# CSci 538 Artificial Intelligence (Machine Learning and Data Analysis)

## 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    81217    Web Based Class

## Course Description

This semester's AI course will focus on the sub field of machine learning with some work in the analysis of big data. We will look at classic unsupervised and supervised learning methods, used in the field to classify data, cluster it and find optimizations. This will include looking at k-means and hierarchical clustering, self-organizing maps, linear regression, decision trees, optimization techniques such as genetic algorithms, etc. We will cover ways of getting hold of interesting datasets, ideas on how to collect data from users, and many different ways to analyze and understand the data once you've found it.

# Requirements and Objectives

- Explore downloading and mining real web data sets.

- Learn about unsupervised methods for grouping and visualization.

- Program optimization algorithms to search for optimal solutions using hill climbing, GA's, etc.

- Become familiar with some advanced classification techniques of the kernel methods family of algorithms.

# Companion Textbooks / Readings

CI Programming Collective Intelligence by Toby Segaran, O'Reilly Media, 2007, ISBN: 0596529325

ML Marsland. (2009). Machine Learning: An Algorithmic Perspective. By Marsland, CRC Press, 2009.

PY How to Think Like a Computer Scientist: Learning with Python 2ed by Jeffrey Elkner, Allen B. Downey and Chris Meyers (Open Book Project) http://www.greenteapress.com/thinkpython/

# Prerequisites

Background in basic programming (Undergraduate equivalent of CSci 151/152) and Data Structures (Undergraduate equivalent of CSci 270) or equivalent knowledge will be assumed for this course.

# Evaluation (Tentative)

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

| | |
|---|---|
| Two Exams (Midterm and Final) | 50% (25% each) |
| Labs / Programming Assignments (appx. 4-5) | 45% |
| Class Participation | 5% |

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 WITH-
DRAW 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 | Notebook | Test |
|---|------|------------------|----------|------|
| 1 | 8/31 | Course Introduction, Set up Python | 1a & 1b | |
| 2 | 9/7 | Introduction to Python Scientific Libraries | 2a, 2b, 2c | |
| 3 | 9/14 | Linear Regression with One Variable | | |
| 4 | 9/21 | Linear Regression with Multiple Variables | | |
| 5 | 9/28 | Logistic Regression and Regularization | | |
| 6 | 10/5 | K Nearest Neighbor | | |
| 7 | 10/12 | Decision Trees and Ensembles | | |
| 8 | 10/19 | | | Midterm |
| 9 | 10/26 | Support Vector Mechines (SVM) and Kernel Methods | | |
| 10 | 11/2 | Unsupervised Learning and Dimensionality Reduction | | |
| 11 | 11/9 | Principle Component Analysis (PCA) | | |
| 12 | 11/16 | K-means and Hierarchical Clustering | | |
| 13 | 11/23 | Anomaly Detection and Recommender Systems | | |
| 14 | 11/30 | Large Scale Machine Learning | | |
| 15 | 12/7 | Advice for Applying Machine Learning | | |
| | 12/14 | Finals Week 12/14 - 12/18 | | Final |

# Student Learning Outcomes

1. Develop familarity with high-level Python scripting language

2. Learn basics of fundamental machine learning techniques, such as regression, clustering, k-nearest neighbor, kernel methods, etc.

3. Learn basic distinction between supervised and unsupervised machine learning methods.

4. Show examples of using Web 2.0 data sources for systems development.

5. Learn basic machine learning training and testing techniques, including cross validation and data optimization.

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.