Create a video file plotting a balanced 3-phase sinusoid in the time domain and showing its resultant (space) vector.



Asif Newaz, AKM Rakib, Mr. Salman Hossain
Lecturer, EEE, IUT