Kodak moments

3-D Visualization and Color

3-D line and scatter plots

Create plots of 3-D curves and points using

- `plot3(x, y, z, ...)`
- `scatter3(x, y, z, ...)`

```
>> angles = linspace(0, 6*pi, 100);
>> plot3(cos(angles), sin(angles), angles, 'g-*', 'Linewidth', 2);
>> scatter3(rand(1,100), rand(1,100), rand(1,100), 40, 'm', 'filled');
```
Graph a 3-D function of two variables

Suppose we want to graph the function
\[ z = x^2 + y^2 \quad -4 \leq x, y \leq +4 \]

We can get there in three steps:

```matlab
>> [x  y] = meshgrid(-4:0.2:4);
>> z = x.^2 + y.^2;
>> mesh(x, y, z);
```

Use meshgrid to create 2-D coordinates

`meshgrid` creates 2-D matrices of coordinates, given an input vector that specifies the values for each coordinate

```matlab
>> [x  y] = meshgrid(-4:2:4);
>> z = x.^2 + y.^2;
>> mesh(x, y, z);
```

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>-4</td>
<td>32</td>
</tr>
<tr>
<td>-4</td>
<td>-2</td>
<td>20</td>
</tr>
<tr>
<td>-4</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>-4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>-4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2</td>
<td>-4</td>
<td>32</td>
</tr>
<tr>
<td>-2</td>
<td>-2</td>
<td>20</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>-4</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>-2</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>-4</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>-2</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>
So, what can I do with my plot?

Let’s begin with something a little more interesting:

```matlab
>> [x y z] = peaks(25);
>> mesh(x, y, z, 'EdgeColor', 'k');
>> axis tight
```

Display a surface,
and add a colorbar:

```matlab
>> surf(x, y, z);
>> colorbar
```

Colormaps

```matlab
>> cmap = colormap
```
```
0 0 0.5625
0 0 0.6250
0 0 0.6875
0 0 0.7500
0 0 0.8125
0 0 0.8750
0 0 0.9375
0 0 1.0000
0 0.0625 1.0000
0 0.1250 1.0000
0 0.1875 1.0000
...
...
...
0.6250 0 0
0.5625 0 0
0.5000 0 0
```

steadily increase blue
add in the green
pure red at the end
Named colormaps

MATLAB has named colormaps that can be set with `colormap`

```
colormap(hot)  colormap(spring)
```

View the colormaps by searching for `colormap` in Help facility

Don’t like the colors? Make your own!

Exercise: Create a matrix named `purples` that stores RGB values for 10 shades of purple, to use as a colormap

```
>> purples = zeros(?, ?);

???
```

```
>> colormap(purples);
```
Change your point of view

Change the point from which your 3-D surface is viewed

\[
\text{view(azimuth, elevation)}
\]

**default:** \text{view(-37.5, 30)}

\gg \text{view(-75, 45)}

\gg \text{rotate3d on} \quad \% \text{then adjust view with mouse}

Add shading...

Create a shaded surface using built-in shading methods

\gg [x y z] = sphere(20);
\gg surf(x, y, z);
\gg colormap(gray)

\gg \text{shading faceted} \quad \gg \text{shading flat} \quad \gg \text{shading interp}
Add a light source...

\[
\begin{align*}
&\texttt{[x y z] = peaks(25);} \\
&\texttt{surf(x, y, z);} \\
&\texttt{axis tight} \\
&\texttt{colormap(gray)} \\
&\texttt{shading interp} \\
&\texttt{light('Position', [0 -1 0])} \\
&\texttt{lighting phong}
\end{align*}
\]

... and adjust the surface material

Start the same as previous slide, then...

\[
\begin{align*}
&\texttt{colormap(copper)} \\
&\texttt{material dull} & \texttt{material shiny}
\end{align*}
\]
Contour plots

```matlab
>> [x y z] = peaks(25);
>> contour(x, y, z, 20, 'Linewidth', 2)
```

![Contour plot example]

Color images

```matlab
>> [x y z] = peaks(25);
>> imagesc(z)
>> colormap(gray)
>> colormap(hot)
```

![Color images example]