USING MATHEMATICA TO CONVEY MATHEMATICAL CONCEPTS: AN INTRODUCTORY MODELING COURSE FOR NON-SCIENCE MAJORS

Silvia Heubach
Department of Mathematics and Computer Science
California State University Los Angeles
5151 State University Drive
Los Angeles, CA 90032-8204
sheubac@calstatela.edu

1 Introduction

This workshop provides a hands-on introduction to a new modeling course designed with a Course and Curriculum Development grant from the National Science Foundation[1]. The unique feature of this course is its intended audience: Non-science majors, especially students who take only one mathematics course during their undergraduate program. For most of these students mathematics applications consist of balancing checkbooks and computing percentages. In the new modeling course, students gain an understanding of the use of mathematical models across many disciplines and learn how models can be used to predict the future or to decide on policy.

The course focuses on model development, test of model against data and interpretation of results. Both discrete and continuous models are used for deriving quantitative and qualitative interpretations of long-term behavior and sensitivity to model parameters. The goal is to have students experience the modeling process and to open their eyes to the fact that their everyday lives are influenced by decisions based on mathematical models.

In the past, a modeling course would not have been possible at a freshman/sophomore level, as Calculus and Differential Equations are common pre-requisites for modeling courses. However, computer algebra systems such as Mathematica or Maple can be utilized to perform some of the advanced computations to allow students without the advanced mathematics background to explore and analyze mathematical models. The presenter has developed special packages to cut down on the learning curve for Mathematica, taking advantage of the improved user interface in versions 3.0 (and 4.0) of the software. This customized interface frees the student from the usual beginner's frustrations of learning Mathematica, and assists in interactively exploring "What if?" scenarios.

Naturally, as the intended student audience for the modeling course has only a limited mathematical background, the models developed in this course are very basic, and many theoretical results are derived heuristically, using a combination of numerical and graphical approach. However, unlike in the more advanced theoretical courses, comparison of the

[1] This work has been supported in part by NSF grant DUE-9653262.

model predictions to real data is an integral part of the course. This connection between the theoretical models and their application engages the students, making the course content relevant to their lives, and increases their appreciation of mathematics. With an understanding of the modeling process, in particular the fact that models need to be validated by data and that their predictions may be limited, students will be able to critically evaluate conflicting information on topics such as Global Warming and to put model predictions into the proper context.

2 History of Course Development

Different teaching experiences over the years have lead to the development of this course. It started when I was teaching a modeling course at Humboldt State University which did not require the differential equations course as a prerequisite. In order to bring the differential equations to life, and also to show how models with linear and non-linear differential equations differ, I took the students to the computer lab and used MacMath (developed by Hubbard and West) to show vector fields for differential equations in one and two variables. It was amazing how this experience changed the students' perception. The most dramatic difference took place for students that did not have a very strong mathematical background - they were asking when we would go to the lab again!

With this experience in mind, and my own steadily increasing use and knowledge of Mathematica, I developed a Mathematica component for the differential equations course at California State University, Los Angeles (CSLA). The main drawback was that it took students quite a bit of time to get going with Mathematica. In the context of the quarter system at CSLA, incorporation of Mathematica seemed to hinder as much as it helped.

At the same time I was teaching one section of our General Educations mathematics course each quarter, and was thinking about the content of that course. It seemed that a more beneficial course for this particular student audience could be designed, oriented toward modeling, in the spirit of the Calculus and Pre-Calculus reform movement. With the advent of Mathematica version 3.0, and in particular the feature of user-defined palettes, the idea for this course was born. Using palettes would avoid the usual beginner's frustrations (misspelling of function names, capitalization) and would allow me to create functions specifically designed for this course. (I also explored the use of Maple, but at the time I designed the course, Maple did not have an analogous feature.)

3 Incorporation of Technology

Use of technology, whether computers or graphing calculators, introduces an additional level of difficulty for students in any course. However, the palette feature of Mathematica 3.0 (and 4.0) provides an intuitive point-and-click interface to access the specially designed functions. For each major section of the course, a package with functions relating to the particular topic has been developed. These functions are pasted into the notebook by clicking on a button in the corresponding palette, thus avoiding frustrations from typos

or incorrect use of parentheses and brackets. The palettes can be accessed from within any notebook via the File menu. In tandem with each palette, a help file explaining the different palette functions is available on-line. Each help file is based on (a) specific example(s) and has hyperlink buttons for each palette function described within. Clicking on a button links the student to a full description on how to use that particular function, and shows an example of how to use the function.

A major concern in any course that relies to a great amount on technology is the creation of a black-box approach, where the computer is the source of all answers blindly accepted by the students. To avoid this problem, students are performing computations initially using paper and pencil (whenever suitable) to strengthen their conceptual understanding before using the corresponding palette functions. In addition, students are required to make predictions about the model results before solving the model using Mathematica. Model results are discussed and interpreted in the context of the problem both orally and in writing.

Student assignments consist of exercises reinforcing new definitions or methods, as well as open-ended questions that require both computations and writing. These exercises extend models presented in class by either asking additional questions based on the model or by introducing variations to the model. A final class project guides students through the complete modeling process: Formulating a model, testing it against data, discussing possible modifications and justifying their final choice for the model. These projects are presented to the class either orally or as a poster session.

4 Course Content

The course starts by describing the sequence of models for the solar system proposed over the centuries. This illustrates the iterative nature of the modeling process, specifically the fact that new data may indicate that the model needs to be revised. The remainder of the course is divided into three principal parts: Part I defines the concept of a function and leads to a model derivation by means of a least squares fit to a given set of data. Part II derives discrete models (both single and multiple equations) from assumptions, while part III focuses on continuous models.

4.1 Functions and Least Squares Fit

Using the input-output paradigm, the concept of a function is defined, and the various representations (list of values, graph, or formula) for functions are presented. Polynomials (of degree ≤ 3), exponential, logistic and sine functions are introduced. Students explore characteristics such as constant first or second unit differences as well as features of the shape of the graph for the different types of functions. These properties are then used to decide which type of function should be fitted to given real-world data using the method of least squares fit. The ideas behind this method, namely measuring squared error between a potential function and the data, and then trying to find the function that has the smallest such error, are illustrated with an example.

The associated package is called *DataFit* and contains special functions to plot the data, compute first and second unit differences as well as unit ratios to test for either linear, quadratic or exponential functions. In addition, there are package functions to produce a least squares fit for all the function types discussed. The result of such a fit is a graph of the data together with the fitted function, as well as the functional expression of the least squares fit function.

4.2 Discrete Models

Discrete models, for both single equations and systems of equations, are derived from assumptions using the paradigm

new = old + change.

The first models are based on known dynamics for the change. From the iterative model equation, values for the output variable are computed recursively, then an explicit solution is derived and verified. Specific question that cannot be answered precisely using the iterative model equation are used to motivate the search for an explicit solution. In addition, the notion of equilibrium is introduced via examples. A brief excursion into classification of discrete dynamical systems (linearity and order) follows, then theorems for the explicit solution and equilibrium values (of a linear DDS) are stated. For non-linear models, Cobb-web diagrams are introduced, and stability of equilibria and the long-term behavior of the model are discussed using tables of model values and their graphs.

Next, the students progress to a model where assumptions about the dynamics of the change need to be made. A basic exponential population model and a logistic model for the spread of a disease are developed. The chapter ends with a model from genetics, which uses a world-wide web site to simulate genetic drift (the loss of one type of gene). The package for this part of the course is called *DDS1* (discrete dynamical system of one variable) and contains functions to compute and graph model values, to create a Cobb-web animation and to solve relevant equations.

The next chapter introduces discrete dynamical systems of several variables. Starting from the set of equations, matrix notation is introduced as a way to capture the essence of the system. The *DDSn* package (for discrete dynamical systems of *n* equations) includes functions such as LiveVecs, which creates an animation of pairs of input/output vectors (including the Eigenvectors) of a 2-by-2 matrix. This animation allows students to discover the special vectors for which the output vector is a multiple of the input vector.Through numerical examples, students are guided towards the realization that the explicit solution can be expressed in terms of the Eigenvectors and Eigenvalues. This form of the general solution is then produced by palette functions to asses the long-term behavior of the system.

Examples discussed in this section include population models which incorporate age structure, simple probabilistic models and predator-prey models (non-linear). For this particular type of non-linear system of two variables, special package functions display the

system dynamics in various regions of the plane. Furthermore, the sequence of model values can be displayed for several starting values at once.

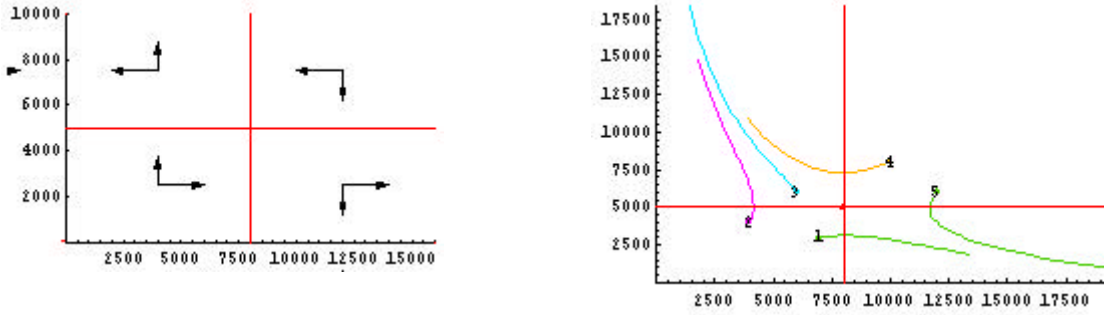


Figure 1: a) System Dynamics; b) Long-term behavior

4.3 Continuous Models

Deriving model equations in the continuous setting involves the rate of change of a quantity. Using the example of average and instantaneous velocity, the interpretation of the derivative as rate of change is introduced. Side by side with the numerical derivation, the graphical interpretation of the derivative as the limit of secant slopes is developed and visualized through an animation. The two notions of the derivative are combined for the qualitative analysis of a differential equation. Using the slope field, students graphically derive the model solution and discuss the sensitivity of the long-term behavior to initial conditions. Examples used are exponential decay and the continuous version of systems like the predator-prey model. All models are restricted to one or two variables, as these are the only ones that can be displayed visually using vector fields.

5 Student Responses

The course has been taught three times at CSLA (Spring 98, Fall 98, and Winter 99) Generally, student reaction was very positive. Students expressed the fact that they could "see" mathematics and that they had a new-found appreciation of its use. The only concern was the fact that students had do the course work in computer labs on campus, which is a limitation at a commuter campus like CSLA. However, with the increased power and pricing of a TI-89 (comparable to the price of the student version of Mathematica) a calculator based version of this course to remedy this problem is becoming feasible.

6 Course Material

The lecture notes, packages, palettes, and introductory Mathematica lessons are posted on the WWW (www.cinenet.net/~silvi); this site can also be accessed via link from the author's home page (www.calstatela.edu/faculty/sheubac/sheubac.htm). Please send any feedback on the lecture notes, packages, or the Mathematica lessons to the author at sheubac@calstatela.edu.