

# **Lecture 4:**

# **Visualization**

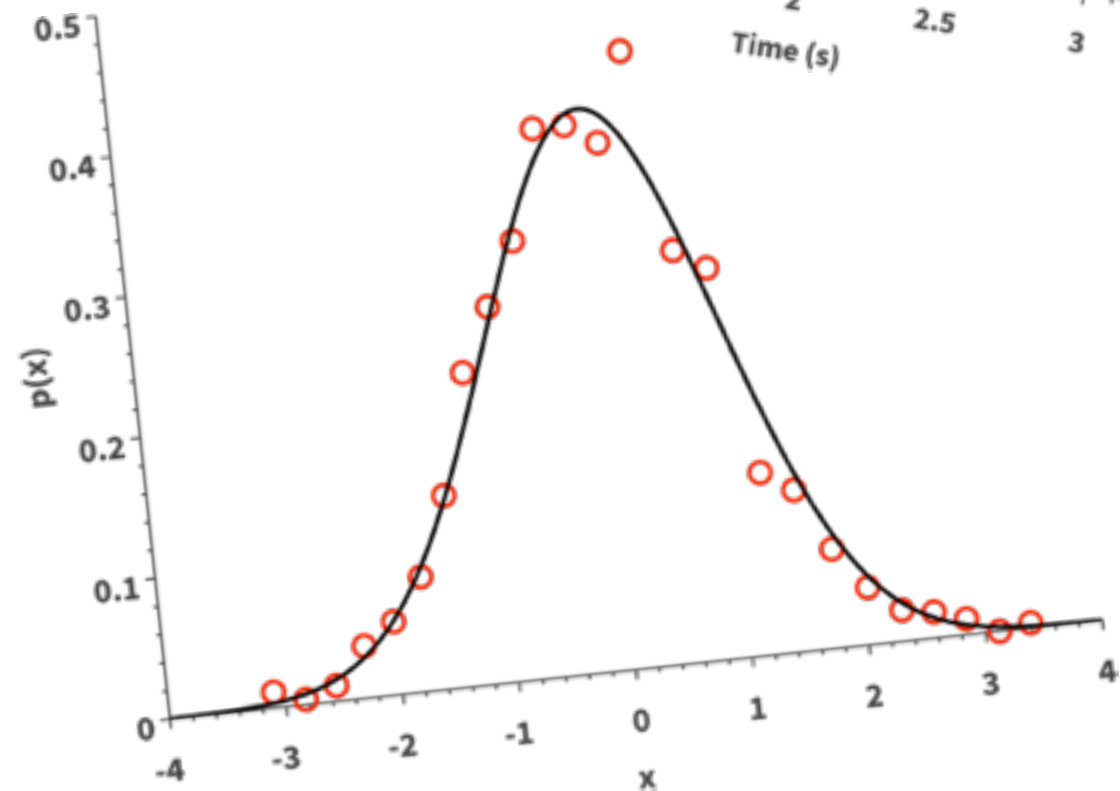
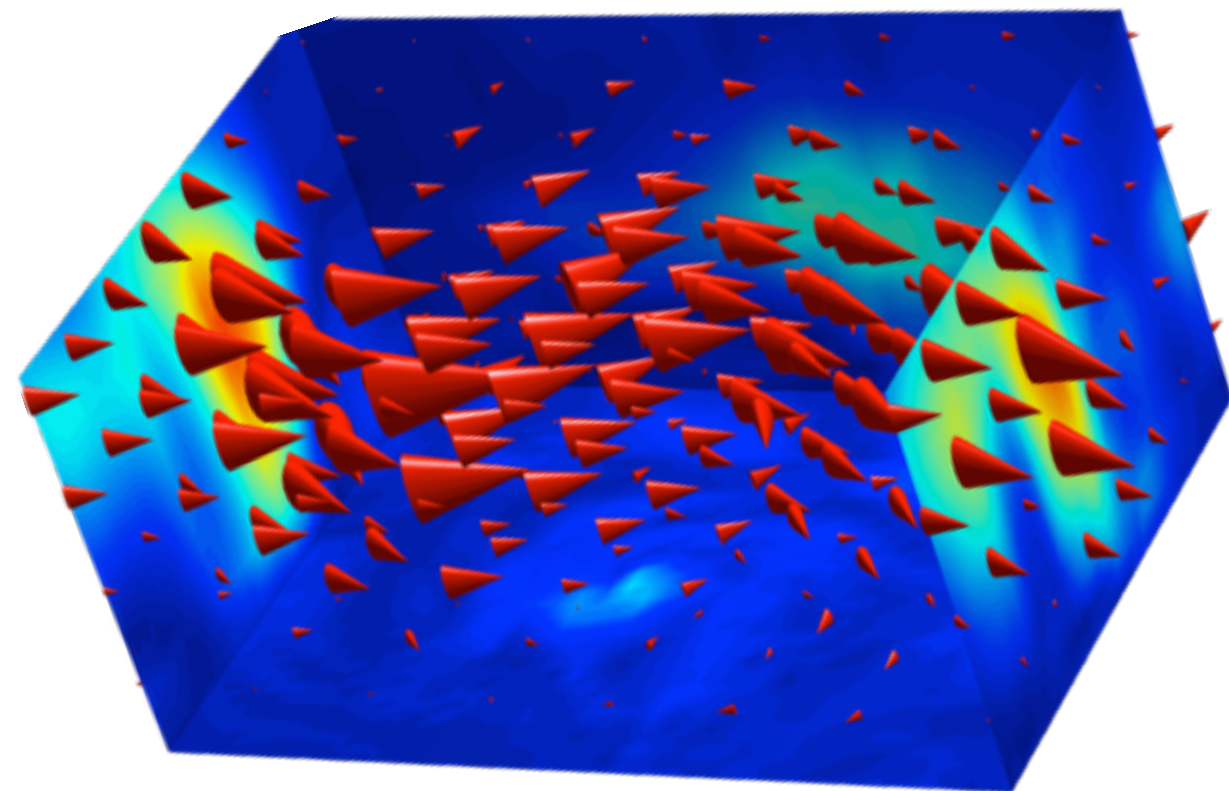
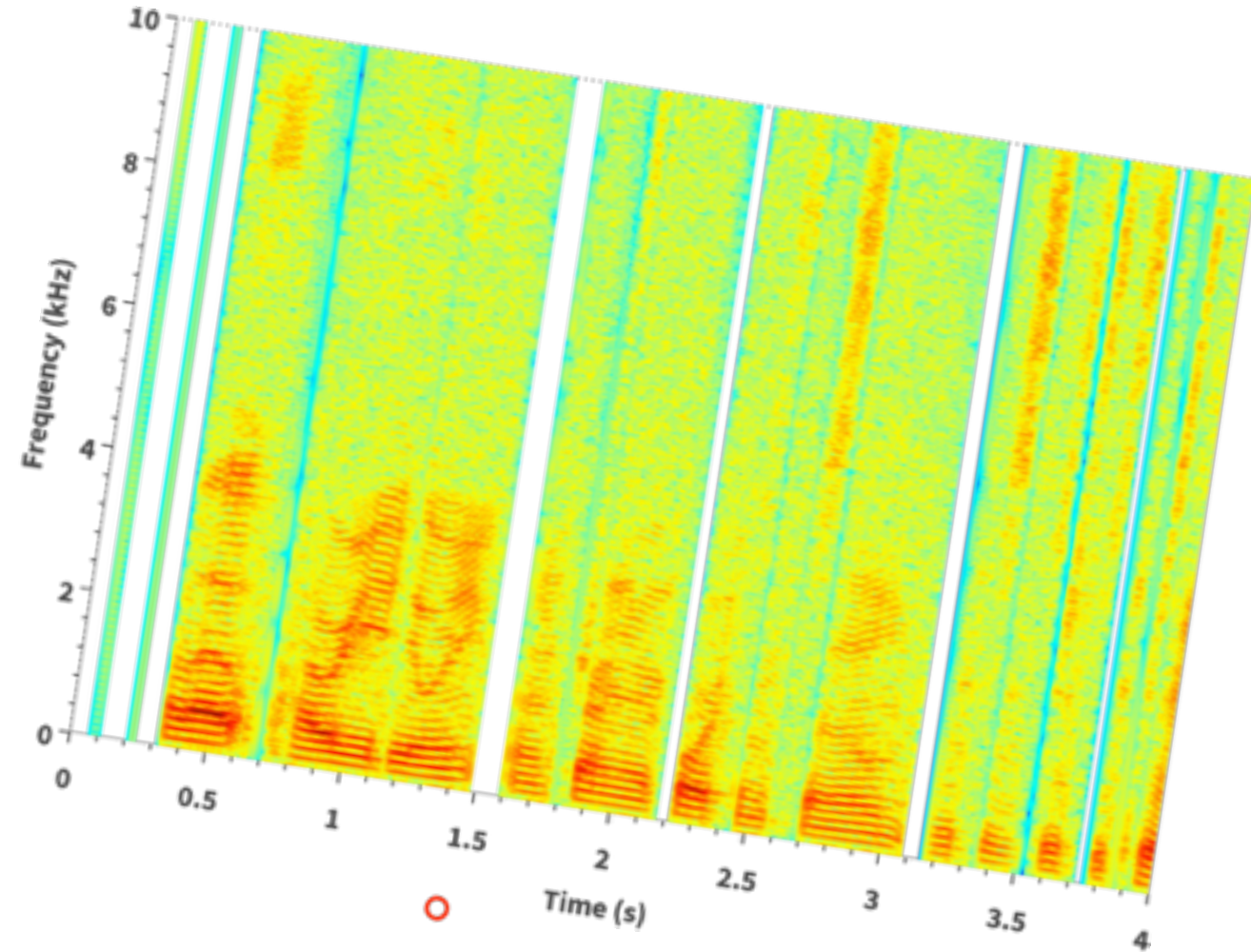
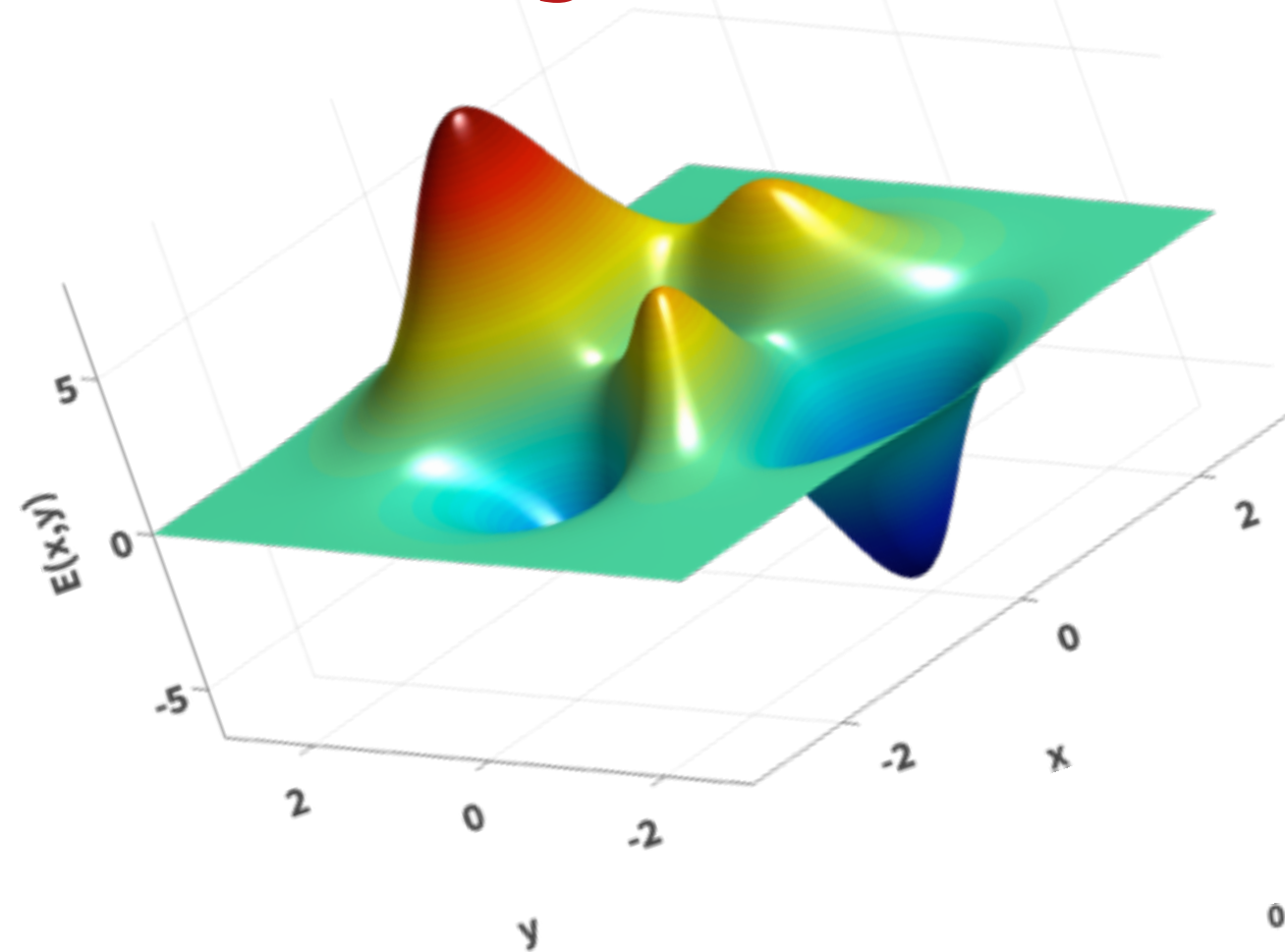
# Outline

- Basic plotting commands
- Types of plots
- Customizing plots graphically
- Color
- Figure formats
- Customizing plots programmatically

# Why visualization in Matlab?

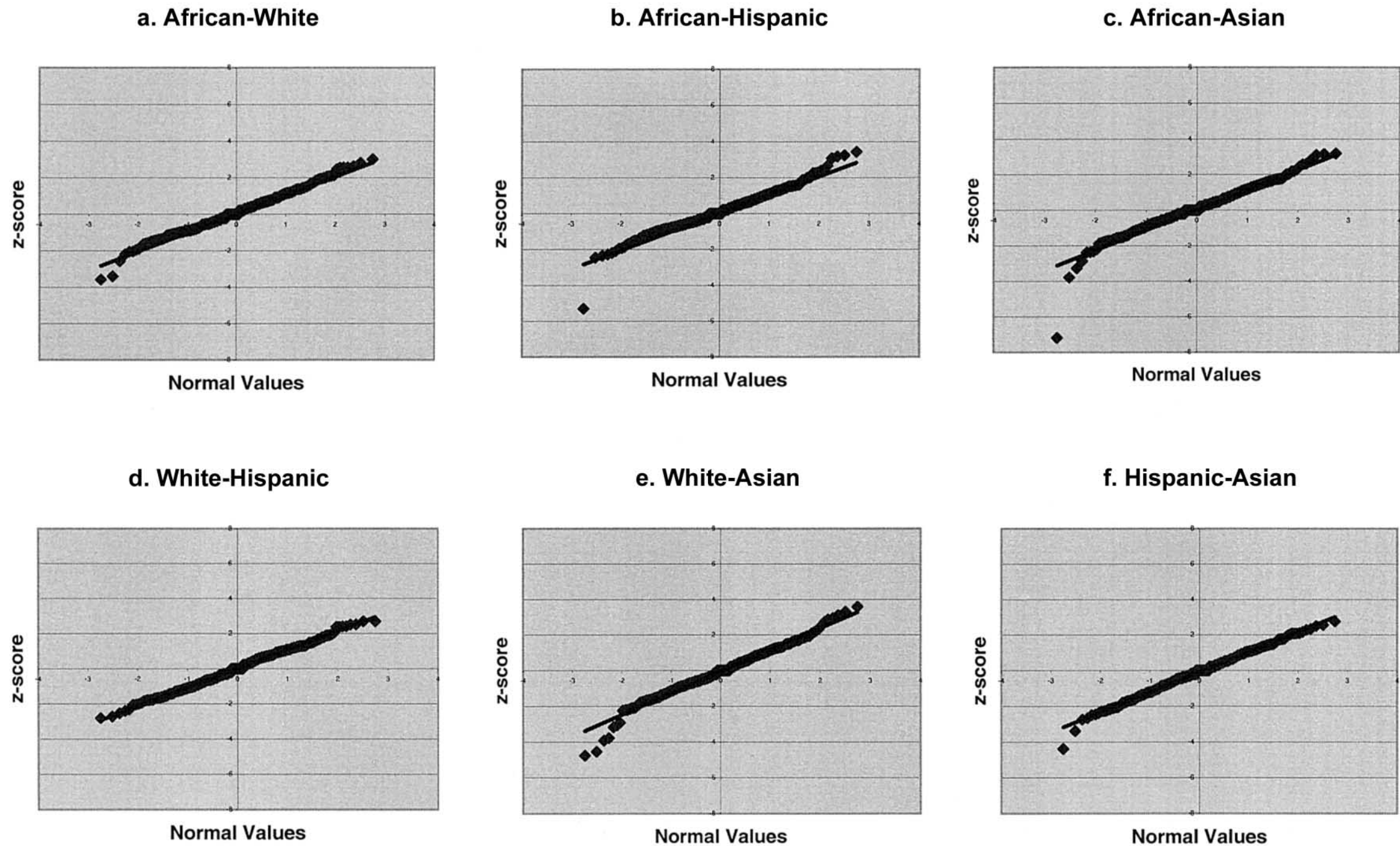
Matlab is **flexible** enough to let you quickly visualize data, and **powerful** enough to give you complete control over the final product

# Why visualization in Matlab?





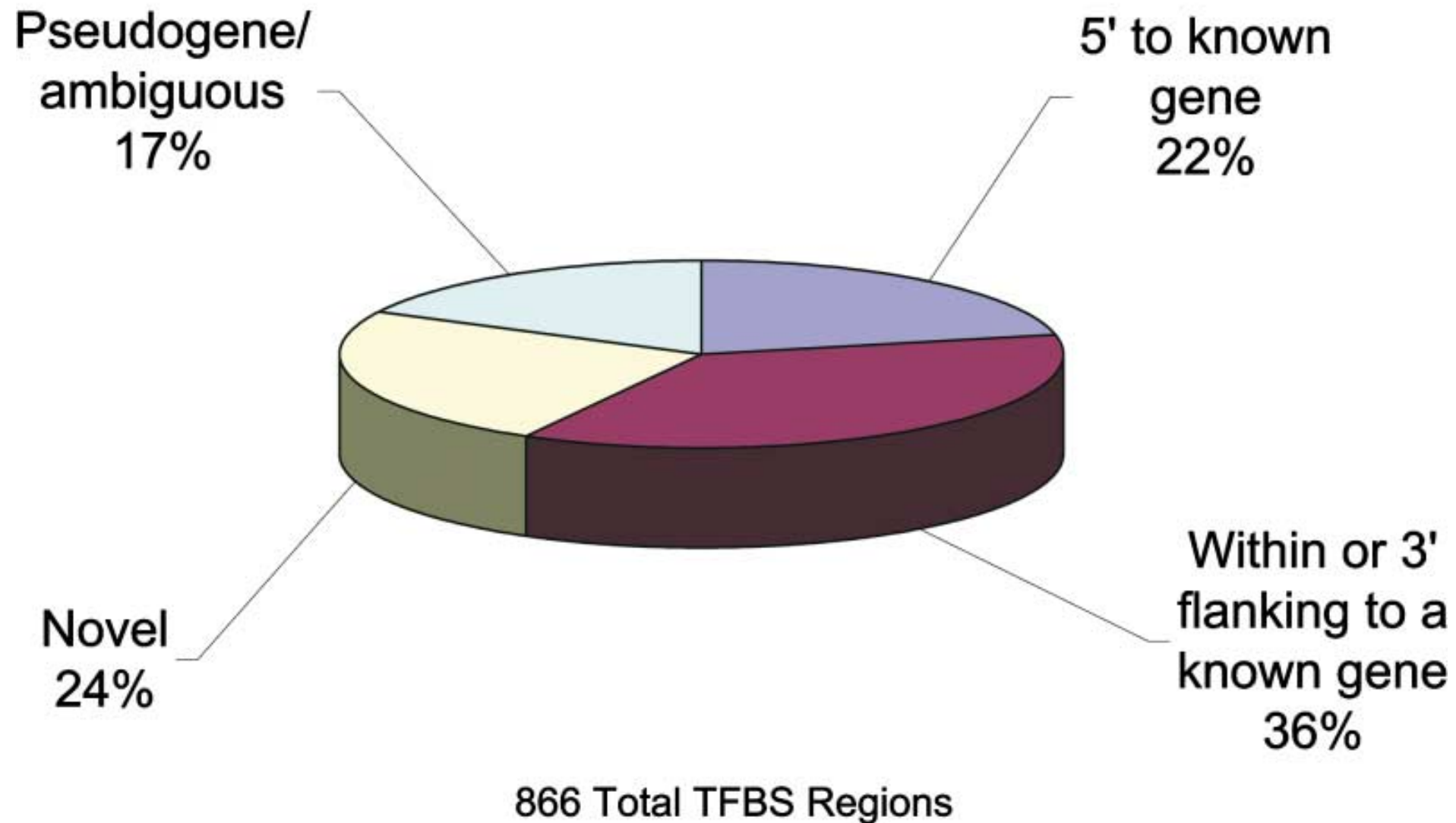
# Poor Graphs



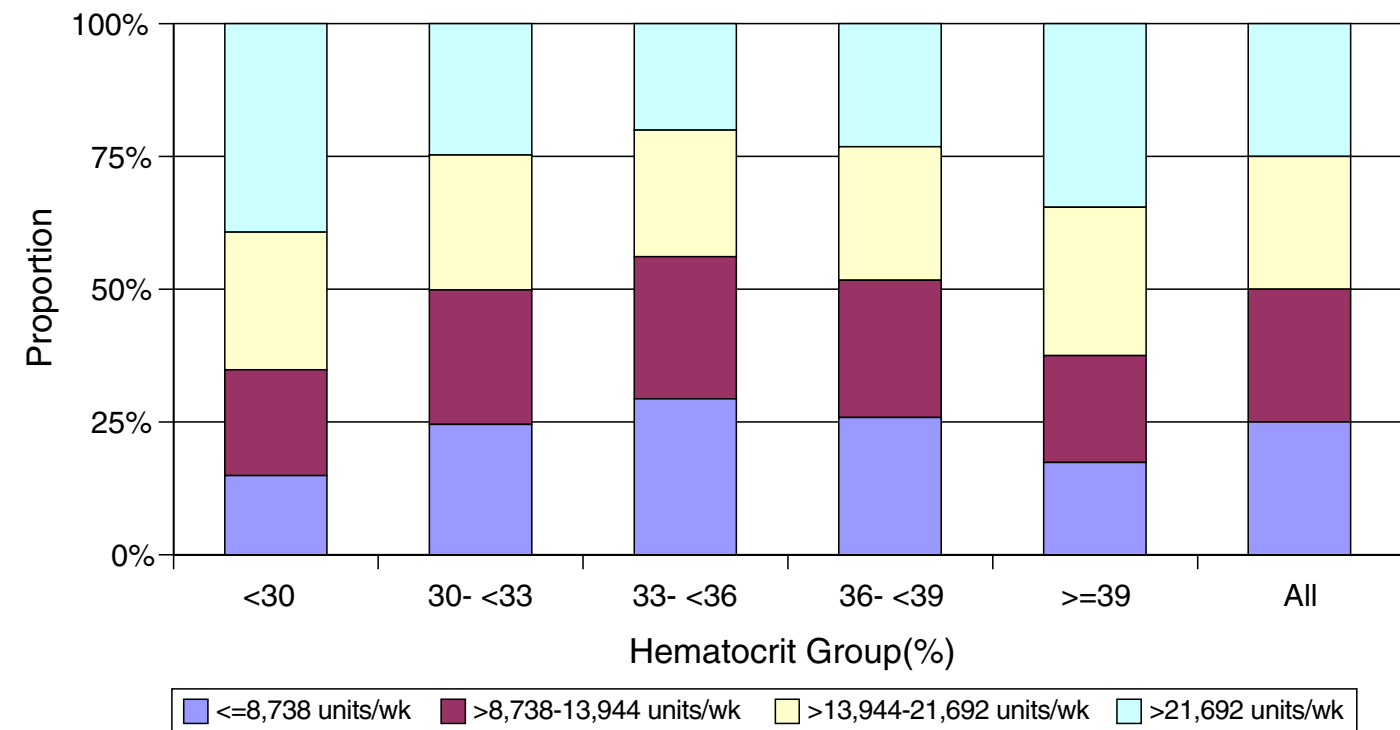
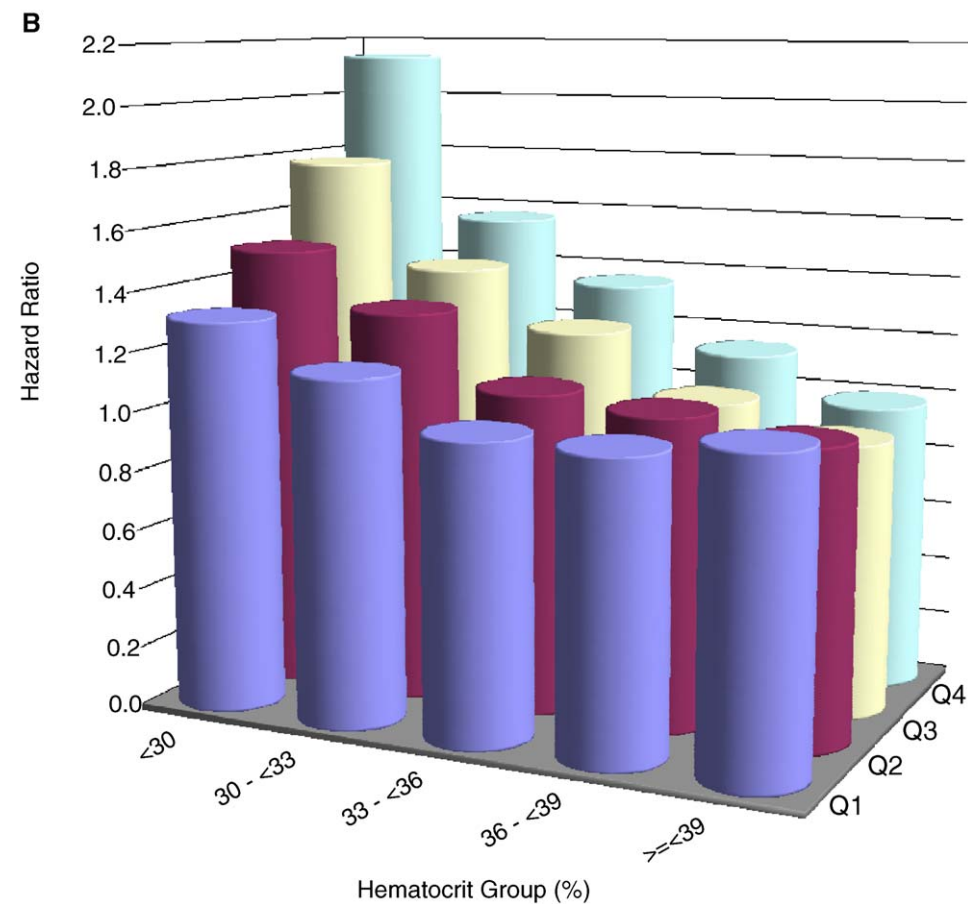
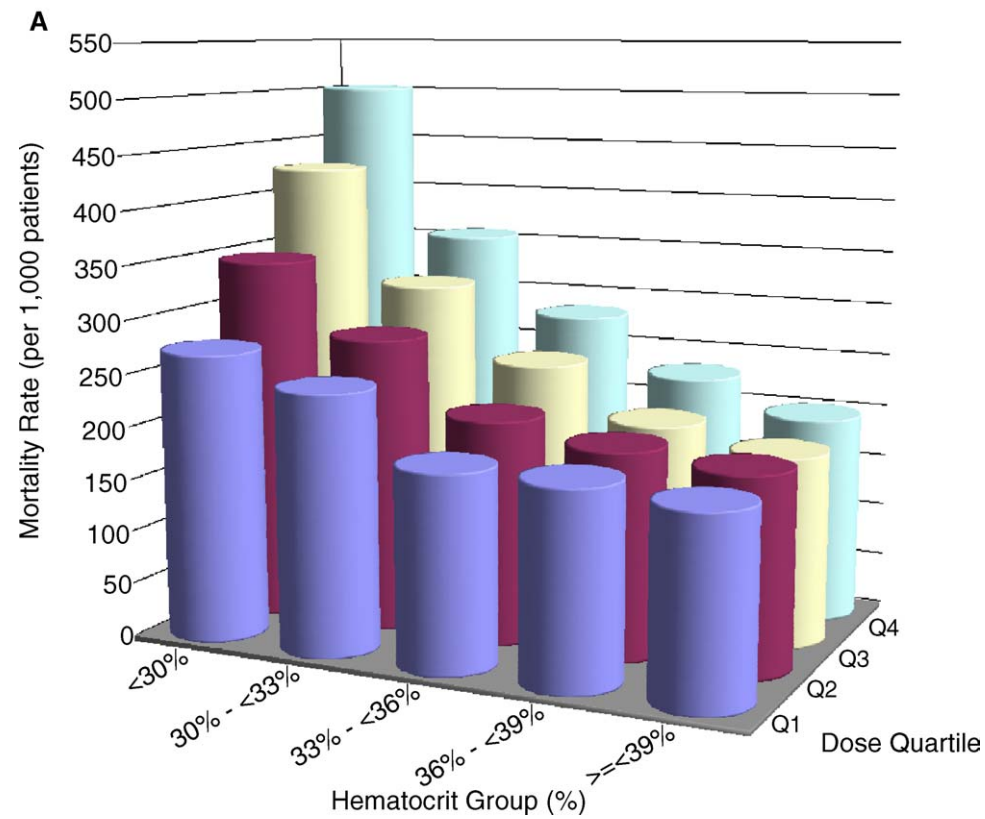
**Figure 1** Q-Q plots of Z scores for individual interval-length differences between racial/ethnic groups. *a*, African Americans versus whites. *b*, African Americans versus Hispanics. *c*, African Americans versus Asians. *d*, Whites versus Hispanics. *e*, Whites versus Asians. *f*, Hispanics versus Asians.

# Poor Graphs

## Distribution of All TFBS Regions



# Poor Graphs



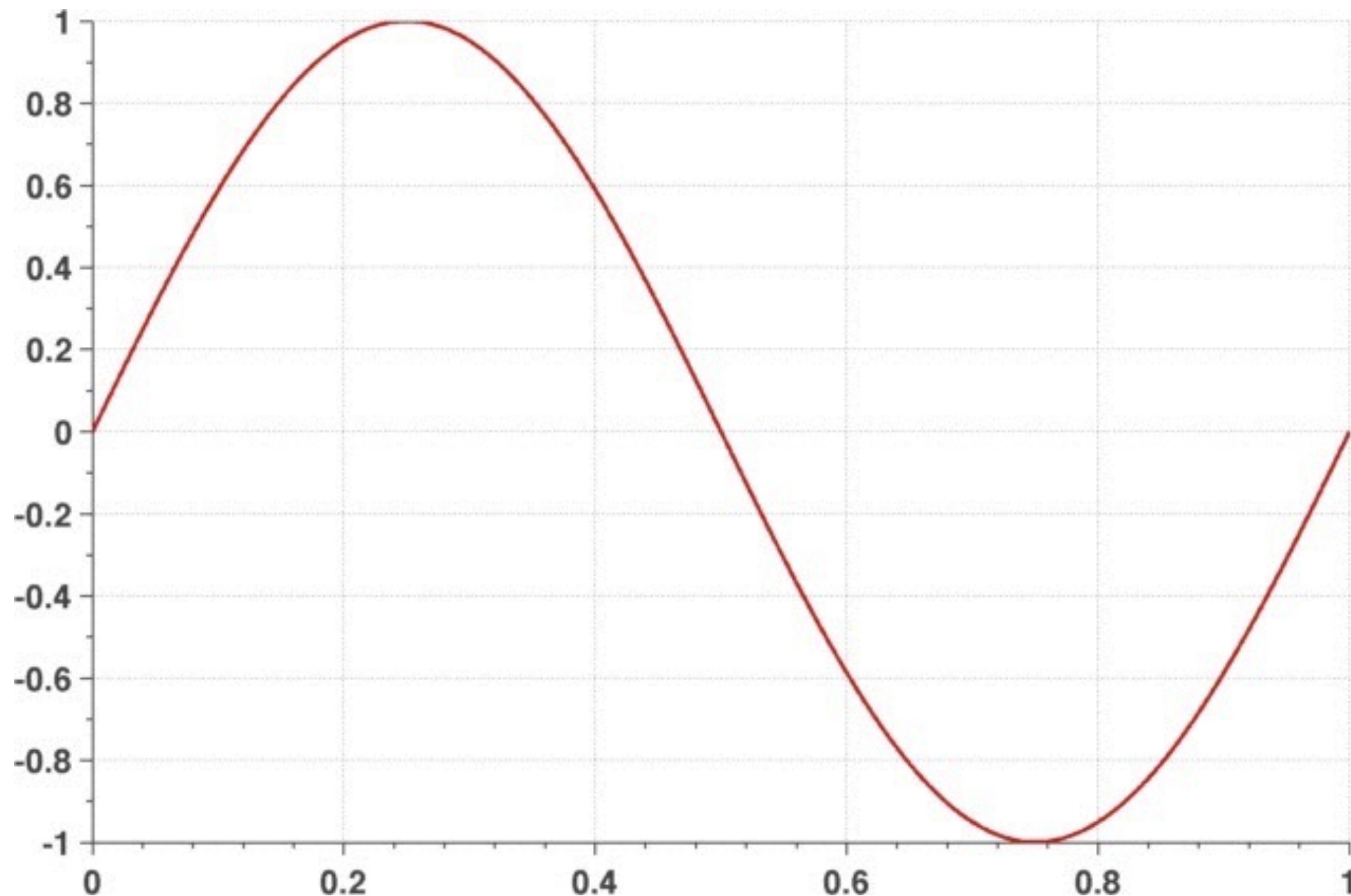
# Types of Plots

- 2D Visualization
  - `plot` (line plots)
  - `image/imagesc` (images)
  - `bar` (histograms)
  - `scatter` (scatter plots)
- 3D Visualization
  - `surf/mesh` (surfaces)
  - `plot3` (lines)



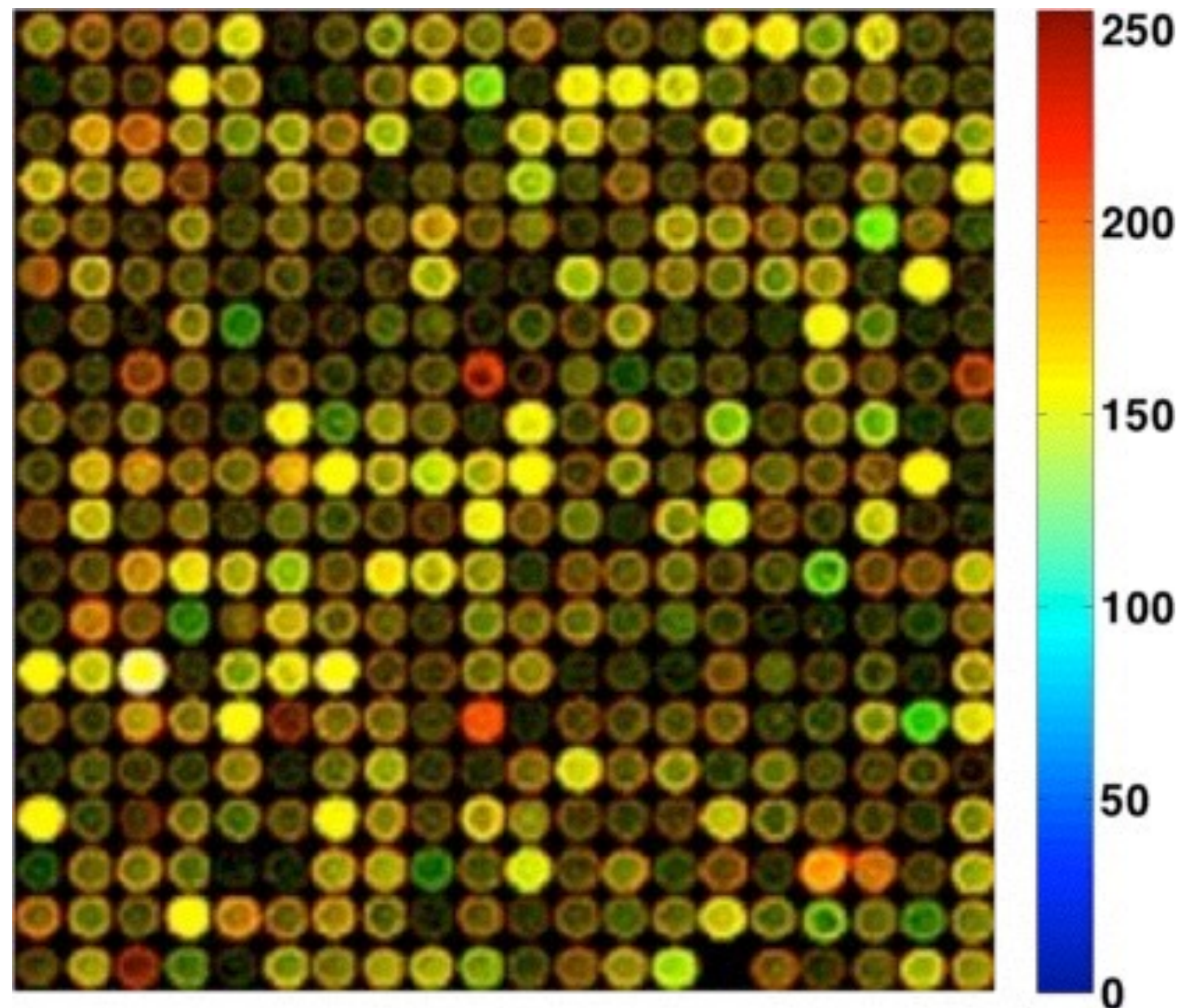
# plot

- Syntax: `plot(x,y)` plots points in the vector `y` against points in the vector `x`



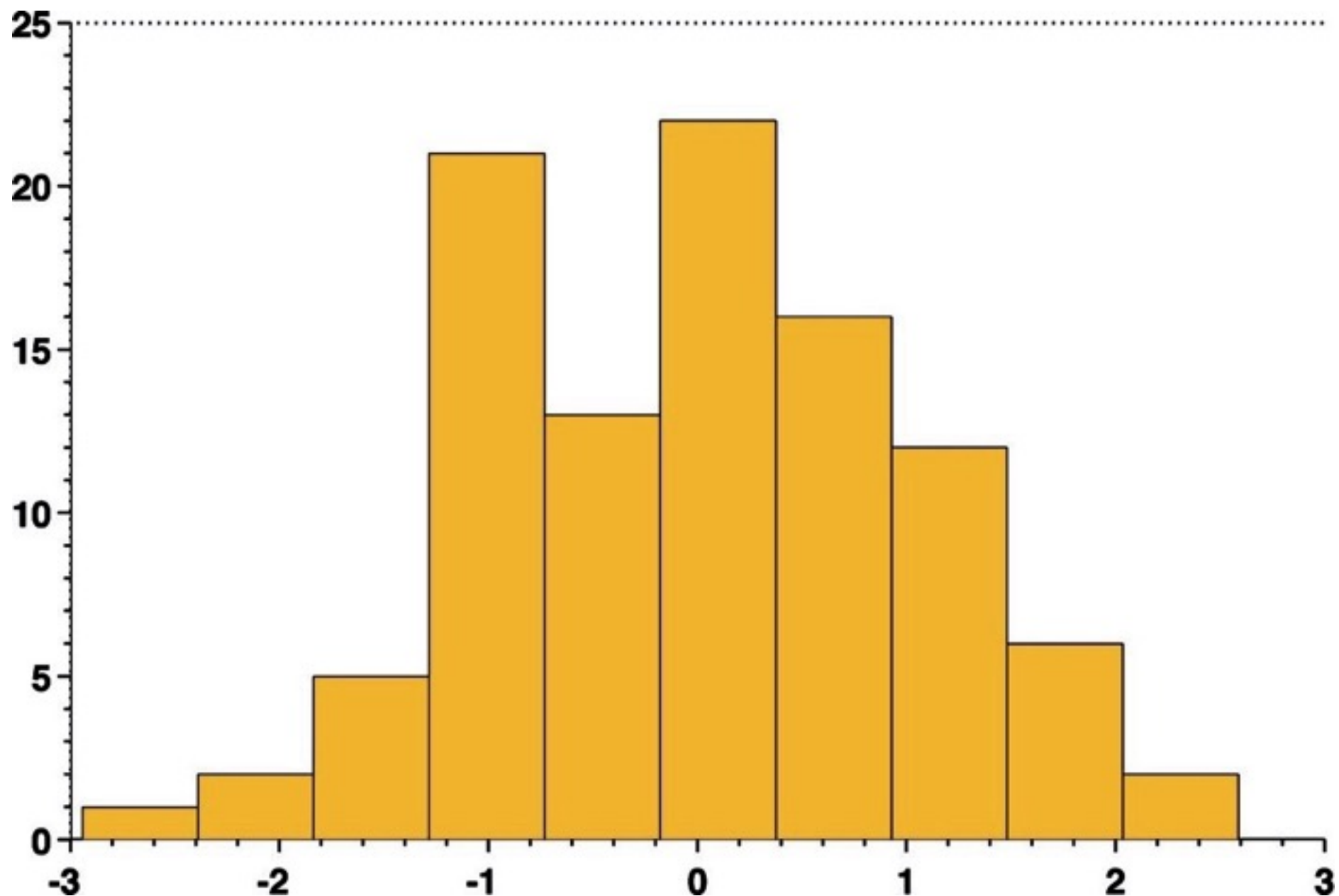
# image/imagesc

- Syntax: `image(C)` plots the values stored in the matrix `C` as an image



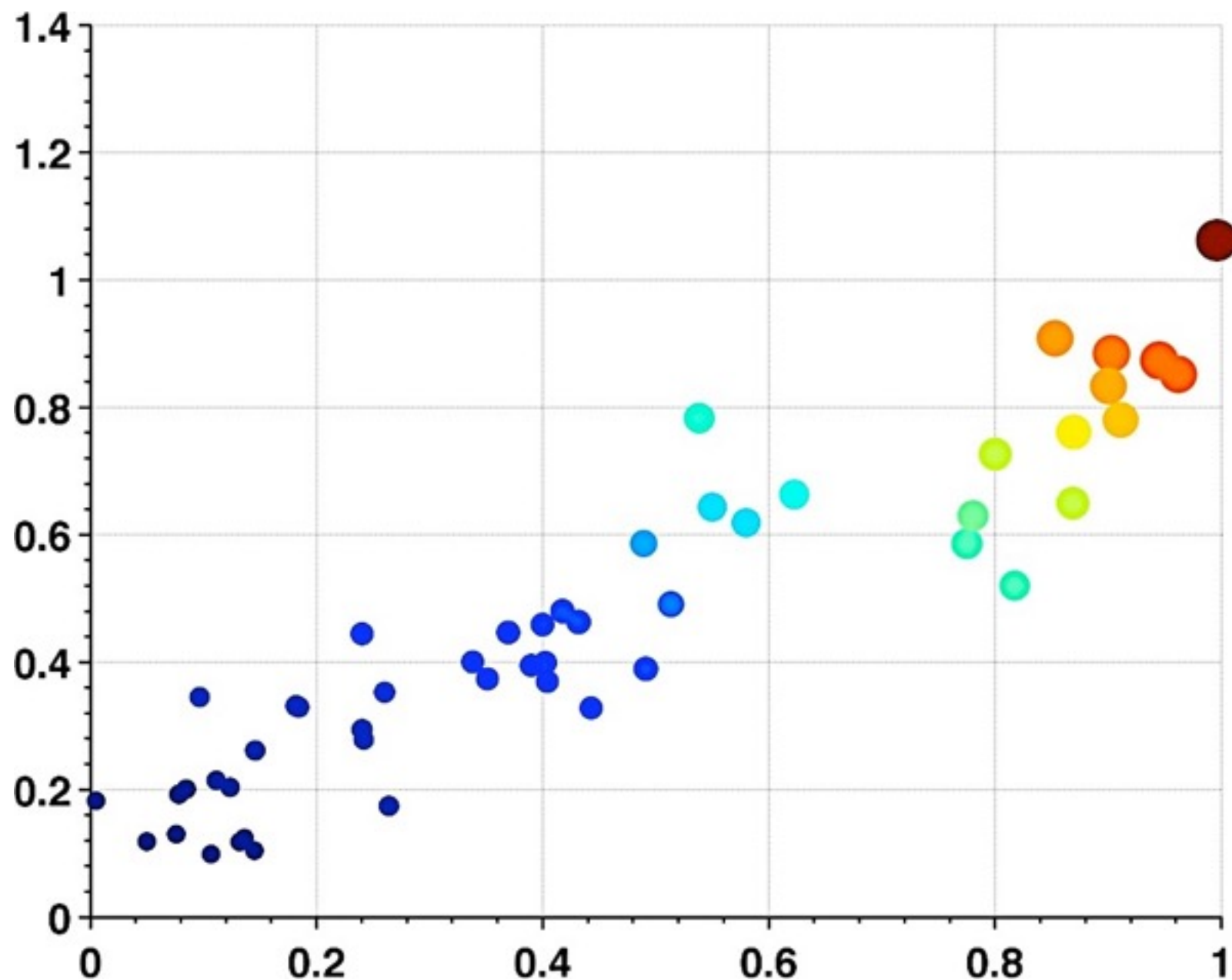
# hist/bar

- Syntax: `hist(y)` plots a histogram of the values in `y`,  
`bar(x,y)` plots bars at the points given by `(x,y)`



# scatter

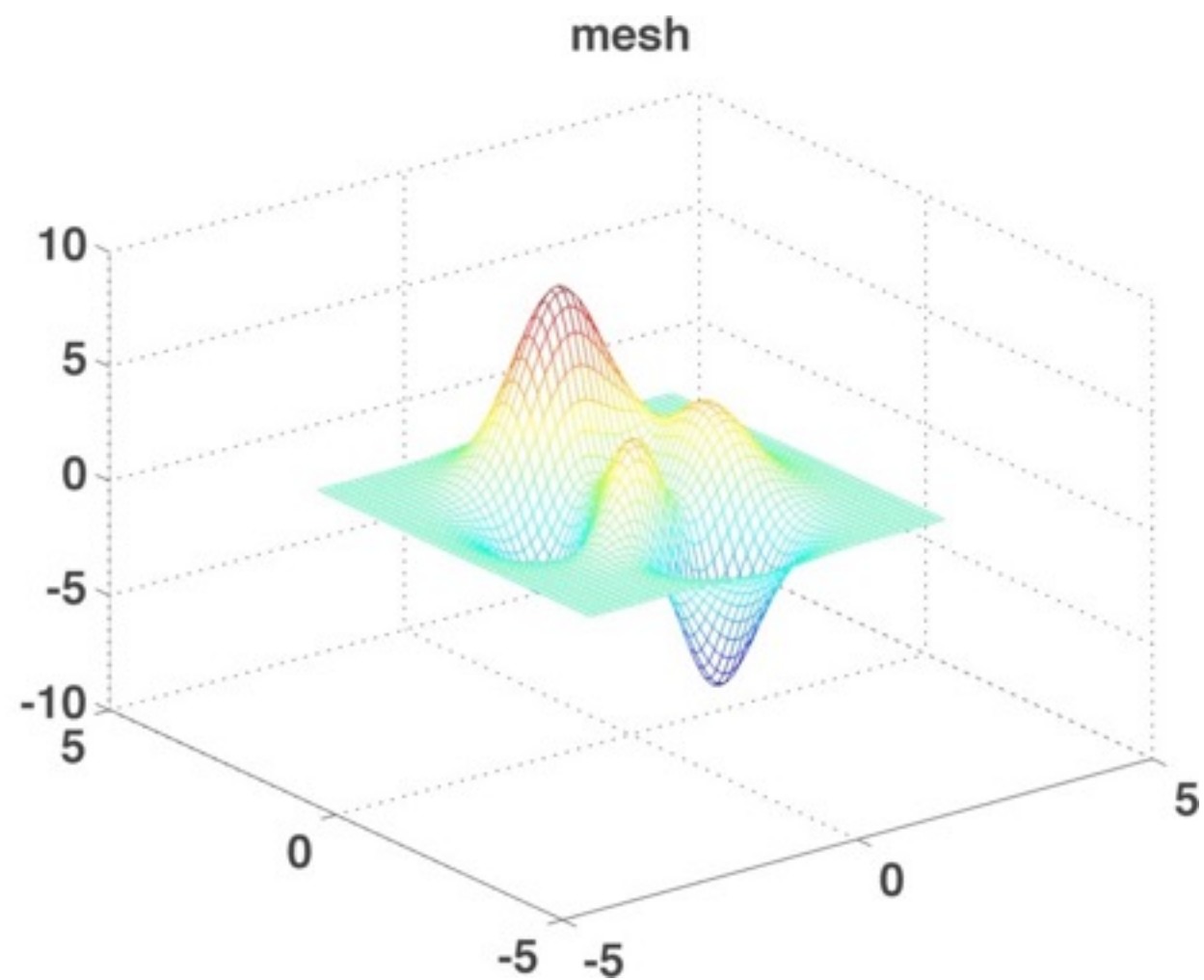
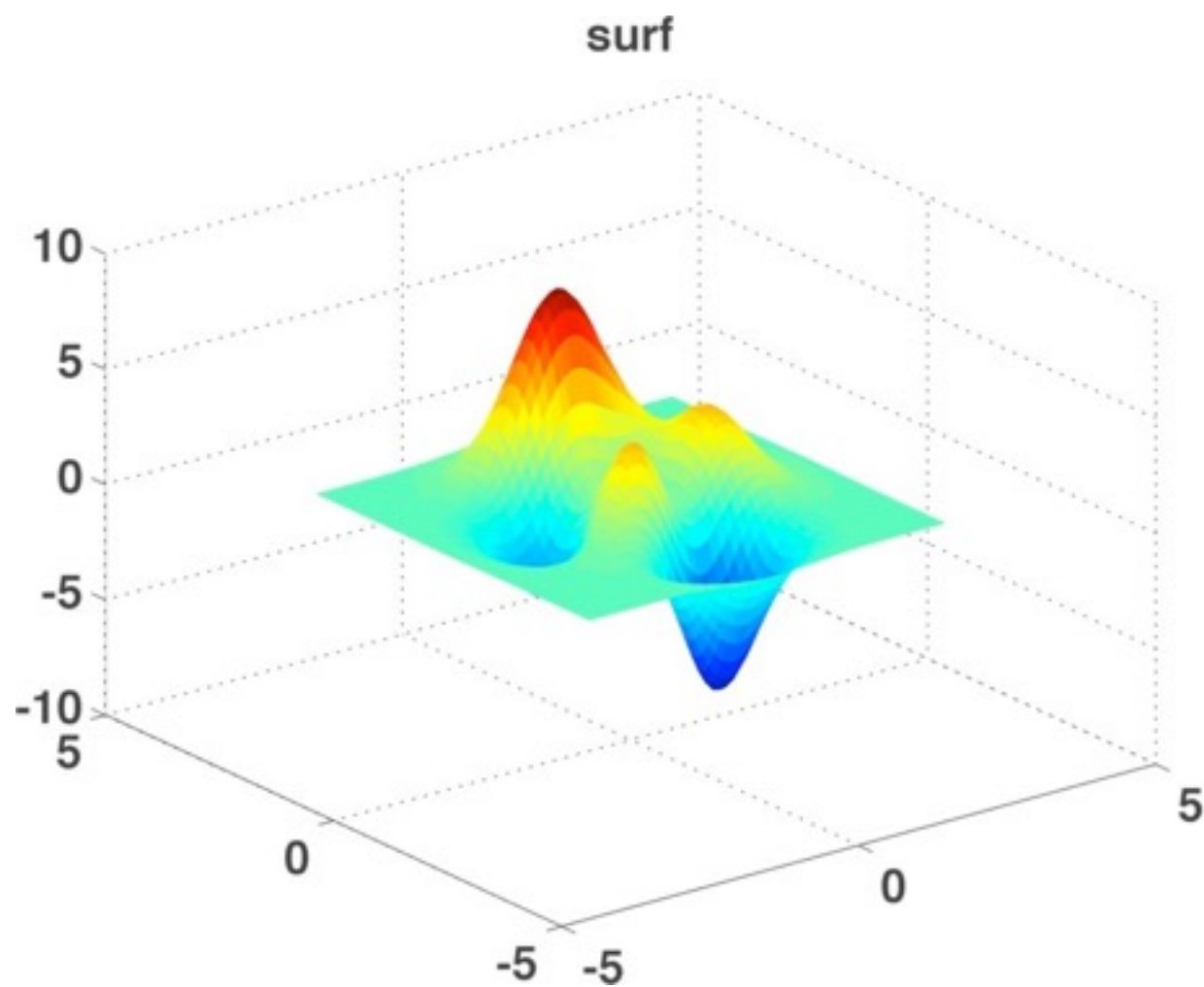
- Syntax: `scatter(x,y,s,c)` lets you specify the size (s) and color (c) of each point given by (x,y)





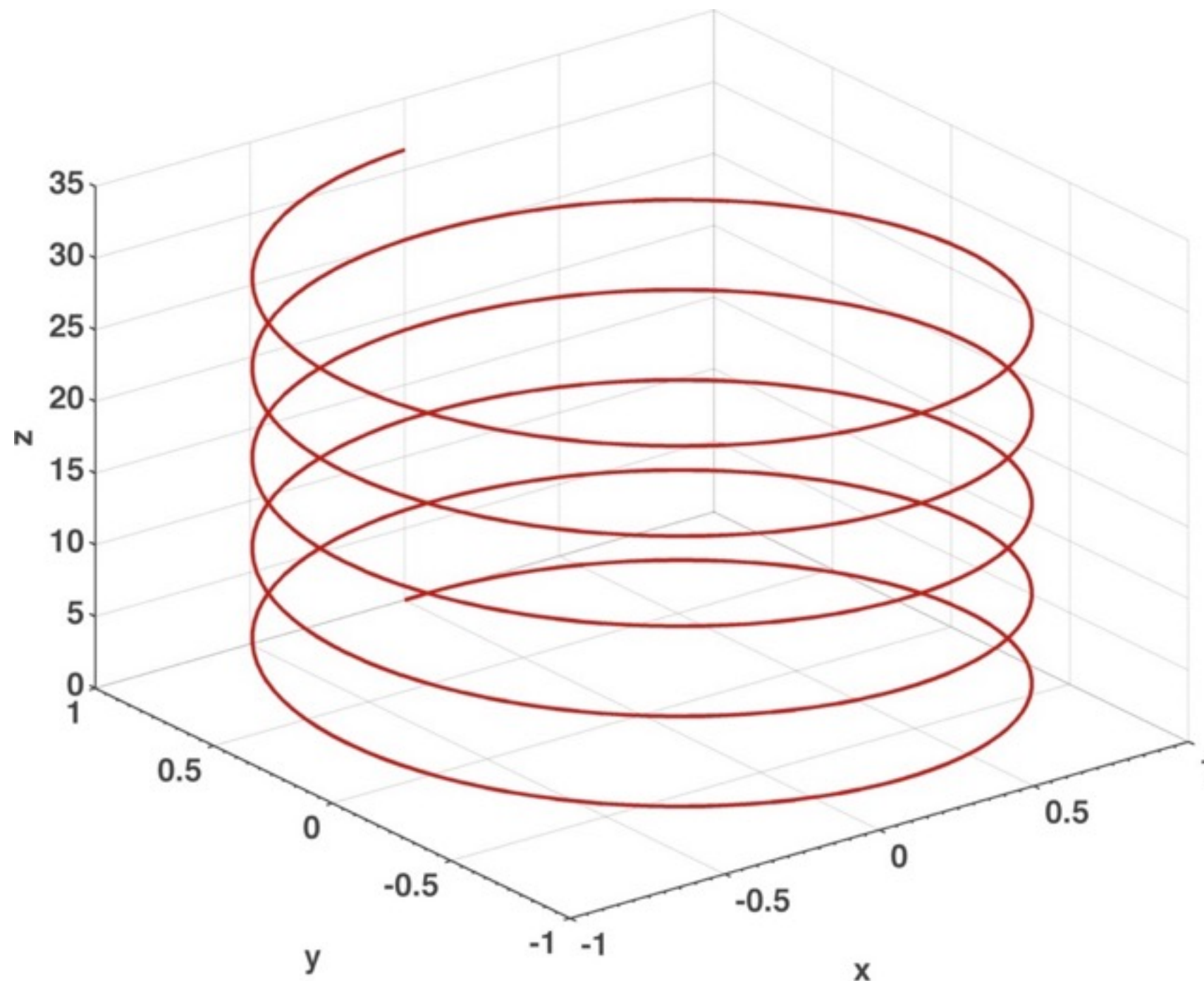
# surf & mesh

- Syntax: `surf(x,y,z)` and `mesh(x,y,z)` are used to visualize a surface in three dimensions



# plot3

- Syntax: `plot3(x,y,z)` plot points in 3D



# Demo: Plot Types

<http://www.mathworks.com/help/matlab/2-and-3d-plots.html>

# subplots

- the 'subplot' command let's you plot multiple plots on one figure
- syntax: subplot(nRows, nCols, index)

(Figure 1)

subplot(1,3,1)

subplot(1,3,2)

subplot(1,3,3)



# subplots

- the 'subplot' command let's you plot multiple plots on one figure
- syntax: subplot(nRows, nCols, index)

(Figure 2)

subplot(3,2,1)

subplot(3,2,4)

subplot(3,2,5)

subplot(3,2,2)

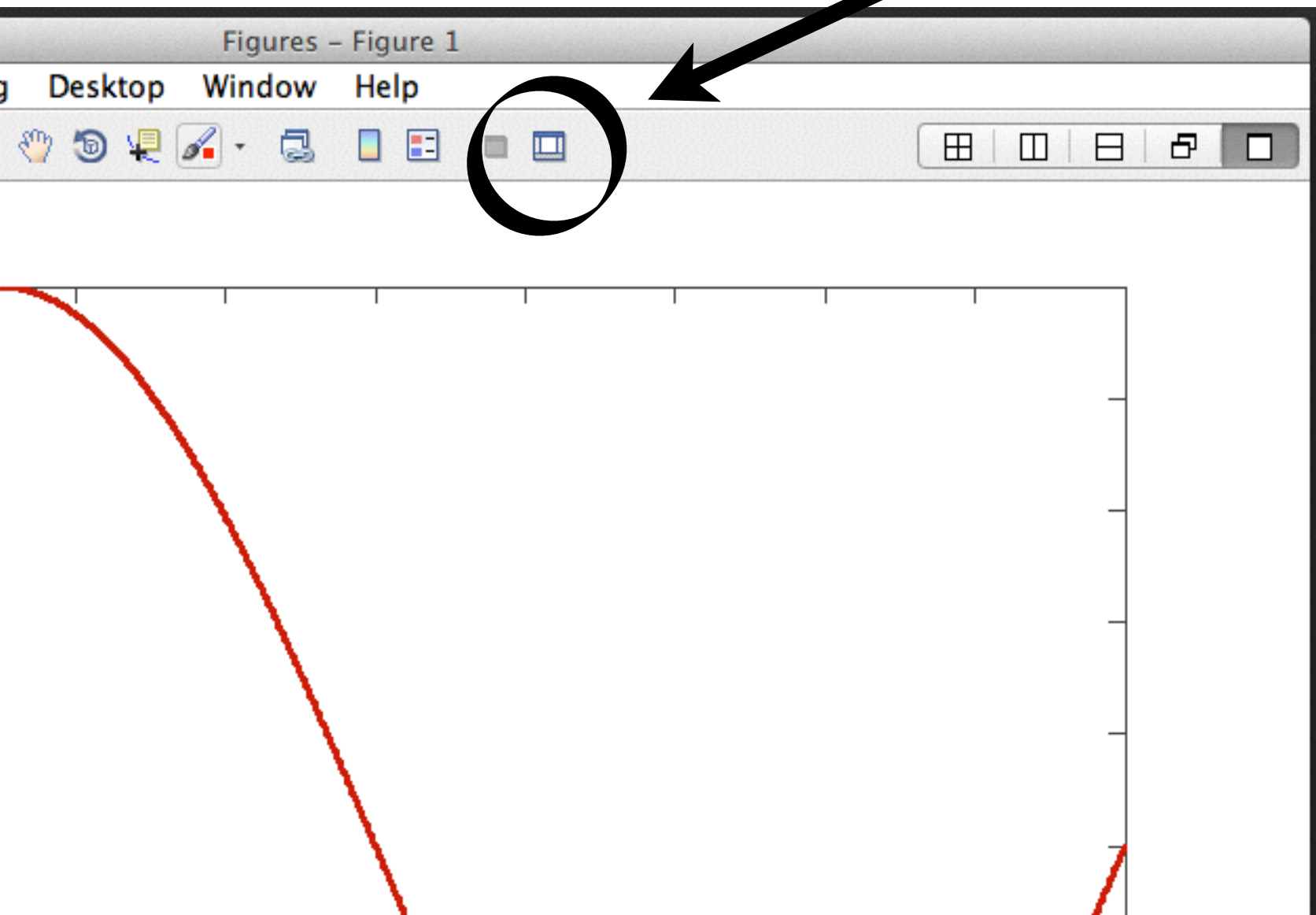
subplot(3,2,4)

subplot(3,2,6)

# Demo: Subplots

# Customizing Graphs Graphically

**Plot Tools**



# **Demo: Customizing Graphically**



# Color in Matlab

Matlab has 8 built-in colors:

Black (k), Red (r), Blue (b), Green (g),  
Cyan (c), Magenta (m), Yellow (y), White (w)

We can specify other colors using RGB (red, green blue) notation:

blue = [0 0 1]

green = [0 1 0]

gray = [0.2 0.2 0.2]

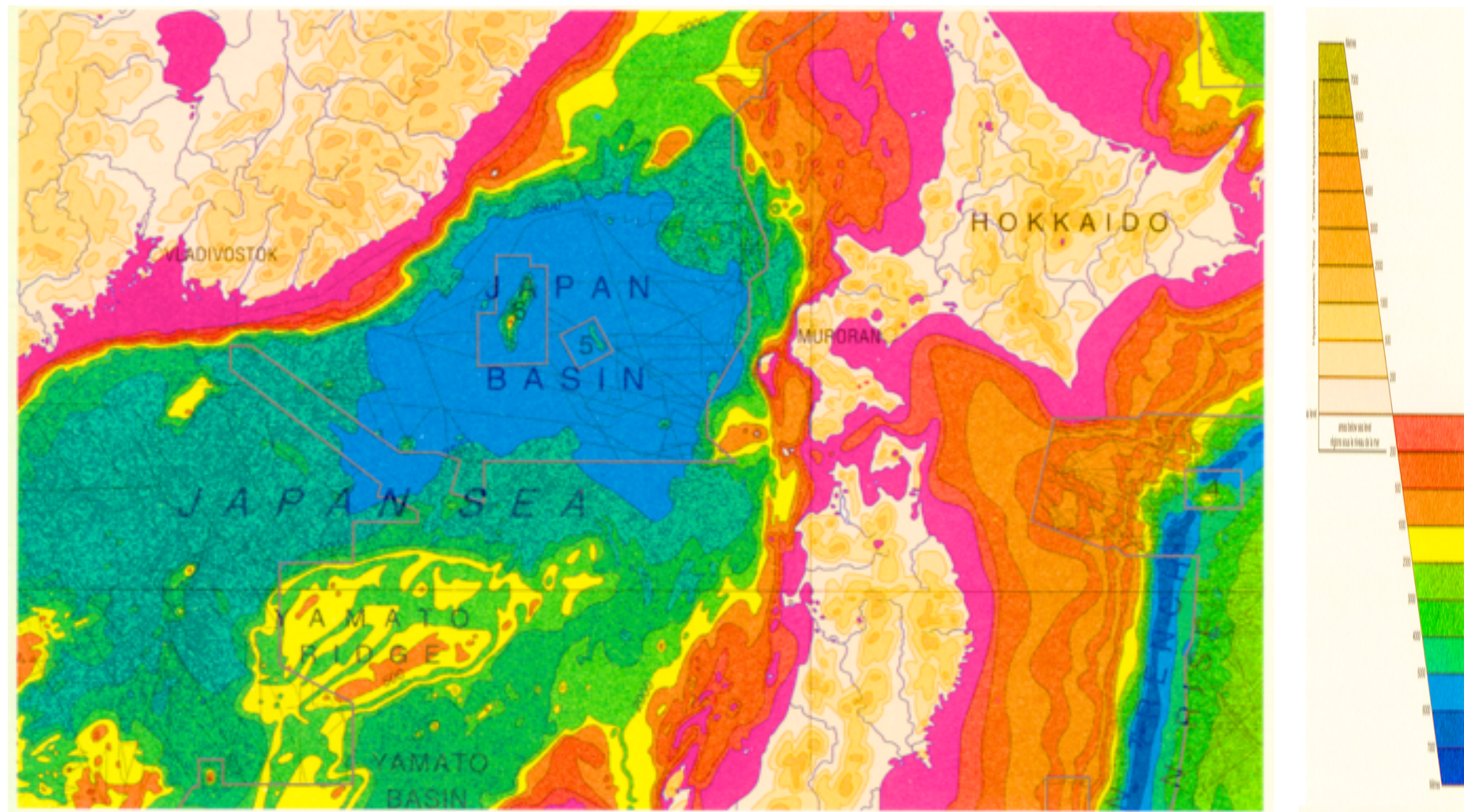
# Colormaps

Colormaps are used to specify how data gets mapped onto different colors.

Matlab has a few built-in colormaps, but you can also specify your own!

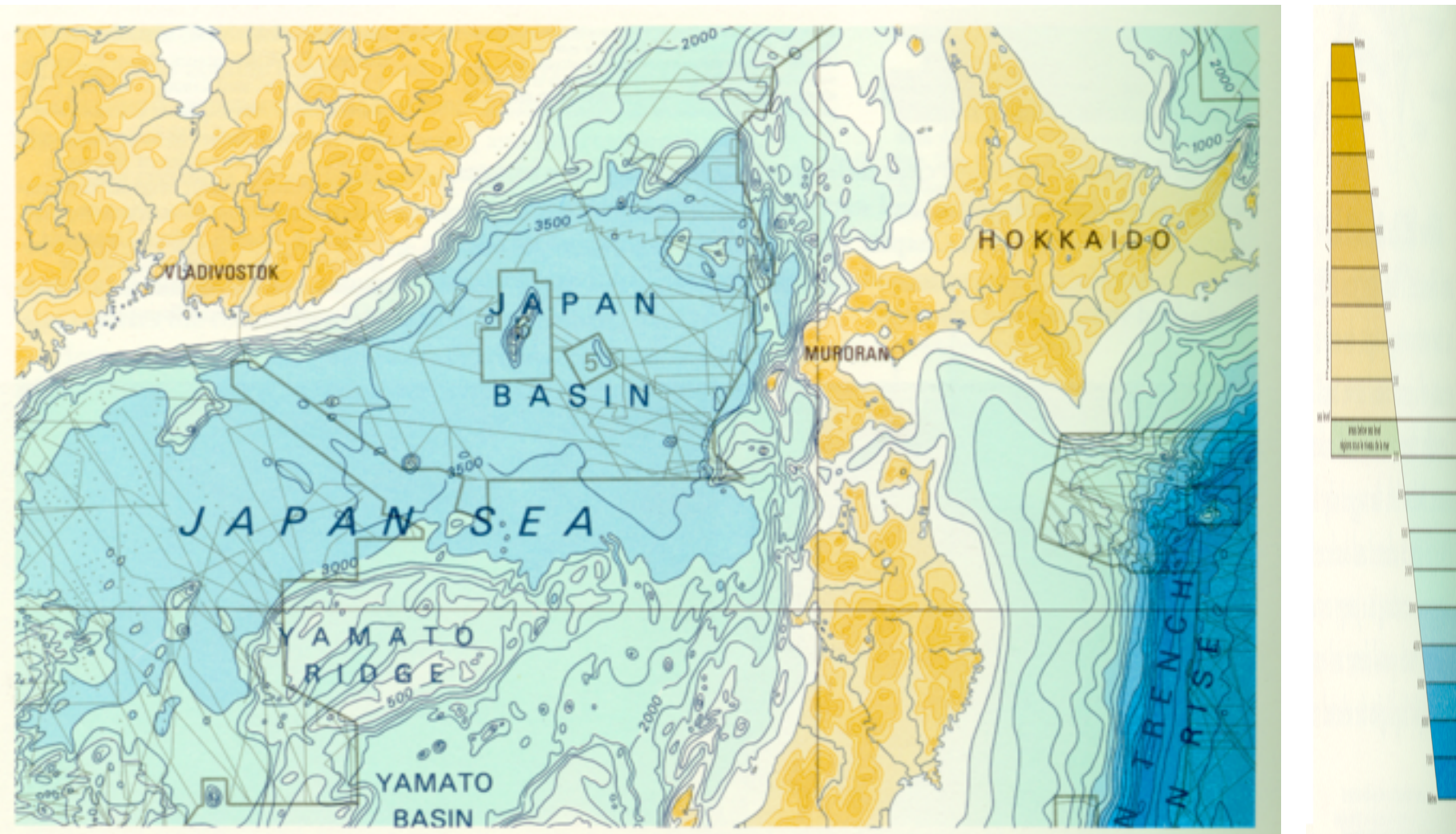


# Why are colormaps important?



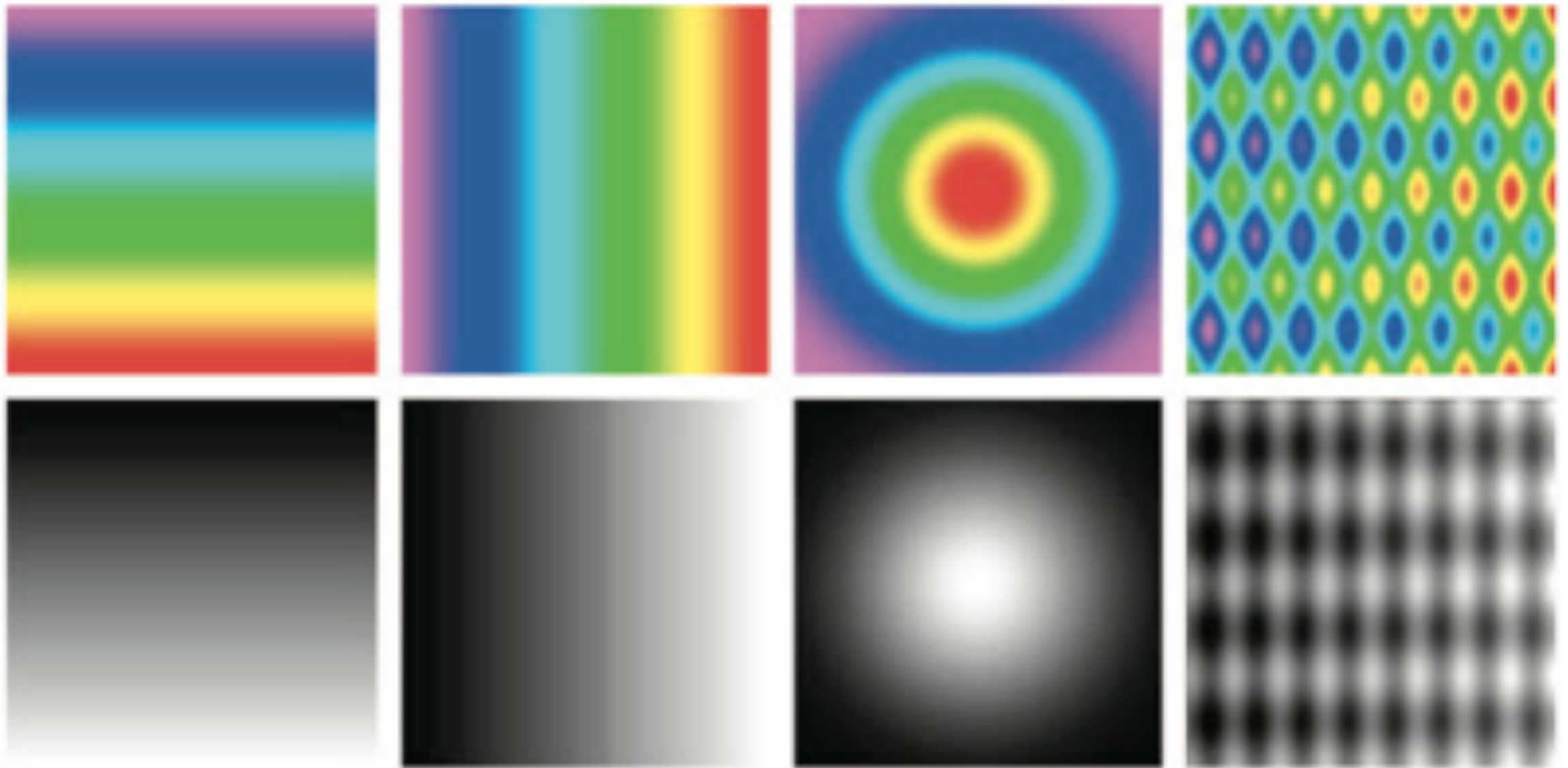


# Much better!





# Avoid the default colormap (jet)



# Demo: Color

# Figure formats

Matlab saves figures using it's own **.fig** format.

If we want to share our figures with others, or put them in a paper, we will export to other formats, including:

**JPG, PNG, EPS, PDF, TIFF**

# Bitmap vs. Vector graphics

Two main classes of image formats: **bitmap** vs. **vector** graphics

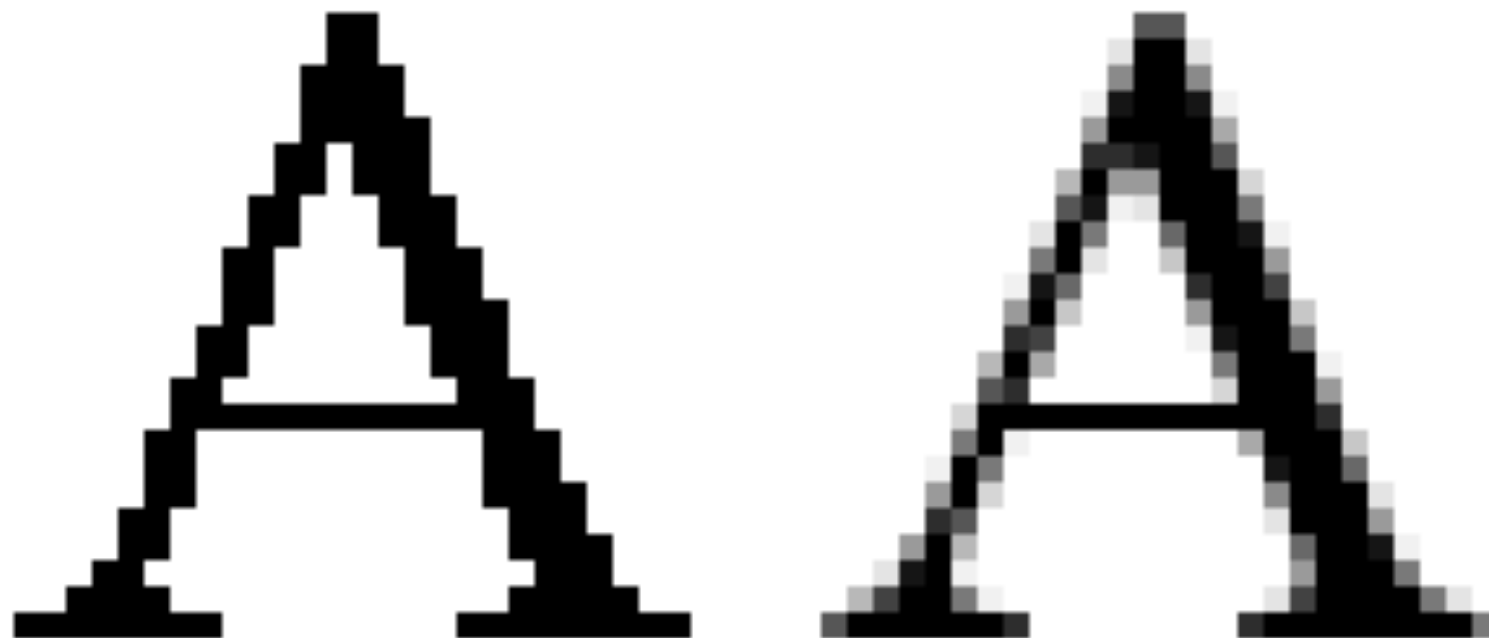
Bitmap (jpg, png):

- Fixed image sizes
- Best for actual images (pictures of stuff)

Vector (eps):

- Variable image sizes
- Best for line / bar graphs, scatter plots, etc.

# Aliasing

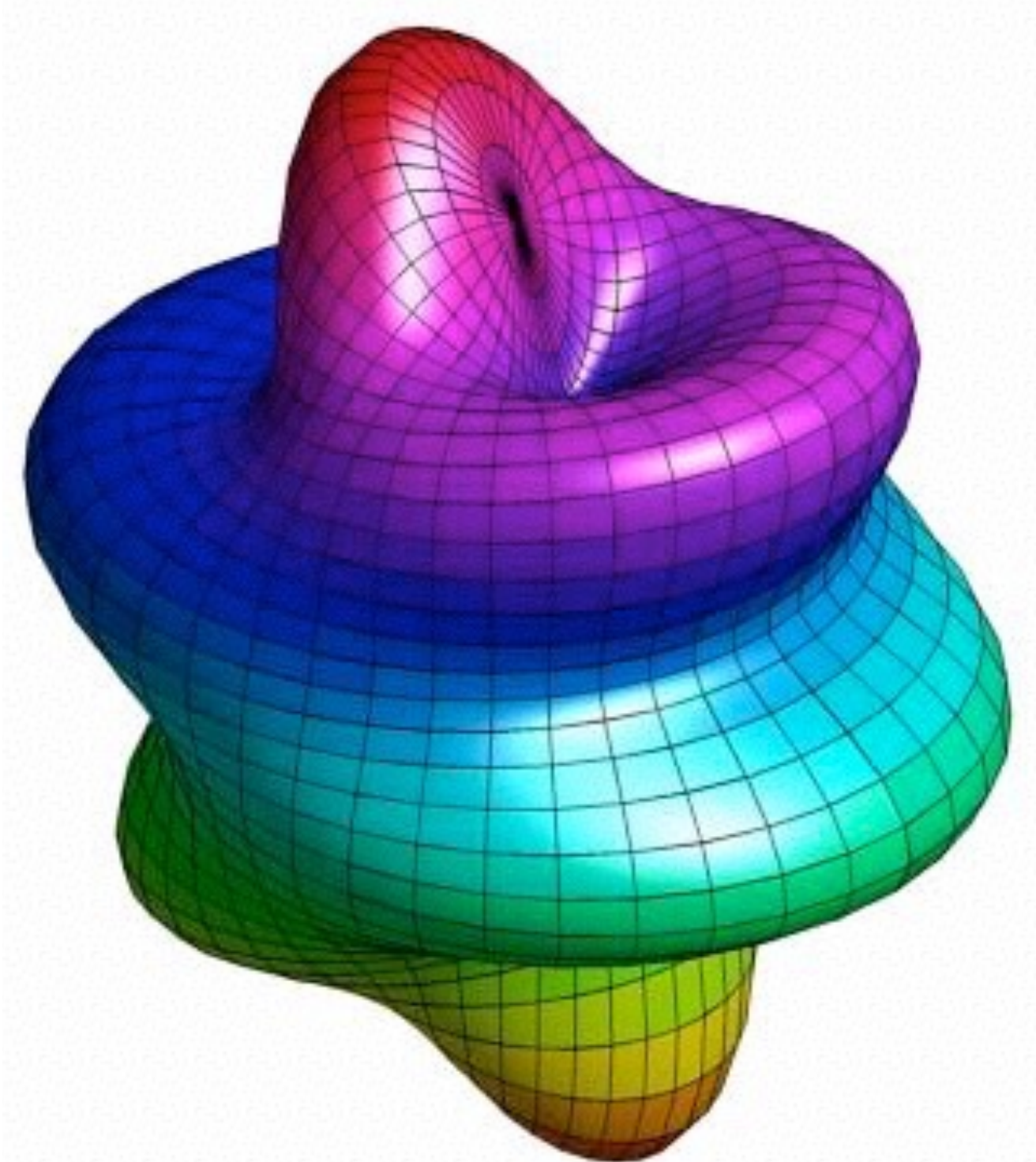
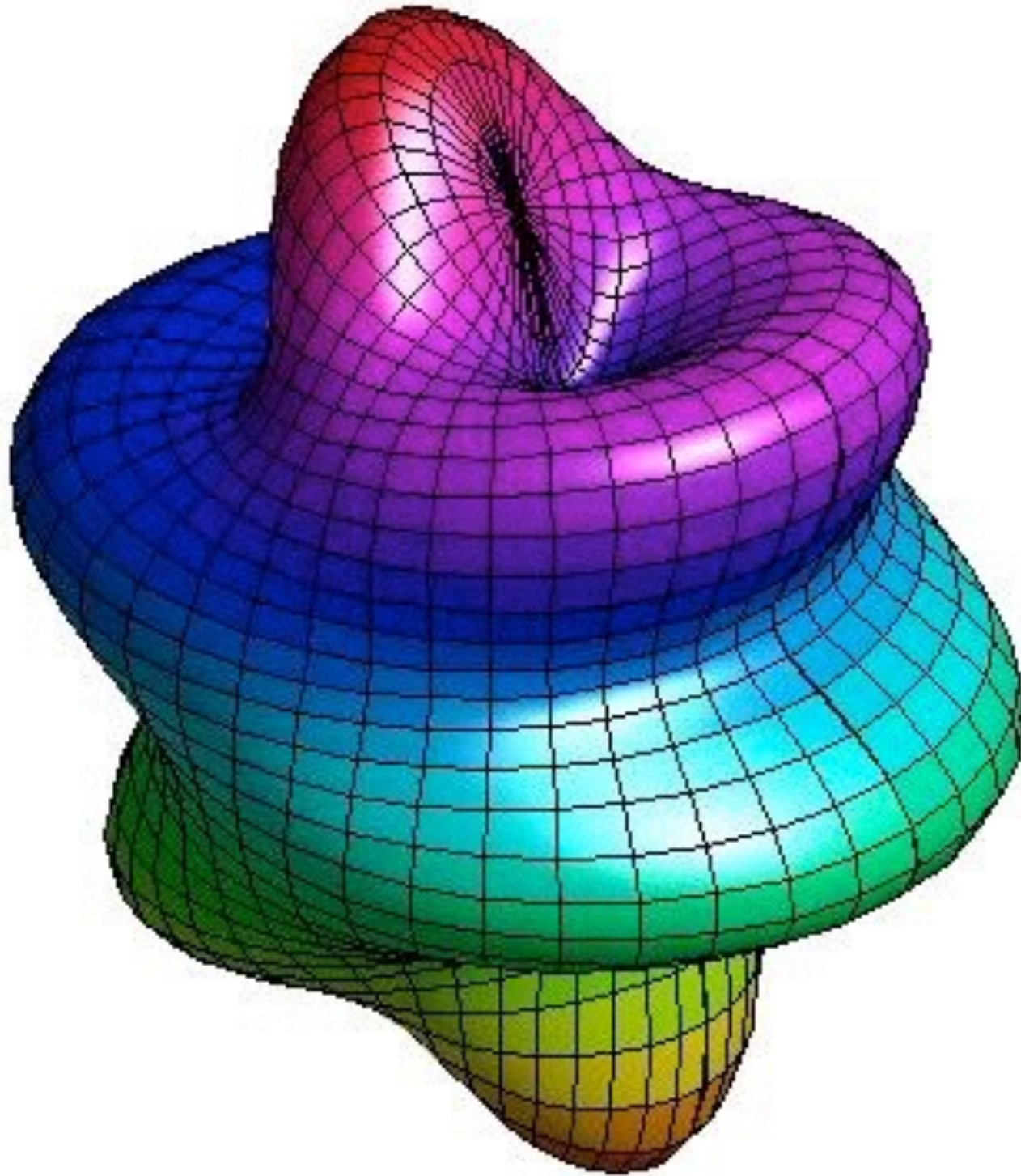


**Left:** Aliasing occurs due to finite pixel resolution.

**Right:** We deal with it by using an anti-aliasing filter



# Aliasing





# **Demo:**

# **Exporting Figures**

# Figure handling

Figures in Matlab are referenced using “handles”, which are pointers to different parts of the figure.

Example:

```
myhandle = plot(x,y);
```

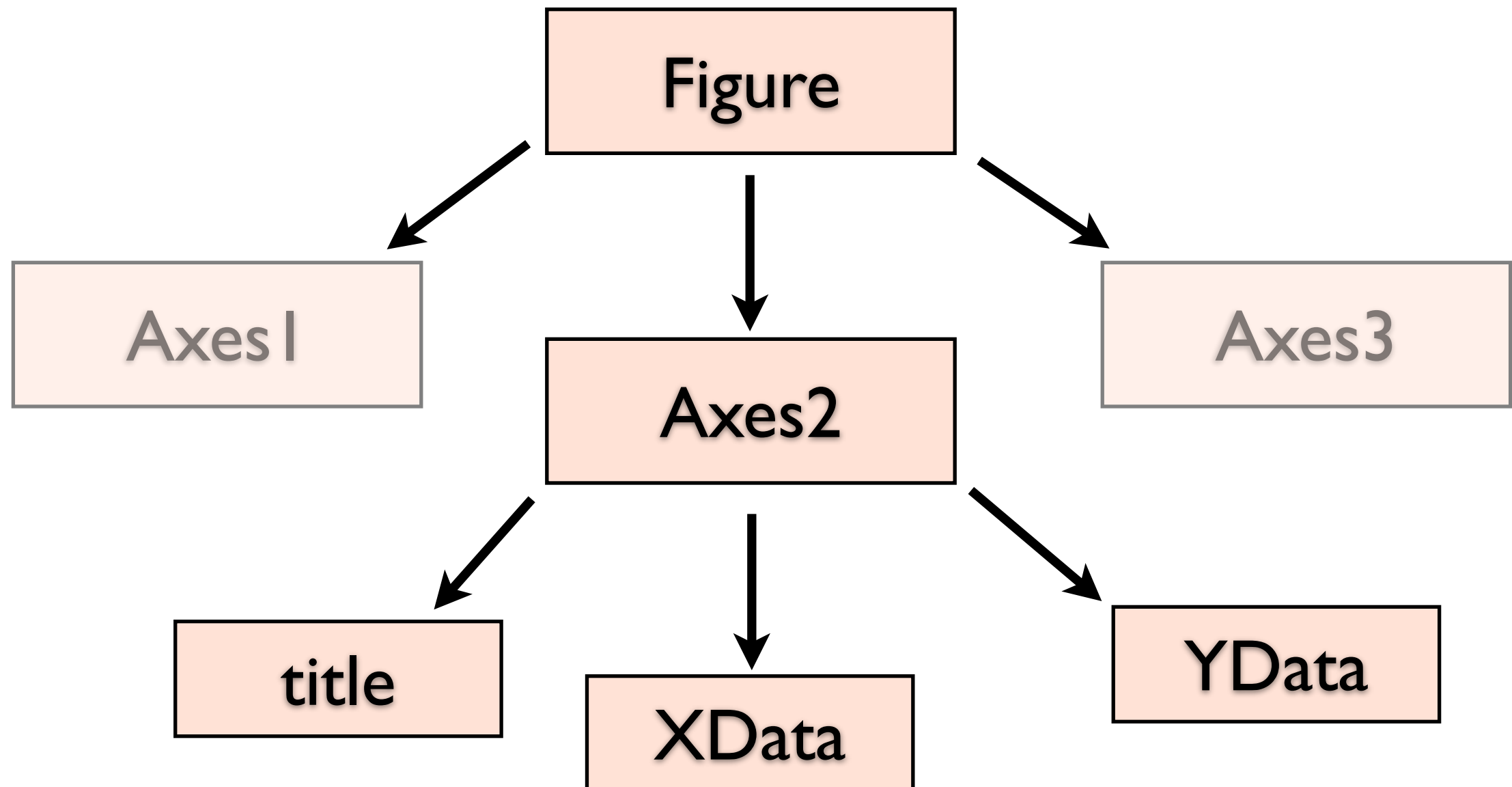
Will return a handle to the plot. Then you can run the following:

```
get(myhandle); % to see a list of properties
```

```
set(myhandle, 'Name', Value); % to set the value of a property
```

# Figure handling

Different parts of the figure are organized hierarchically:



# **Demo: Customizing Programmatically**

# **Demo:**

# **Annotating plots**

# Other Resources

2D and 3D visualization examples:

<http://www.mathworks.com/help/matlab/2-and-3d-plots.html>

Custom colormaps:

<http://colorbrewer.org>

Anti-aliasing filter:

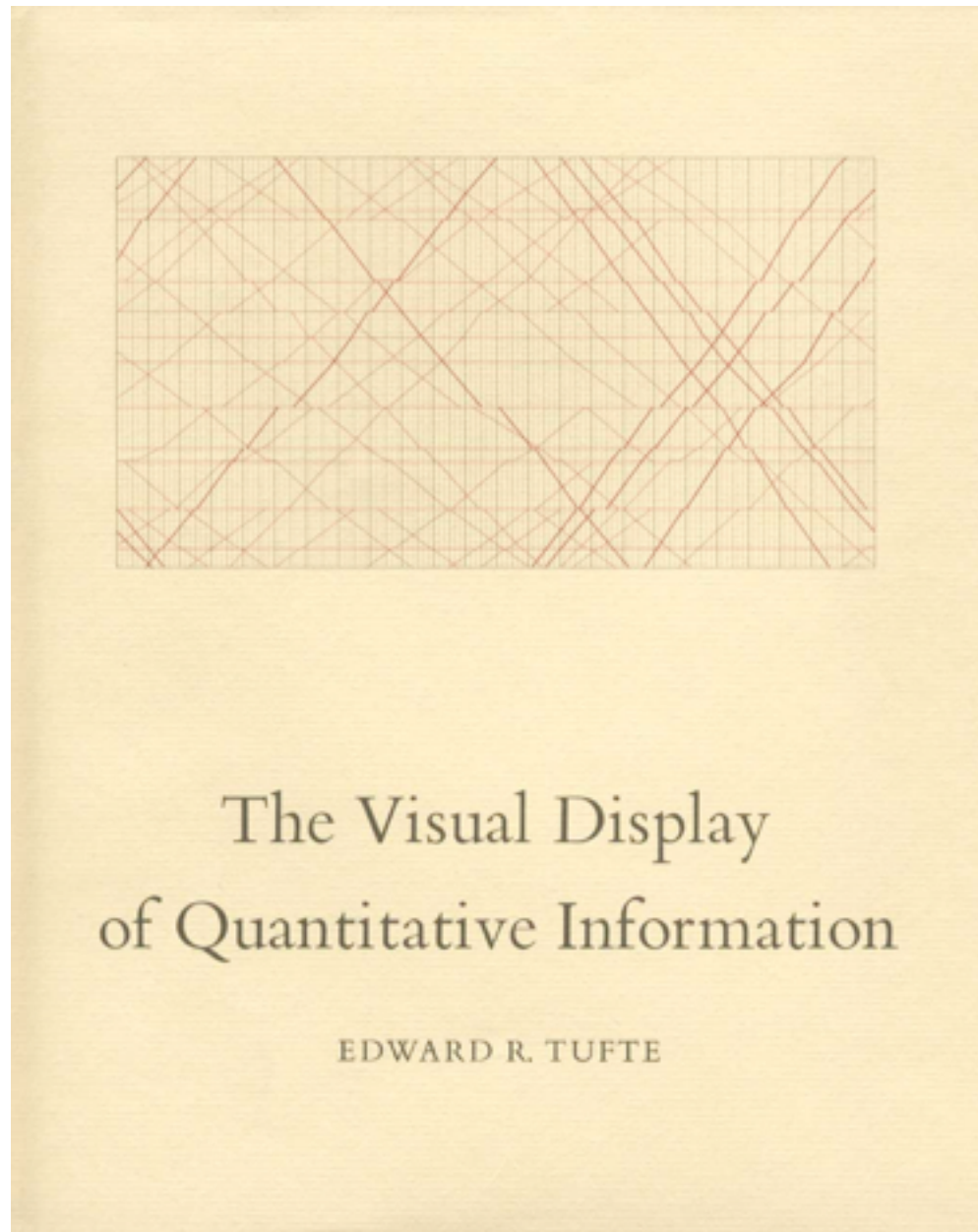
<http://www.mathworks.com/matlabcentral/fileexchange/20979-myaa-my-anti-alias-for-matlab>

Colors in figures (blog post)

<http://figuredesign.blogspot.com/2012/04/meeting-recap-colors-in-figures.html>



# Other Resources



<- Book by Edward Tufte

Lots of principles / guidelines  
for making good figures