Advanced Plotting in MATLAB

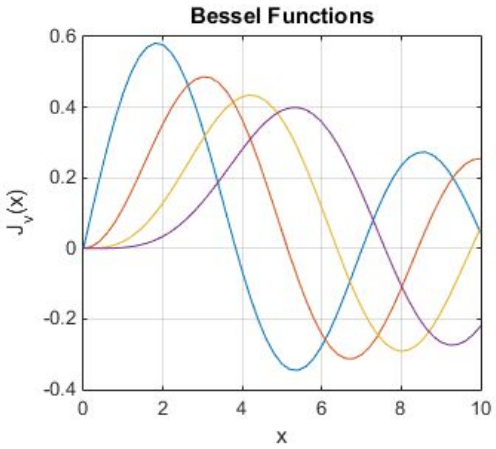
CME 192 Lecture 2

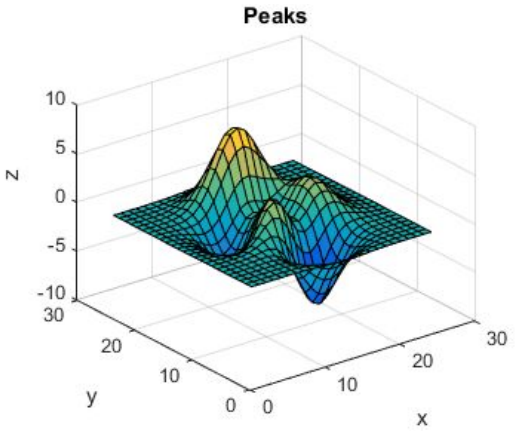
01/07/2025

Stanford University

Refresher on MATLAB Graphics

How does MATLAB plot things under the hood?





Graphics Objects

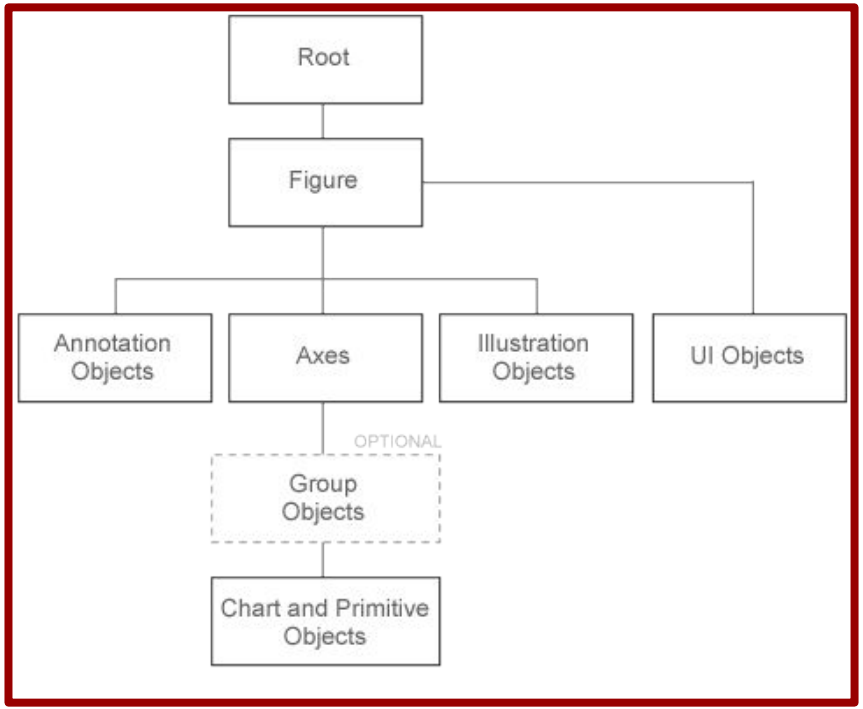
Basic drawing elements used by MATLAB to display data

- Each object instance has unique identifier, the **handle**.
- Graphics objects behave like other MATLAB objects.
- Objects are organized in hierarchy.

Parent-Child Relationship

- E.g., the parent of an axes is a figure.

A handle refers to a specific instance of a graphics object used to **set and query** the values of the object properties.



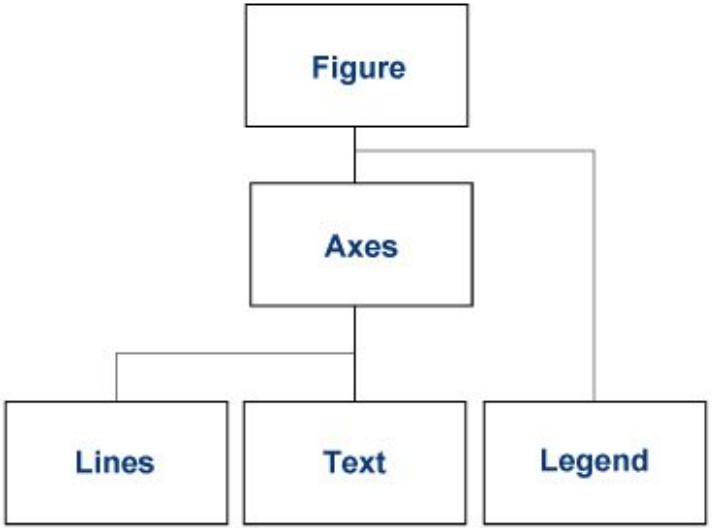
Graphics Objects

Core graphics object

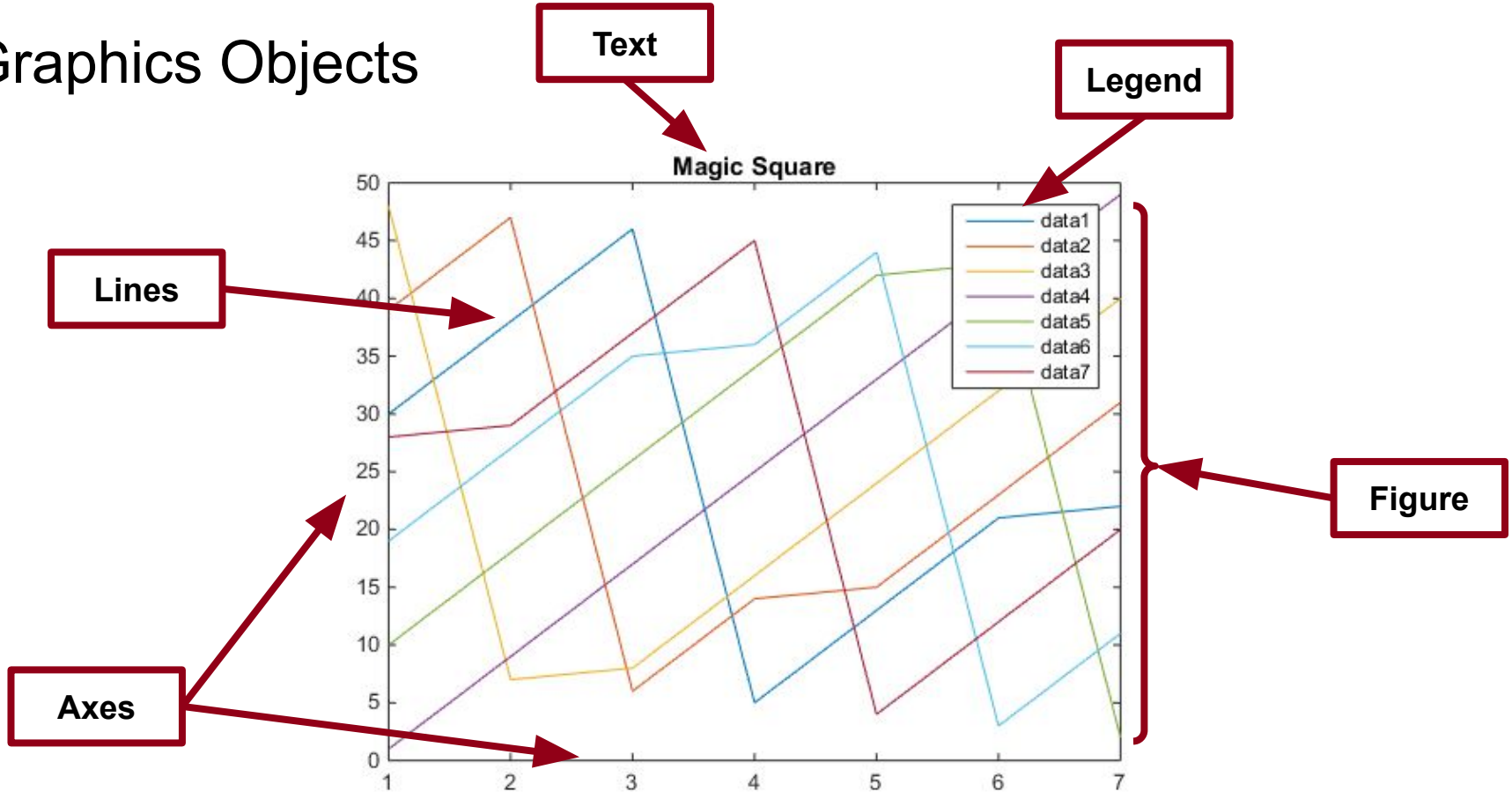
- `axes, image, line, patch, rectangle, surface`

Composite graphics object

- Plot objects: `areaseries, barseries, contourgroup, errorbarseries, lineseries, quivergroup, scattergroup, staircase, stemseries, surfaceplot`
- Annotation objects: `arrow, doublearrow, ellipse, line, rectangle, textarrow, textbox`
- Illustration objects: `legend, colorbar`
- Group objects: `hggroup, hgtransform`
  - Group objects can contain any of the objects that axes can contain, such as lines, surfaces, text, etc., also other group objects.
- User Interface objects



Graphics Objects



Working with Graphics Objects

There are 2 ways to access/modify object properties.

1. Use dot notation to refer to a particular object and property.
2. Use the `set` and `get` functions to access properties.
   - `s,V: structure; pn, pv: cell array`
   - Set property-value pairs: `set(han,s)`
   - Set value of property `pn{i}` to `pv{i}:set(han,pn,pv)`
   - Store all properties-value pairs in a structure: `V = get(han)`
   - Store property value in V: `V = get(han,'Property')`

Property inspector

- Open the figure inspector, click “Select & Plot Edit”, click “More Properties”
- Use the command prompt inspect.

Properties of Figure, Axes, and Plot Objects

Type object name in command to see all properties and defaults.

Figure

- `Colormap, Position, PaperPositionMode`

Axes (contain the lines, surfaces, and other objects that represent the data visualized in a graph)

- `XLim, YLim, ZLim, CLim, XGrid, YGrid, ZGrid, XTick, XTickLabel, YTick, YTickLabel, ZTick, ZTickLabel, XScale, YScale, ZScale`

Plot (composite graphics objects of one or more core objects in a group)

- `XData, YData, ZData, Color, LineStyle, LineWidth`

Properties Common to All Objects

- `Parent, Children, BeingDeleted (onwhen object's DeleteFcnis called), DeleteFcn (Callback rountine that excutes when object is deleted), CreateFcn, Selected,Visible, etc.`

MATLAB R2025b

HOME PLOTS APPS LIVE EDITOR INSERT VIEW

Search (⌘⇧Space) John

Datatips Full Screen Line Numbers Zoom In Zoom Out Reset Zoom Clear All Output Output on Right Output Inline Hide Code Spelling Outline

DISPLAY ZOOM OUTPUT VIEW REVIEW TOOLS

/ › Users › winnicki › Documents › MATLAB › Lecture_02 ›

Files

Name

data
figures_lecture2
Lecture2_NHANES_Kaggle_v7_lectureStyle.asv
Lecture2_NHANES_Kaggle_v7_lectureStyle.mlx

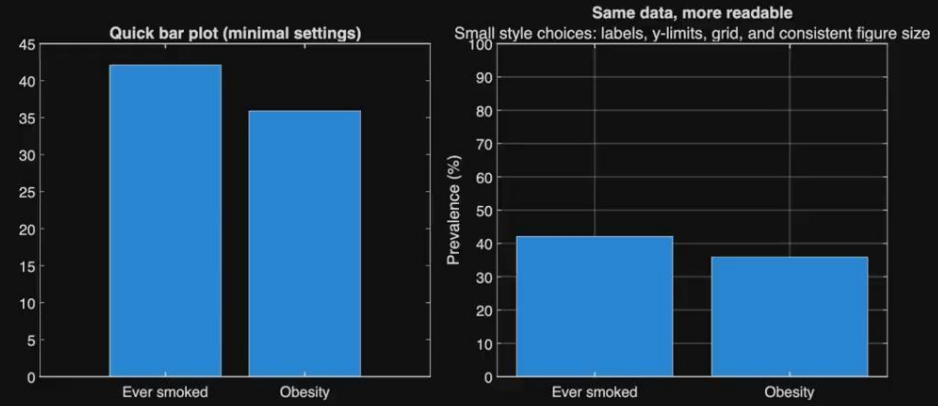
Workspace

Name	Value	Size	Class
ageBin	960×1 dou...	960×1	double
ageCen...	1×21 double	1×21	double
ageDec	1000×1 do...	1000×1	double
ageEdges	1×22 double	1×22	double
ans	1×1 Figure	1×1	matlab.ui.Fi...
ax	1×1 Axes	1×1	matlab.gra...
ax1	1×1 Axes	1×1	matlab.gra...
ax2	1×1 Axes	1×1	matlab.gra...
AGE	28×21 dou...	28×21	double
bmiBin	960×1 dou...	960×1	double
bmiCen...	1×28 double	1×28	double
bmiEdges	1×29 double	1×29	double
bwidthB...	1	1×1	double
bwidthC...	10	1×1	double

Lecture_03_NLA.mlx × Lecture2_NHANES_Kaggle_v7_lectureStyle.mlx * ×

/Users/winnicki/Documents/MATLAB/Lecture_02/Lecture2_NHANES_Kaggle_v7_lectureStyle.mlx

```
197  subtitleIfAvailable("Small style choices: labels, y-limits, grid, and consistent figure size");
```




```
198
199  %% Prevalence by sex: using tiledlayout and shared y-axis
200
201  % Next, we take the same idea (prevalence of obesity and "ever smoked") and
202  % look at it by sex. This gives a natural example for using tiledlayout to
203  % make a small set of related panels with a shared y-axis, which makes
204  % comparisons much easier for the viewer.
205
206  G = groupsummary(Core, "Sex", "mean", ["Obese","SmokeEver"]);
207
208  f = newFig("Prevalence by sex", [980 420]);
209  tl = tiledlayout(f, 1, 2, "Padding","compact", "TileSpacing","compact");
210
```

Editor: 100% UTF-8 LF Script Ln 197 Col 1

Legend

typical syntax

- legend('First plotted’, 'Second plotted’, 'Location’, 'Northwest')

fine-grained control

- legend(han, 'han(1)label’, 'han(2)label’, 'Location’, 'Northwest')

legend handle

- get the handle by leg = legend()
- use handle to control size/location (more control than 'Location'), font size/style, line style, etc.

Callback Routines

Function associated with graphics handle that gets called in response to a specific action applied to the associated graphics object

- Object creation, deletion
- Mouse motion, mouse press, mouse release, scroll wheel
- Key press, key release

All callback routines are automatically passed two inputs.

- Handle of component whose callback is being executed
- Event data

Callback routines can be specified in many possible forms.

- String: expression evaluated in base workspace.
- Function handle
- Cell arrays to pass additional arguments to callback routine

Don’t forget to reset root!

Key Dataset for Today: NHANES

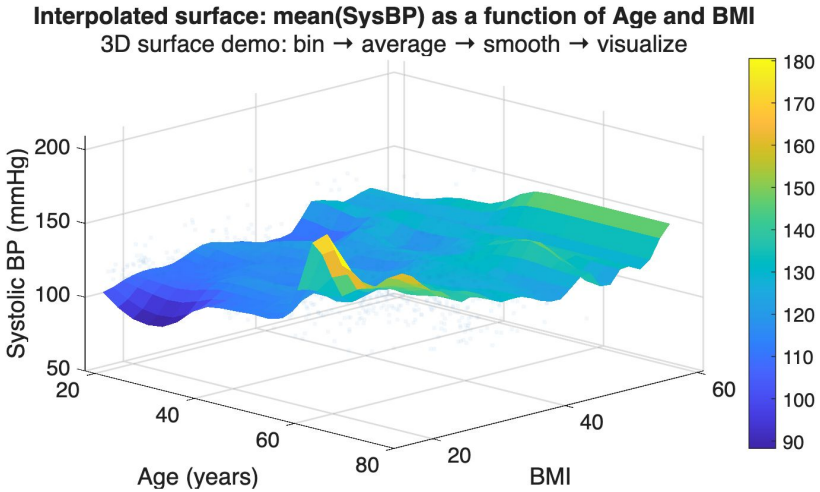
Open your livescripts to try the examples for yourself

NHANES Dataset Overview

- NHANES = National Health and Nutrition Examination Survey
- Conducted by CDC's National Center for Health Statistics (NCHS)
- Ongoing program: combines interviews, physical exams, and laboratory tests
- Designed to assess the health and nutritional status of adults and children in the U.S.
- Data publicly available at https://www.cdc.gov/nchs/nhanes/
- We will be using a subset of the data downloaded from Kaggle: https://www.kaggle.com/datasets/cdc/national-health-and-nutrition-examination-survey/data

Key Plotting Tools: Bar charts, Scatter Plots, 3D Surfaces, and more!

Open your livescripts to try the examples for yourself



Visualization is everywhere in scientific computing and data analysis, … and it is important!

"Saper Vedere": To see is to understand

Line Plots	Scatter and Bubble Charts	Data Distribution Plots	Discrete Data Plots	Geographic Plots	Polar Plots	Contour Plots	Vector Fields	Surface and Mesh Plots	Volume Visualization	Animation	Images
plot	scatter	histogram	bar	geoplot	polarplot	contour	quiver	surf	streamline	animatedline	image
plot3	scatter3	histogram2	barh	geoscatter	polarhistogram	contourf	quiver3	surfc	streamslice	comet	imagesc
stairs	bubblechart	scatterhistogram	bar3	geobubble	polarscatter	contour3	feather	surfl	streamparticles	comet3	
errorbar	bubblechart3	boxchart	bar3h		polarbubblechart	contourslice		ribbon	streamribbon		
area	swarmchart	swarmchart	pareto		compassplot	fcontour		pcolor	streamtube		
stackedplot	swarmchart3	swarmchart3	stem		fpolarplot			fsurf	coneplot		
loglog	spy	piechart	stem3					fimplicit3	slice		
semilogx		donutchart	stairs					mesh			
semilogy		wordcloud						meshc			
fplot		bubblecloud						meshz			
fplot3		heatmap						waterfall			
fimplicit		parallelplot						fmesh			
		plotmatrix									

MATLAB Plotting Functions

Goals

- Pick the right plot type for your data
- Learn a few modern charting functions (histogram, boxchart, swarmchart, heatmap, …)
- Remember to properly format your plots after generating them: labels, legends, limits, and exporting

Line Plots

Compare series, track changes over x (time, index, distance, …)

2-D & 3-D lines

`plot`	2-D line plot (x,y) or multiple series
`plot3`	3-D line/point plot (x,y,z)
`stairs`	Stairstep graph (sampled/quantized data)
`errorbar`	Line plot with error bars
`area`	Filled area plot (emphasize cumulative magnitude)
`stackedplot`	Stacked axes with common x-axis (many signals)

Log scaling

`loglog`	Log–log scale plot
`semilogx`	Log x-axis, linear y-axis
`semilogy`	Linear x-axis, log y-axis

Function plots (avoid manual sampling)

`fplot`	Plot expression/function handle over an interval
`fimplicit`	Plot implicit curve f(x,y)=0
`fplot3`	Plot 3-D parametric curve

Quick tips

- Label everything: title, xlabel/ylabel, legend
- For multi-panel figures: tiledlayout + nexttile
- For log plots, handle zeros/negatives first

```
tiledlayout(2,1)
nexttile; plot(t,y); grid on
nexttile; semilogy(t,abs(y));
grid on
xlabel('t (s)'); ylabel('|y|')
```

Data Distribution Plots

Histograms, box charts, and “show me the spread” visuals

Histograms

`histogram`	Histogram of 1-D data (modern replacement for hist)
`histogram2`	Bivariate histogram (2-D bins / heatmap style)
`morebins`	Increase number of histogram bins
`fewerbins`	Decrease number of histogram bins
`histcounts`	Compute bin counts (no plotting)
`histcounts2`	Compute 2-D bin counts

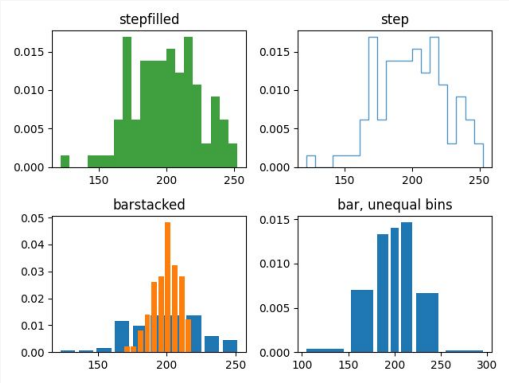
Robust summaries & point clouds

`boxchart`	Box chart (quartiles + whiskers; modern)
`swarmchart`	Swarm scatter (jittered points to show density)
`swarmchart3`	3-D swarm scatter

Quick tips

- Start with histogram(x); tune NumBins/BinWidth
- Use 'Normalization','pdf' to compare groups
- Prefer boxchart + swarmchart to show outliers

```
histogram(x,'Normalization','pdf
')
hold on
xline(mean(x),'--');
xline(median(x),':')
title('Distribution with
mean/median')
```



Bubble Charts

Encode an additional scalar via marker size (and optionally color)

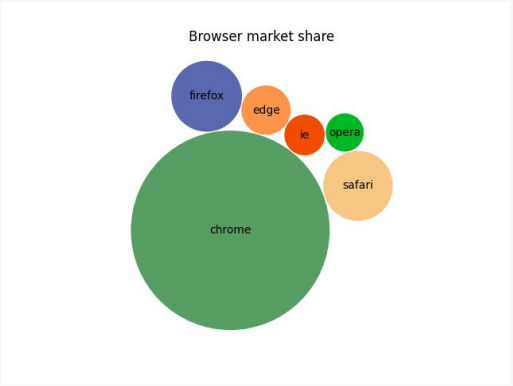
`bubblechart`	Bubble chart (x,y,size; color optional)
`bubblechart3`	3-D bubble chart
`bubblelim`	Map bubble sizes to a data range
`bubblesize`	Set min/max bubble size (points)
`bubblelegend`	Legend for bubble sizes

Tip: Bubble area (not radius) is what viewers perceive.
Use bubblelim/bubblesize so sizes remain comparable across plots.

Quick tips

- Use bubblelegend so size is interpretable
- Keep point counts modest (sample/aggregate)
- Use MarkerFaceAlpha when points overlap

```
bubblechart(x,y,s)
bubblelim([min(s) max(s)])
bubblesize([4 20]);
bubblelegend("s")
```



Scatter Plots

Relationships between variables (and diagnostics for matrices)

Core scatter

`scatter`	2-D scatter plot
`scatter3`	3-D scatter plot
`binscatter`	Binned scatter (density via bin counts)
`scatterhistogram`	Scatter with marginal histograms

Many-variable and matrix views

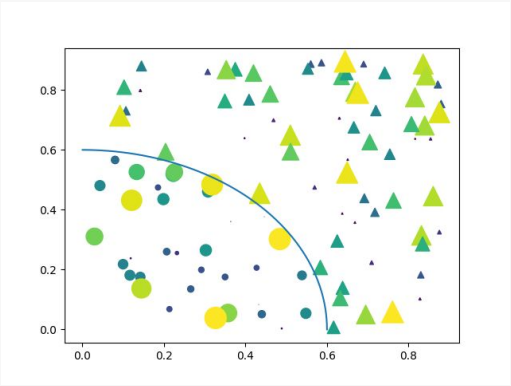
`plotmatrix`	Scatter plot matrix (pairwise relationships)
`parallelplot`	Parallel coordinates (many variables per row)
`spy`	Sparsity pattern of a matrix (nonzeros)

Workflow: define (X,Y) grid → compute (U,V) components → quiver/streamline.

Quick tips

- For dense data: binscatter or MarkerFaceAlpha
- Show trend: lsline or fit + plot (when appropriate)
- Use axis equal when geometry matters

```
scatter(x,y,12,'filled','MarkerFaceAlpha',0.35)
grid on; axis tight
xlabel('x'); ylabel('y');
title('x vs y')
```



Composition, Text, and Heatmaps

When values are counts, categories, or words

Text & composition

`wordcloud`	Word cloud chart from text data
`bubblecloud`	Bubble cloud chart (labeled circles)
`pie`	Pie chart
`pie3`	3-D pie chart

Heatmaps & summaries

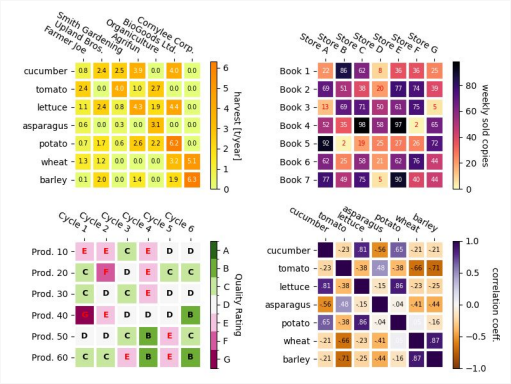
`heatmap`	Heatmap chart (good for tables/correlation)
`sortx`	Sort elements in heatmap rows
`sorty`	Sort elements in heatmap columns

Tip: prefer bar charts for precise comparisons; use pie/word clouds for quick “big picture” storytelling.

Quick tips

- For many variables: heatmap or stackedplot
- Use sortx/sorty to reveal structure
- Clean labels: rotate tick labels if needed

```
h = heatmap(A);
sortx(h,"descending");
sorty(h,"descending")
title('Heatmap')
```



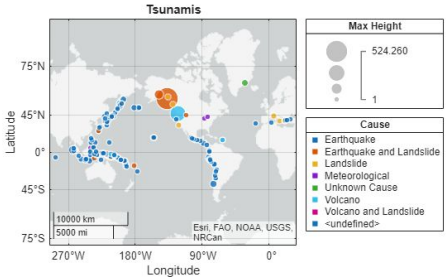
Geographic Plots

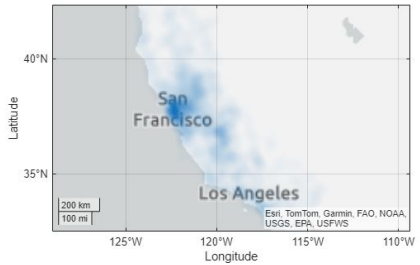
Latitude/longitude over interactive basemaps

`geoplot`	Plot line in geographic coordinates
`geoscatter`	Scatter chart in geographic coordinates
`geobubble`	Visualize data values at specific locations
`geodensityplot`	Geographic density plot

Helpful control (not exhaustive):

`geobasemap, geolimits, geotickformat, geolayout`

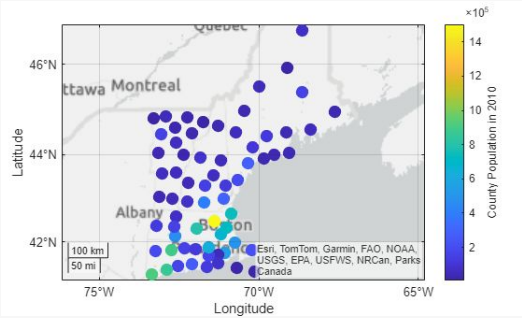




Quick tips

- Choose a basemap (geobasemap) early
- Use geolimits to focus your region
- For many points: geodensityplot

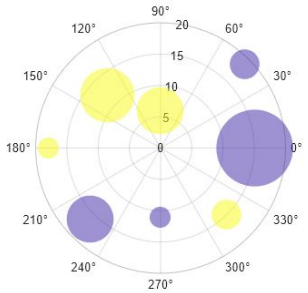
```
geoscatter(lat,lon,12,'filled')
geobasemap streets
geolimits([37 38],[-123 -121])
```

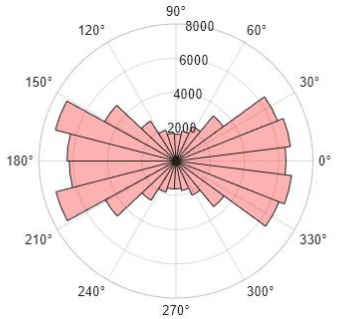


Polar Plots

Angles + radii (periodic phenomena, directions, spectra)

`polarplot`	Plot line in polar coordinates
`polarscatter`	Scatter chart in polar coordinates
`polarbubblechart`	Polar bubble chart
`polarhistogram`	Histogram chart in polar coordinates
`compass`	Arrows emanating from origin
`ezpolar`	Easy polar coordinate plotter

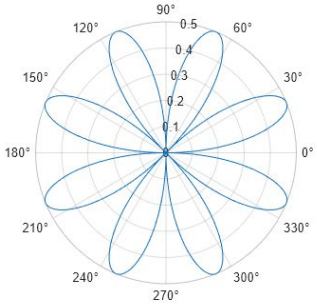




Quick tips

- Angles are in radians by default
- Use rlim/thetalim to focus the view
- polarhistogram is great for direction data

```
polarplot(theta,r);
rlim([0 1]); thetalim([0 180])
grid on
```

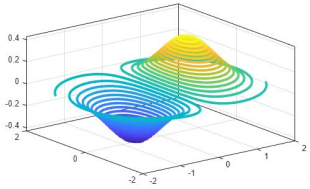


Contour, Surface, and Mesh Plots

Functions of two variables and gridded data

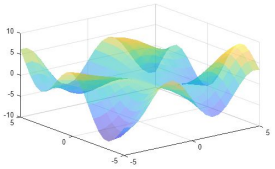
Contour plots

`contour`	Contour lines (levels)
`contourf`	Filled contour plot
`contourc`	Compute contour matrix (low-level)
`contour3`	3-D contour plot
`contourslice`	Contours in volume slice planes
`clabel`	Label contour elevation
`fcontour`	Contours of a function handle



Surface plots

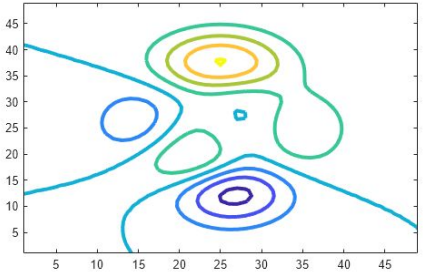
`surf`	Surface plot
`surfc`	Surface with contour plot underneath
`surface`	Primitive surface plot
`surfl`	Surface with colormap-based lighting
`surfnorm`	Surface normals
`hidden`	Remove hidden lines from mesh plots
`fsurf`	Surface plot of a function handle



Quick tips

- Use meshgrid to create a grid (or fcontour/fsurf)
- Add colorbar and set view(3) for 3-D
- For lighting: camlight + lighting gouraud

```
[X,Y] = meshgrid(x,y);
surf(X,Y,Z); shading interp
colorbar; view(3); camlight;
lighting gouraud
```



Mesh, Pseudocolor, and 3-D Geometry

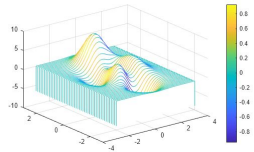
Wireframes, ribbons, and primitives

Mesh plots (wireframe)

`mesh`	Mesh surface plot
`meshc`	Mesh with contour plot beneath
`meshz`	Mesh with curtain
`fmesh`	Plot 3-D mesh
`fimplicit3`	Plot 3-D implicit function

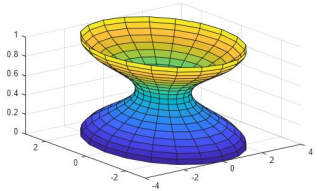
Specialized 3-D charts

`waterfall`	Waterfall plot (stacked lines in 3-D)
`ribbon`	Ribbon plot
`pcolor`	Pseudocolor plot (colored cells)
`surf2patch`	Convert surface data to patch data



Primitives

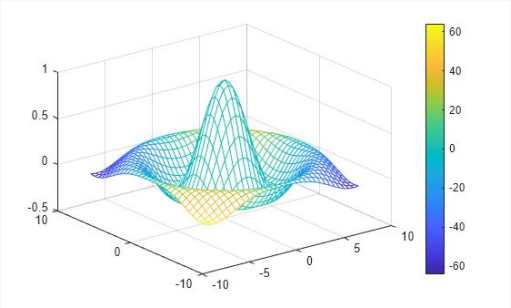
`cylinder`	Create cylinder coordinates
`ellipsoid`	Create ellipsoid coordinates
`sphere`	Create sphere coordinates



Quick tips

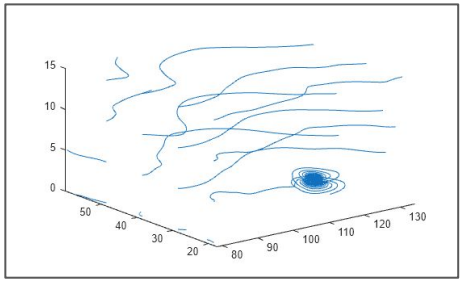
- Use pcolor + shading flat for matrix “cells”
- For patch workflows, start with surf2patch
- Simplify heavy meshes with reducepatch

```
pcolor(A); shading flat;
colorbar
patch(surf2patch(X,Y,Z));
reducepatch(p,0.2)
```



Vector Fields

Velocity, force, gradients, flow patterns



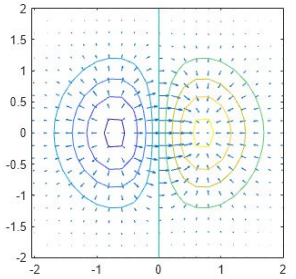
`quiver`	2-D quiver (vector) plot
`quiver3`	3-D quiver (vector) plot
`compass`	Arrows from origin (direction/magnitude)
`feather`	Arrows from x-axis (often used for spectra)
`streamline`	Streamlines from 2-D or 3-D vector data
`streamslice`	Streamlines in slice planes

Workflow: define (X,Y) grid → compute (U,V) components → quiver/streamline.
For readability, downsample vectors (e.g., every 2–5 points).

Quick tips

- Normalize or scale vectors to compare direction
- Use fewer arrows: subsample the grid
- Streamlines show flow better than arrows for dense fields

```
skip = 3;
quiver(X(1:skip:end,1:skip:end),
...
       Y(1:skip:end,1:skip:end),
...
       U(1:skip:end,1:skip:end),
...
       V(1:skip:end,1:skip:end))
```

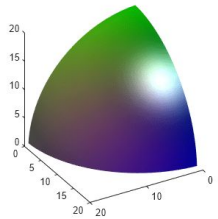


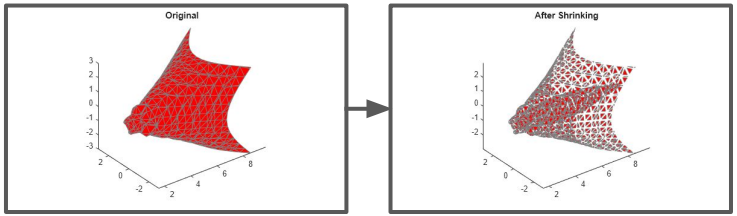
Volume Visualization (Core)

3-D gridded scalar fields (medical, CFD, tomography, …)

Scalar volume tools (3-D gridded scalar fields)

`slice`	Orthogonal slice planes	`smooth3`	Denoise / smooth 3D data
`contourslice`	Contours on slice planes	`reducevolume`	Downsample volume grid
`isosurface`	Surface at constant value	`reducepatch`	Simplify mesh faces
`isocaps`	Close open ends (“caps”)	`subvolume`	Crop to region of interest
`isocolors`	Color patches from volume	`volumebounds`	Coordinate + color limits
`isonormals`	Normals for lighting	`shrinkfaces`	Separate faces visually

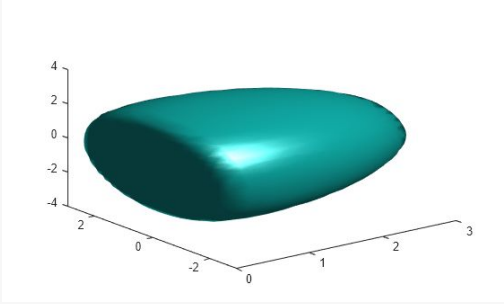




Quick tips

- Pick an isovalue (threshold) with physical meaning
- Use isonormals + lighting for depth cues
- Downsample/crop first to keep it interactive

```
V = smooth3(V,'box',5);
p = patch(isosurface(V,iso));
isonormals(V,p);
set(p,'FaceColor','red','EdgeColor','none');
camlight; lighting gouraud;
axis tight; daspect([1 1 1]);
```



Volume Visualization (Flow + Derived Fields)

Streamlines, particles, and operators like curl/divergence

Vector volume visualizations

`coneplot`	Velocity vectors as cones in 3-D
`stream2`	Compute 2-D streamline data
`stream3`	Compute 3-D streamline data
`streamline`	Plot streamlines from vector data
`streamparticles`	3-D stream particles (like motion)
`streamribbon`	3-D stream ribbon plot
`streamslice`	3-D streamlines in slice planes
`streamtube`	3-D stream tube plot

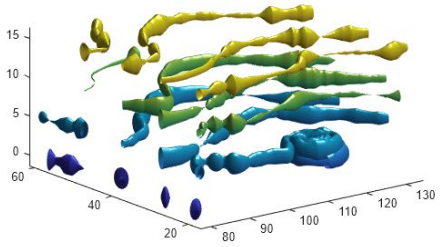
Derived field diagnostics

`curl`	Curl / angular velocity of vector field
`divergence`	Divergence (sources/sinks)
`interpstreamspeed`	Interpolate streamline vertices from flow speed

Quick tips

- Highlight rotation: curl (vorticity)
- Highlight sources/sinks: divergence
- Seed points matter: choose a plane/line

```
sx = linspace(xmin,xmax,10);
sy = linspace(ymin,ymax,10);
[SX,SY] = meshgrid(sx,sy);
SZ = z0*ones(size(SX));
streamtube(X,Y,Z,U,V,W,SX,SY,SZ)
```



Cleaning up your plots

MATLAB provides a couple convenient commands to clean up your plots so that they look professional

Layout + axes hygiene

- tiledlayout / nexttile for multi-panel figures
- grid on, axis tight/equal, xlim/ylim for framing
- xlabel/ylabel/title + legend for meaning
- colorbar + clim/caxis for color encodings

```
tiledlayout(2,2);
nexttile; plot(x,y); grid on
xlabel('x'); ylabel('y'); title('Demo')
axis tight
```

Export + reproducibility

- exportgraphics for consistent output
- Use vector formats (PDF/SVG) for line art
- Prefer code over manual formatting for repeatability
- Set figure size before exporting

```
set(gcf,'Color','w')
exportgraphics(gcf,'figure.png','Resolution',300)
exportgraphics(gcf,'figure.pdf','ContentType','vecto
r')
```

Generating Plots for Publication

Generating Publication-Quality Graphics

- Use graphics handles to deal with aesthetics
- Generate data in MATLAB and plot in document
- Save to a proper format
  - e.g., PNG, SVG, PDF, TikZ/PGF (popular choices for LATEX)
- Matlab2tikz(a popular script available via File Exchange)
  - Convert MATLAB figures into native TikZ/Pgfplots figures
- exportgraphics(ax,filename) (R2020a or later)
  - The graphics object can be any type of axes, a figure, a standalone visualization, a tiled chart layout, or a container within the figure. The resulting graphic is tightly cropped to a thin margin surrounding the content.
  - E.g., exportgraphics(gca,"myplot.jpg","Resolution",300)

Generating Publication-Quality Graphics

- Generate plot with all lines/labels/annotations/legends/etc Set properties (graphics handles or interactively)
  - Figure width/height
  - Axes line width, object line width, marker size
  - Font sizes
  - Adjust white space if necessary
- Save figure to file
  - WYSIWYG: 'PaperPositionMode' is 'auto' by default. You may also set it by `f.PaperPositionMode = 'auto';`
  - Print to file for inclusion in document: `print(gcf,'-depsc2',filename);` `saveas(gcf, filename); matlab2tikz(filename);`

Publish MATLAB Code Files (.m)

Publishing a MATLAB Code file (.m) creates a formatted document that includes the code, comments, and output.

To publish your code:

1. Create a MATLAB script or function. Divide the code into steps or sections by inserting two percent signs (%%) at the beginning of each section.
2. Document the code by adding explanatory comments at the beginning of the file and within each section.
3. Publish the code. On the Publish tab, click **Publish**, or Use `publish(file,format)`

Animation

Publish MATLAB Code Files (.m)

3 basic techniques for creating animations in MATLAB

1. Update the properties of a graphics object and display the updates on the screen.
   a. Useful for creating animations when most of the graph remains the same.
2. Apply transforms to objects.
   a. Useful when operating on the position and orientation of a group of objects together.
   b. Group the objects as children under a transform object. Create the transform object using `hgtransform`. Setting the Matrix property of the transform object adjusts the position of all its children.
3. Create a movie.
   a. Useful for complex animation that does not draw quickly in real time, and for replay.
   b. Use the `getframe` and `movie` functions to create a movie.

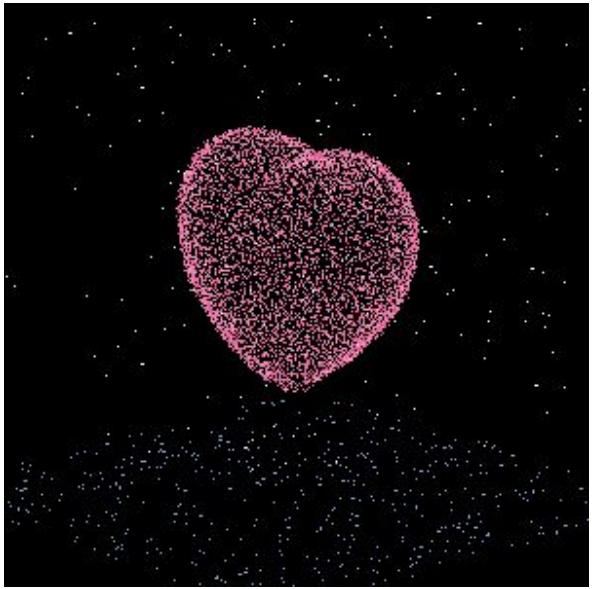
How to Create an Animation

Before entering loop

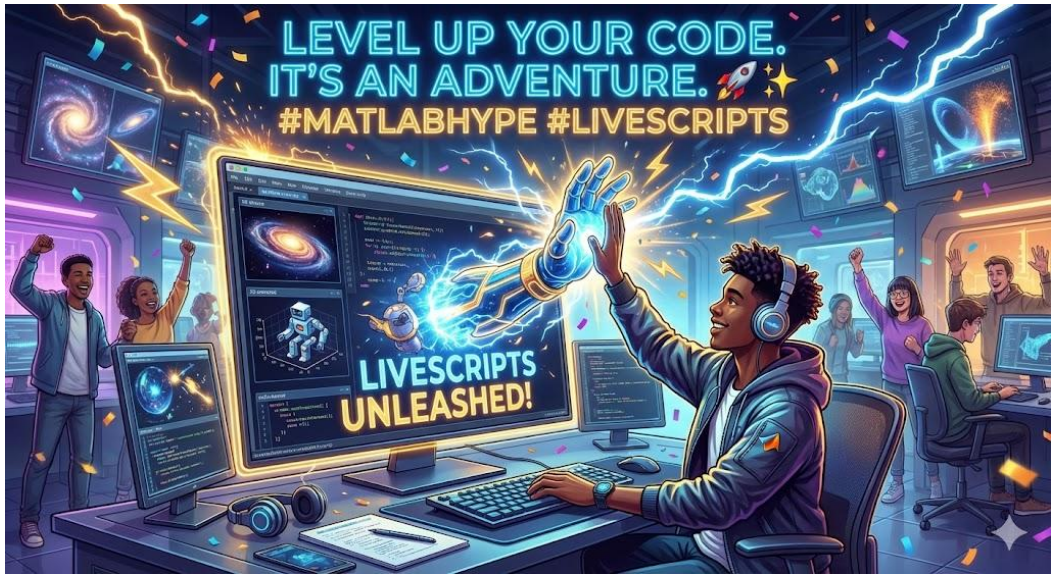
- Create figure and axes
- Modify object using handles to achieve desired appearance
- Use command `set(gca,'nextplot','replacechildren')` to ensure only children of axes object will be replaced upon next plot command (will not modify axes properties)

During loop

- Plot command to generate data on plot
- Modify object using handle to achieve desired appearance
- Use command `drawnow` to draw object, otherwise it will not be drawn until execution is complete (MATLAB optimization, as plotting is expensive) Alternatively, modify XData, YData, ZData properties of initial plot object Additionally, save sequence of plotting command as frames (`getframe`) and play back from MATLAB window (`movie`)



Livescript


LEVEL UP YOUR CODE.
IT'S AN ADVENTURE.
#MATLABHYPE #LIVESCRIPTS
LIVESCRIPTS
UNLEASHED!

Outline

- Key dataset: NHANES (National Health and Nutrition Examination Survey) dataset
  - Biostatistics datapoints: age, sex, BMI, height, weight, blood pressure, smoking, cholesterol, etc.,
  - There are ready-made teaching subsets with ~1,000 adults that are explicitly designed for coursework
- Bar charts: demo -> plot prevalence of obesity or smoking
- Histograms: demo -> plot empirical distribution of BMI, blood pressure, etc.
- Scatter plots: demo -> plot Blood Pressure vs Age, BMI vs Age
- Visualize 3d surfaces: demo -> plot interpolations of 3D surface
- Volume Visualizations: **key demo** -> MRI data: https://www.mathworks.com/help/MATLAB/visualize/techniques-for-visualizing-scalar-volume-data.html
- Plotting function animations: demo -> Plot ECG (heart rate monitor) and overlay sine function fitted animation
- Mesh visualizations: **key demo** -> Brain Mesh Visualization
  - Download a brain mesh
  - Download cortical thickness maps from neuromaps and Clinica
  - Visualize brain mesh with cortical thickness using Surfstat toolbox