Introduction to Matlab

Sasha Lukyanov, 2018 Xenopus Bioinformatics Workshop, MBL, Woods Hole
MATLAB Environment
What do we use?

```
MATLAB Version: 9.3.0.713579 (R2017b)
MATLAB License Number: 596681
Operating System: Mac OS X Version: 10.12.6 Build: 16G1036
Java Version: Java 1.8.0_121-b13 with Oracle Corporation Java HotSpot(TM) 64-Bit Server VM mixed mode

<table>
<thead>
<tr>
<th>Toolbox</th>
<th>Version</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATLAB</td>
<td>9.3</td>
<td>R2017b</td>
</tr>
<tr>
<td>Bioinformatics Toolbox</td>
<td>4.9</td>
<td>R2017b</td>
</tr>
<tr>
<td>Curve Fitting Toolbox</td>
<td>3.5.6</td>
<td>R2017b</td>
</tr>
<tr>
<td>Global Optimization Toolbox</td>
<td>3.4.3</td>
<td>R2017b</td>
</tr>
<tr>
<td>Image Processing Toolbox</td>
<td>10.1</td>
<td>R2017b</td>
</tr>
<tr>
<td>Neural Network Toolbox</td>
<td>11.0</td>
<td>R2017b</td>
</tr>
<tr>
<td>Optimization Toolbox</td>
<td>8.0</td>
<td>R2017b</td>
</tr>
<tr>
<td>Parallel Computing Toolbox</td>
<td>6.11</td>
<td>R2017b</td>
</tr>
<tr>
<td>Partial Differential Equation Toolbox</td>
<td>2.5</td>
<td>R2017b</td>
</tr>
<tr>
<td>Signal Processing Toolbox</td>
<td>7.5</td>
<td>R2017b</td>
</tr>
<tr>
<td>Statistics and Machine Learning Toolbox</td>
<td>11.2</td>
<td>R2017b</td>
</tr>
<tr>
<td>Symbolic Math Toolbox</td>
<td>8.0</td>
<td>R2017b</td>
</tr>
</tbody>
</table>
```
If you know the name of the command, type `help <command>` at the MATLAB command prompt for a quick refresher. To get more detailed information, including related commands, type `doc <command>` to open up the help browser.

If you don't know the name of the command, you can type `lookfor <keywords>` to get a list of commands that have `<keywords>` in their description.

Finally like most other programs, there is a help menu in the command bar at the top of the screen. Clicking “Product Help” will open the MATLAB documentation browser. The same materials are also online at [http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.html](http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.html).
MATLAB: Command Line

To get started, select MATLAB Help or Demos from the Help menu.

```
ans = 5
>> 2+3
ans =
5
>>
```
Basic Commands

- Some basic syntax
  - Defining a variable: \( x = 2 \)
  - Defining a vector \([1 \ 3 \ 5]\): \( y = [1 \ 3 \ 5] \)
  - Defining a matrix \( \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \): \( z = [1 \ 2; \ 3 \ 4] \)
  - Suppressing output: ;
  - Accessing an element in a vector: \( y(2) \) (equals 3)
  - Accessing an element in a matrix: \( z(2,1) \) (equals 3)
  - Accessing a row/column in a matrix: \( z(2,:) \) (equals [3 4])
  - Accessing elements in a vector that satisfy some condition \( y(y>2) \) (equals [3 5])
  - Accessing indices of elements in a vector that satisfy some condition \( \text{find}(y>2) \) (equals [2 3])
Basic Commands

- Transposing a vector: \[ y' \] (equals \[
\begin{bmatrix}
1 \\
3 \\
\end{bmatrix}
\])

- Mathematical operators: \(+, -, *, /, ^\)
- Element-wise operations: \(.+, .-, .*, ./, .^\)
  Note the dot!
  \[
  [1 \ 2 \ 3].*y = [1 \ 6 \ 15]
  \]

- Special numbers (\(\pi, \infty\)): \(\text{pi, Inf, i}\)
- Exponential (\(e^x\)): \(\text{exp(x)}\)
- Natural logarithm (\(\ln x\)): \(\text{log(x)}\)
- Log base 10, base 2: \(\text{log10(x)}, \text{log2(x)}\)
- Trigonometric functions: \(\cos(x), \sin(x), \tan(x)\)
- Absolute value (\(|x|\)): \(\text{abs(x)}\)
Basic Commands

- Some basic commands
  - Ways to make vectors:
    - Equally spaced intervals: \( x = 1:0.5:3 \)
      (equals \([1 \ 1.5 \ 2 \ 2.5 \ 3]\))
    - All zeros: \( x = \text{zeros}(1, n) \)
      (equals \(n\) zeros)
    - All ones: \( x = \text{ones}(1, n) \)
      (equals \(n\) ones)
  - Draw \(n\) uniformly distributed random numbers between 0 and 1
    \( \text{rand}(1, n) \)
  - Minimum of vector \( \text{min}(y) \)
  - Maximum of vector \( \text{max}(y) \)
  - Sum of vector \( \text{sum}(y) \)
  - Mean of vector \( \text{mean}(y) \)
  - Variance of vector \( \text{var}(y) \)
  - Standard deviation of vector \( \text{std}(y) \)
  - Size of vector \( \text{size}(y) \)
  - Length of vector \( \text{length}(y) \)
Plotting

$>> \text{plot}(t, g);$  

Here we are plotting the values of $g$ (y-axis) versus the values of $t$ (x-axis).
MATLAB: Tables

>> Table = readtable('germPlasm2try.152642725094/expr.csv', 'ReadVariableNames', false);

readtable determines the file format from the file extension:
  • .txt, .dat, or .csv for delimited text files
  • .xls, .xlsx, .xltm, .xltx, or .ods for spreadsheet files
MATLAB: Tables

TableRes = importdata('germPlasm2try.152642725094/expr.csv');

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>26550x5 double</td>
</tr>
<tr>
<td>textdata</td>
<td>26550x1 cell</td>
</tr>
<tr>
<td>rowheaders</td>
<td>26550x1 cell</td>
</tr>
</tbody>
</table>

See Also
- imread
- load
- readtable
- save
- textscan
- uimport
- xlsread

Topics
- Supported File Formats for Import and Export
- Ways to Import Text Files
- Ways to Import Spreadsheets
- Import or Export a Sequence of Files
COORD = importdata('germPlasm2try.152642725094/coordinates.csv'); <- Read Table

figure; <- Open figure window
colormap(jet); <- Define your color map
scatter(COORD(:,2),COORD(:,3),[], abs(COORD(:,1)), ...
        'filled', 'MarkerFaceColor','flat', ...
        'MarkerEdgeColor','none'); <- Scatter plot

... <- Color of points depending on difference
MATLAB: Tables - Examples

```matlab
Table  = readtable('germPlasm2try.152642725094/expr.csv', 'ReadVariableNames', false);

Names   = table2array(Table(1:end, 'Var1'));
Cell1   = table2array(Table(1:end, 'Var2'));
Cell2   = table2array(Table(1:end, 'Var3'));
Cell3   = table2array(Table(1:end, 'Var4'));
Cell4   = table2array(Table(1:end, 'Var5'));
Cell5   = table2array(Table(1:end, 'Var6'));

% % Figures
% figure;
colormap(jet);
scatter(log10(Cell1),log10(Cell2),50, ... 
    abs(log10(Cell1)-log10(Cell2)), ... 
    'filled', 'MarkerFaceColor','flat', ...) 
    'MarkerEdgeColor','none');
```

<- Read Table
<- Read different fields in the Table
<- Scatter plot of Cell1 vs. Cell2
[coeff,score,latent,tsquared,explained,mu] = pca(X); <- PCA Compute

figure;
vbls = {'V1','V2','V3','V4','V5'};
biplot(coeff(:,1:2),'scores',score(:,1:2),'varlabels',vbls); <- Plot PCA
MATLAB: Bioinformatics ToolBox

**High-Throughput Sequencing**
Gene expression, transcription factor, and methylation analysis of Next-Generation Sequencing (NGS) data, including RNA-Seq and ChIP-Seq

**Microarray Analysis**
Gene expression and genetic variant analysis of microarray data

**Sequence Analysis**
Genomic and proteomic sequences, alignment, and phylogenetics

**Structural Analysis**
Visualize and manipulate 3-D structures of proteins and other biomolecules; RNA secondary structure prediction and visualization

**Mass Spectrometry and Bioanalytics**
Data from separation techniques that produce traces with peaks, including MS, LC/MS, NMR, chromatography, and electrophoresis
Perform search on local BLAST database to create BLAST report

\[
\text{blastlocal('InputQuery', InputQueryValue)}
\]
\[
\text{Data} = \text{blastlocal('InputQuery', InputQueryValue)}
\]

Object from short-read sequence data

\[
\text{BioMapobj} = \text{BioMap(File)}
\]

Reading the FASTA data

\%
\%
\% Read FASTA file
\%

[FASTAData, FASTASeq] = fastaread('germPlasm2try.152642725094/unknown_cDNA.fa');
Identifying Differentially Expressed Genes from RNA-Seq Data

Statistics and Machine Learning Toolbox
Generalized linear model regression