# INTRODUCTION TO SCIENTIFIC MODELING

**UNM CS 365**
Fall Term, 2014
Mon/Wed 1:00 - 2:50 pm
Centennial 1026

**Professor:** Stephanie Forrest

Office: Farris Engineering Center 355
Phone: 277-7104 (during office hours only)
Email: forrest@cs.unm.edu (anytime)
Office Hours: Mon. 3:00 - 5:00; Wed. 10:00 - 12:00
Web site: `http://cs.unm.edu/forrest/classes/cs365-2014/`

**Textbook:**

*Complexity: A Guided Tour* M. Mitchell, Oxford 2009
Selected Readings
*Getting Started With MatLab*
Recommended Secondary Sources
`http://www.complexityexplorer.org/online-courses/11`
*Networks* M. .E. J. Newman, Oxford 2010

**Course Description:**

CS 365 will focus on using the computer as a laboratory for understanding the world around us. We will study and develop computational models of complex systems, study different modeling techniques, learn how to design simulation experiments, and how to analyze results. Data analysis will include simple curve fitting, looking for power laws, testing for goodness of fit, and interpreting the results of experiments. We will explore these topics in the context of compelling interdisciplinary examples, including: predator-prey models from ecology, evolutionary computation, simple games such as the Prisoner's Dilemma, artificial life and machine learning.

**Course Assignments and Grading:**

The course will consist of lectures and class discussions based on assigned readings and online lectures. There will be a moderate amount of reading (plan on one or two readings per week), three or four programming projects, and one or two exams. Grades will be based on programming assignments and exercises ($\approx 40\%$), exams ($\approx 40\%$), and class participation ($\approx 20\%$).

The official programming language for the course will be MatLab, which provides built in plotting, curve fitting routines, and other useful packages. Some assignments will be most easily completed using MatLab, and access to introductory tutorials will be provided for those who are new to MatLab. However, you are free to use your favorite programming language if you choose. Some programming experience is expected as a pre-requisite, and you will be responsible for writing and debugging your own computer programs.

**Collaboration, online help and academic honesty**

Programming is often a collaborative endeavor, but in this class you are expected to develop code for assignments independently and hand in only your own work. Students are encouraged to help each other with concepts from class, but they are not allowed to copy any part of another students code. In other words, students may help each other by communicating in English, but not by communicating computer programs or snippets of computer programs. You are strongly encouraged to seek help from the instructor to clarify assignments, algorithm design and tips on writing up lab reports, although you are expected to debug your own code. You are responsible for documenting all code that you do not write yourself, for example, open source code that you use in one of your assignments. Any code, images, or text that is obtained online and modified, must be cited. The original code and web address must be provided in the program comments and lab report when it is turned in. If you end up collaborating with another student, the details of the collaboration must be documented in program comments and lab reports. Any student caught copying code from any source and presenting it as his or her own will be failed and reported to the University for cheating. Any student who is unclear whether something is cheating should ask the instructor.

## Course Topics

**Introduction** (1 week)
Topics:
- General course introduction, MatLab resources
- "What is a model?" What can models teach us about the natural and engineered world?
- How do we evaluate the quality of a model?
- Modeling approaches: Continuous and discrete methods

Readings:
- Mitchell preface, Ch 1, Ch 14 (p 209-212 only)
- C. Hall and J. Day "Revisiting the limits to growth"
- P.J. Denning "Computing is a natural science"
- MatLab Tutorial. Download software from **http://it.unm.edu/download**

**Modeling Case Studies** (2 weeks)
**Assignment 1: To be announced**
Topics:
- Malware on the Internet
- Networks in nature and technology
- Agent-based and dynamical models of Influenza
- Modeling ant foraging strategies

Readings:
- Mitchell, Ch 2
- Mitchell, Ch. 15 - 16
- Mitchell, Ch. 9

**Cellular automata and agent-based models** (2 weeks)
**Assignment 2: To be announced**
Topics:
- 1-dimensional cellular automata

The game of life
Extensions and applications

Readings:
Mitchell Ch. 10
Wolfram *A New Kind of Science* Ch. 8 (available from
http://www.wolframscience.com/nksonline/toc.html)
Axelrod: Agent-based modeling as a bridge between disciplines

**Simulation Underpinnings** (1 week)

Topics:
Random numbers
Discrete event vs. discrete time simulations
Validation issues
Reproducible results

Readings:
To be announced

**Data Analysis and Scaling Laws** (3 weeks)
**Assignment 3: To be announced**

Topics:
Statistical distributions and testing for statistical significance
Data collection and analysis
Graphing
Curve fitting
Maximum Likelihood Estimation
Power laws
Scaling laws in nature
Complex networks

Readings:
Review Mitchell Ch. 15-16, Mitchell 17
Newman Ch. 8
Sibly, Borwn, and Kodric-Brown *Metabolic Ecology*, Ch. 24 (Beyond Biology)

**Midterm**

**Predator/Prey Models and epidemics** (2 weeks)

Topics:
Brief introduction to dynamical systems and chaos
Lotka-Volterra systems
Individual-based modeling of predator/prey systems
SIR models

Readings:
Review Mitchell Ch. 2
G. Flake *The Computational Beauty of Nature* Ch. 12
Newman *Networks* Ch. 17, pp. 627 - 641

**Optimization methods** (2 weeks)
**Assignment 4: To be announced**
Topics:
- Search spaces, random search, hillclimbing
- Genetic algorithms
- Applications

Readings:
- Mitchell Ch. 5, 6, 9, 18
- G. Flake *The Computational Beauty of Nature* Ch. 20

**Game Theory and Modeling Social Systems** (1 week)
Topics:
- Non-zero sum games and the Iterated Prisoner's Dilemma
- Hawks and Doves

Readings:
- Mitchell Ch. 14 (213-224)
- G. Flake *The Computational Beauty of Nature* Ch. 17
- R. Axelrod and D. S. Bennet *A Landscape Theory of Aggregation*
- T. Maurer and R. Morgus *Tipping the scale: An analysis of global swing states in the Internet governance debate*

**Bayesian Models** (1 week)
Topics:
- Bayes Rule
- Computing with Bayes Rule
- Applications

Readings:
- E. Charniak "Bayesian Networks Without Tears"
- T. Mitchell *Machine Learning* Ch. 6, McGraw Hill, 1997

**Review and catchup** (1 week)
Readings:
- H. Simon "The architecture of complexity"