Useful Numerical Tools for Final Projects

Plotting functions in Mathematica

`Plot[f, {x, xmin, xmax}]` generates a plot of \( f \) as a function of \( x \) from \( x_{\text{min}} \) to \( x_{\text{max}} \).  

`Plot[f_1, f_2, ... , {x, xmin, xmax}]` plots several functions \( f_i \).

Here is an example of plotting the function \( y = \frac{x^3}{e^x - 1} \) on the interval \( x \in [0,10] \).

```
In[1]:= Plot[x^3/(Exp[x]-1),{x,0,10}]
```

![Plot of \( x^3/(e^x-1) \) from 0 to 10]

Below is an example of plotting two functions on the same graph. Note that the option to label the axes has been added at the end of the command. The `Plot` function has many options that you can set. You can find more information at [http://reference.wolfram.com/mathematica/tutorial/Options.html](http://reference.wolfram.com/mathematica/tutorial/Options.html). Another option is to copy your graphs and edit them (e.g. add labels) in PowerPoint. You can then save it as a jpeg and include it into your paper.

```
In[2]:= Plot[{x^3/(Exp[x]-1),1},{x,0,10},AxesLabel->\{x,y\}]
```

![Plot of two functions with axes labeled]

**Exercise:**

Plot the function \( q_g (g) = 1.07 \frac{0.018 + g}{0.06 + g} \) on the interval \( 0 < g < 1 \) (the Michaelis-Menten function in the Grassland project).
Finding Exact Solutions to Differential Equations in Mathematica

DSolve[eqn, y, x] solves a differential equation for the function y, with independent variable x, and
DSolve[{eqn1, eqn2, ...}, {y1, y2, ...}, x] solves a list of differential equations.

For example, the following command line solves the differential equation \( y'(x) = y \).

\[
\text{sol} = \text{DSolve}[y'[x] == y[x], y[x], x]
\]

You can also specify an initial condition.

\[
\text{sol} = \text{DSolve}\{y'[x] == y[x], y[0] == 1\}, y[x], x\}
\]

In order to plot the solution, you need to tell Mathematica where to look for the solution—this is what \( \text{/. sol} \) does.

\[
\text{Plot}[y[x]/.\text{sol}, \{x, -3, 3\}]
\]

**Exercise**: Use DSolve to solve the following linear system (from the project on Radioactive Tracers):

\[
\frac{dx_1}{dt} = -1.25x_1 + 0.69x_2
\]
\[
\frac{dx_2}{dt} = 1.06x_1 - 0.72x_2
\]
\[
\frac{dx_3}{dt} = 0.19x_1 + 0.03x_2 - 2.50x_3
\]
\[
\frac{dx_4}{dt} = 2.50x_3
\]

with initial condition \( x_1 = 1, x_2 = x_3 = x_4 = 0 \). Plot the solution on the interval \( 0 < t < 40 \).

Finding Numerical Solutions to Differential Equations in Mathematica

If you give Mathematica an equation that doesn’t have an analytical solution, it will sit there for a long time, if not forever. If you have a really messy differential equation that you want to solve numerically, try NDSolve instead. The command line is the same as DSolve. Below is an example.

We would like to solve the following nonlinear system of ODEs:
\[
\begin{align*}
\frac{dx}{dt} &= -y - x^2, \quad x(0) = 1 \\
\frac{dy}{dt} &= 2x - y^3, \quad y(0) = 1
\end{align*}
\]

\[
\text{s = NDSolve}\left\{\{x'[t] == -y[t] - x[t]^2, y'[t] == 2x[t] - y[t]^3, x[0] == y[0] == 1\}, \{x, y\}, \{t, 20\}\right\}
\]

\[
\{x \to \text{InterpolatingFunction}[\{\{0., 20.\}\}, \langle\rangle],
  \\
y \to \text{InterpolatingFunction}[\{\{0., 20.\}\}, \langle\rangle]\}
\]

When you run \text{NDSolve}, it outputs the solutions stored in a function called \text{InterpolatingFunction} in a list. In order to plot those two \text{InterpolatingFunctions}, you need to extract them from the list. That’s what \text{Evaluate} and \text{/}.\text{solution} does. \text{Evaluate} tells Mathematica what the name of the functions are to extract from the list (\text{x} and \text{y}) and \text{/}.\text{solution} tells it where to look for them.

\[
\text{Plot[Evaluate[x[t]/.s],\{t,0,10\}]}
\]

You can plot the solution curve (\text{x}(t), \text{y}(t)) in the phase plane using the command \text{ParametricPlot}.

\[
\text{ParametricPlot[Evaluate[\{x[t], y[t]\}/.s],\{t,0,10\}]}
\]

Using \text{PPLANE} to study differential equations:

\text{PPLANE} is a free tool that you can run from your web browser. Go to \url{http://math.rice.edu/~dfield/dpp.html} and click on \text{PPLANE 2005.10}. Given a system of two ordinary differential equations, the program can draw the direction field, phase portrait, and other tools similar to \text{DETools}. However, \text{PPLANE} has the ability to draw nullclines. We will explore this tool in class.