MATLAB Programming

Visualization

Copyright © Software Carpentry 2011
This work is licensed under the Creative Commons Attribution License
See http://software-carpentry.org/license.html for more information.
Good science requires good visualizations.
Why use MATLAB for plots?

- Produces publication quality plots and images
- Coupled with computation program.

Proper visualization might take exploration.
Simplest plot is a line plot:

$$\text{>> plot}(M)$$

Plot each column of $M$ as a separate line.
Sample data sets:

dow.txt : Daily closing value of Dow Jones Industrial Average, an index of the price of 30 stocks on the New York Stock Exchange

sp.txt : Daily closing value of the Standard and Poors 500, a broader index containing 500 stocks.
Data looks like:

<table>
<thead>
<tr>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1950</td>
<td>01</td>
<td>03</td>
<td>198.89</td>
</tr>
<tr>
<td>1950</td>
<td>01</td>
<td>04</td>
<td>200.20</td>
</tr>
<tr>
<td>1950</td>
<td>01</td>
<td>05</td>
<td>200.57</td>
</tr>
<tr>
<td>1950</td>
<td>01</td>
<td>06</td>
<td>200.96</td>
</tr>
<tr>
<td>1950</td>
<td>01</td>
<td>07</td>
<td>201.90</td>
</tr>
</tbody>
</table>
Load data:

```matlab
>> dow = importdata('dow.txt');
>> sp = importdata('sp.txt');
```

Simple plot:

```matlab
>> plot(dow(:,4))
```
>> plot(dow(:,4))
>> plot(dow(:,4))

Was this 1987?
>> plot(dow(:,4))

What does the X axis mean?
>> dow(1,1:3)
ans =
    1950  1   3
   Year  Month  Day
>> time = dow(1,:) + (dow(2,:)-1) / 12 +
    (dow(3,:)-1) / 30 / 12

Plot the Dow’s value versus time:

>> plot(time, dow( :,4 ))
>> plot(time, dow(:,4))

Figure 1
What if we want to edit the plot?
What if we want to edit the plot?
What if we want to edit the plot?

Provides access to plot details
What if we want to edit the plot?

- Change line
- Add markers
- Many other options.
Compare the Dow to the S&P:

```matlab
>> stocks = [dow(:,4) sp(:,4)];
>> plot(time, stocks);
```

Plotting a matrix (stocks) against a vector (time) plots each column of the matrix with the shared X-axis.
>> plot(time, stocks);
Rescale the indices to start at the same place:

```matlab
>> d2 = dow(:,4) / dow(1:4);
>> s2 = sp(:,4) / sp(1,4);
>> plot(time, [d2 s2]);
```
>> plot(time, [d2 s2]);
MATLAB has utilities to plot many kinds of data:

- **hist**: histograms
- **pie**: pie charts
- **bar, barh**: bar charts

Even many kinds of 3D charts:

- **pie3**
- **bar3**
- **pareto**
A lot of data is one dimensional. What about 2-D data?

Example: geographically oriented data.

Ever wondered where people tweet the most?
Question: what places in Toronto are the most popular locations for people to send a geolocated tweet?

Data collection:

Record all geolocated tweets for 2 months.

Divide the city into a grid and count the number of tweets in each cell of the grid.
Question: what places in Toronto are the most popular locations for people to send a geo-located tweet?

Data collection:
- Record all geolocated tweets for 2 months.
- Divide the city into a grid and count the number of tweets in each cell of the grid.

Data: a matrix of grid centers and the relative number of tweets in that spot.
Question: what places in Toronto are the most popular locations for people to send a geo-located tweet?

Data collection:

Record all geolocated tweets for 2 months.
Divide the city into a grid and count the number of tweets in each cell of the grid.

Data: a matrix of grid centers and the relative number of tweets in that spot.
>> image(data);
>> image(data);
image():

Take either an N X M or N X M X 3 array.

Third dimension is for three channels of a color image.

Map each location a color using a colormap.
Only use first column because Data is 4x4
A colormap is a color guide that maps the values 0.0 to 64.0 to colors.

Many colormapsé

Just checké

>> help colormaps

é for all the options.
A colormap is a color guide that maps the values 0.0 to 64.0 to colors.

Many colormapsé

Just checké

>> help colormaps

é for all the options.

What if our matrix has a different range?
>> imagesc(data);

Scales the matrix to use the entire colormap.
Key: imagesc scales the data **linearly**.

Our data:

```matlab
>> max(max(data))
1.93 e+04
>> min(min(data))
2.05 e-24
>> mean(mean(data))
1.1190
```

Our data is scaled exponentially
>> imagesc(log(data));
`>> imagesc(log(data)), colormap gray;`
>> imagesc(log(data)), colormap hot;
Conclusion:

Imaging is a powerful way to explore data, but be sure to take full advantage of the pattern in that data.