CSci 501 Introduction to Computational Sciences

Course Syllabus

Fall 2015

Instructor

Derek Harter, Ph.D., Associate Professor
Department of Computer Science
Texas A&M University - Commerce
Office: SCI 355
Office Hours: M, T, Th 1 - 3 pm
E-mail: Derek.Harter@tamuc.edu

Class Meetings

Lectures and course materials will be distributed through our University's eCollege online course system.

01W 82527 Web Based Class

Course Description

Big scientific data sets are growing exponentially both in size and complexity. Extracting meaningful information from this data requires not only programming skills, but also understanding the analysis work-flows and mathematical models and visualization tools that help to condense large amounts of information into a comprehensible story. Computational science is the scientific investigation of problems through modeling, simulation and analysis of physical processes on a computer. Computational science is now considered by most scientists to be on par with the development of scientific theory and the use of experimentation in order to understand more about our world. Computational science is not the same as computer science. Rather, it is an interdisciplinary blend of scientific models, applied mathematics, computational techniques, and practices. This Introduction to Computational

Science course focuses upon simple and intuitive computational models and methods.

Requirements and Objectives

This course forms one of the core subjects in the new master's degree program in the Computational Sciences. Computational sciences differs from the traditional computer science discipline in several ways, but most importantly as being focused on applying computational methods to solving large scientific problems. Thus this type of scientific data analysis of large complex data sets is both increasingly crucial to scientific research, as well as being in great demand for practitioners who can apply computational analysis and modeling to such data sets. This course directly addresses this area, and forms a cornerstone subject for any student wishing to understand and practice computational science research.

Goals include:

- Understand the scientific process and the philosophy of science.
- Understand the purpose and value of computational science.
- Be exposed to the common tools and practices of working computational scientists.
- Learn to use basic computational simulation and modeling tools, specifically the Python toolkit stack.
- Be exposed to basic data analysis and modeling concepts and practices.
- Learn about common computational algorithms for performing scientific modeling, including computational integration and differentation, random Monte Carlo methods, solutions of ordinary differential equations, etc.

Companion Textbooks / Readings

Most materials, work and readings will be provided as iPython notebooks for students to read and experiment with. Many materials were developed using the following sources. Additional readings from these sources are encouraged for a deeper understanding of the topics:

- Langtangen, H.P. (2012). A Primer on Scientific Programming with Python 3rd ed. Springer.
- Newman, M. (2012). Computational Physics. CreateSpace.

Prerequisites

Successful enrollment in Computational Sciences Master's Program. This course assumes a basic proficiency with computer programming and computational concepts. A basic understanding of how a sequential computer works is assumed. Fundamental familiarity with basic CS topics such as algorithms and analysis, data types, data storage, I/O, loops, branches, subprograms and object-oriented programming are also assumed. Fundamental mathematical ability is assumed, such as familiarity with discrete and continuous mathematical models. Enough ability to use vectors, matrices, integration and differentiation. The REAL prerequisite is a desire to learn and explore new ideas.

Evaluation (Tentative)

Your grade for the course will be based on the following (approximate) percentages:

Two Exams	50% (25% each)
Labs / Programming Assignments (appx. 6-8)	50%

Letter grades will be assigned according to the following scale:

Final Average (%)	Letter Grade
90 - 100	A
80 - 89	B
70 - 79	C
60 - 69	D
Below 60	F

Student's with Disabilities

The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you have a disability requiring an accommodation, please contact: Office of Student Disability Resources and Services, Texas A&M University-Commerce, Gee Library, Room 132, Phone (903) 886-5150, StudentDisabilityServices@tamuc.edu

Academic Ethics

"All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct). Ethics also includes the issue of plagiarism, and copying code for programming/lab assignments is just as serious as any other type of plagiarism. If you are caught sharing or using other people's work in this class, you will receive a 0 grade and a warning on the first instance. A subsequent instance will result in receiving an F grade for the course, and possible disciplinary proceedings.

Attendance Policy

Students are expected to follow all instructions and visit eCollege regularly many times weekly to complete the materials for this online course. If a student is unable to submit assignments by the due date for the assignment, they are expected to make alternative arrangements to assure that the assignment is turned in ON TIME, before the assignment is actually due. Any student wishing to withdraw from the course must do so officially as outlined in the class schedule. THE INSTRUCTOR CANNOT DROP OR WITHDRAW ANY STUDENT.

Course Requirement Deadlines

Credit will be given for ONLY those exam(s), program(s), and/or project(s) turned in no later than the deadline(s) as announced by the instructor of this class unless prior arrangement has been made with the instructor.

Course Schedule (Preliminary)

W	Date	Topic / Activity	Assg	Test
1	8/31	Introduction to Computational Science and Course, Introduction to Python		
2	9/7	NumPy - Python Numerical Computing Library		
3	9/14	Matplotlib, Mathematical Functions and vectorization, Discrete Approximations	#1	
4	9/21	Speed and Accuracy in Numerical Simulations	#2	
5	9/28	Sequences and Difference Equations		
6	10/5	Taylor Expansion	#3	
7	10/12	Integration and Differentation		
8	10/19	Integration and Differentation	#4	Midterm
9	10/26	Numerical Solutions of Ordinary Differential Equations		
10	11/2	Numerical Solutions of Ordinary Differential Equations	#5	
11	11/9	System of Equations, Discrete Grid/Cell Models		
12	11/16	System of Equations, Discrete Grid/Cell Models	#6	
13	11/23	Random Numbers, Statistics and Monte Carlo Simulations		
14	11/30	Random Numbers, Statistics and Monte Carlo Simulations	#7	
15	12/7	Large Scale Network Models and Network Science Methods	#8	
	12/14	Finals Week 12/14 - 12/18		Final

Student Learning Outcomes

1. Learn basic scientific modeling paradigms, discrete and stochastic models.

2. Apply computational techniques to tackling scientific research questions.

3. Familiarize with standard tools of computational science: HPC, R/Python/Numpy/Scipy toolkit stack.

Learning outcomes will be measured through mapping assignment and test questions to specific outcome items, as well as through exit surveys of student experiences with the outcome familiarity.