PURDUE
UNIVERSITY

SCHOOL OF ELECTRICAL
AND COMPUTER ENGINEERING
UNDERGRADUATE COUNSELING OFFICE

Engineering Faculty Document 96-19
February 8, 2019
Page 1 of 1

To: The Engineering Faculty
From: School of Electrical and Computer Engineering
Re: ECE 20875

The School of Electrical and Computer Engineering has approved the following new course. This action is now submitted to the Engineering Faculty with a recommendation for approval.

ECE 20875 Python for Data Science
Semesters offered: Fall, Spring
Non-repeatable
Credit 3

Pre/Co-requisites: CS 15900, minimum grade of C-

Requisites by Topic: Basic programming knowledge

Course Description
This course will introduce Python programming to students through data science problems. Students will learn Python concepts as well as introductory data science topics, and will use their knowledge of Python (and prior programming experience) to implement data analyses.

Reason
Content from current ECE 36400 has been evaluated and modified to remove the lab component and add a focus on using python for a focus on data science.

History of Previous Offering
This is the first offering of this course as it will be a replacement of ECE 36400 for those students entering Purdue for the Fall 2019 semester (202010 catalog term).

Michael R Melloch

Michael R. Melloch, Associate Department Head of ECE

ECE 20875:
Python for Data Science

Fall 2019

Lectures: TBD
Course web page: TBD
Piazza discussion page: TBD
Course calendar: TBD

Instructors:

Milind Kulkarni **(**milind@purdue.edu)
Office: EE 324A
Office Hours: TBD

TAs:

TBD

TA Office Hours: TBD

Course Outcomes: A student who successfully fulfills the course requirements will have demonstrated:

1. An understanding of regular expressions [1, 2]
2. An ability to use Python to write data analyses [1, 2, 6]
3. An ability to explain when particular data analyses are appropriate [1, 2, 3, 6]
4. An ability to explain the results of data analyses [2, 3, 5]
5. An ability to incorporate classes in their Python programs [1, 2, 6]
6. An ability to incorporate associative arrays in their programs [1, 2, 6]
7. An ability to work with a partner to choose appropriate analyses to solve a problem, perform those analyses, and interpret the results of those analyses [1, 2, 3, 5, 6]

These outcomes are extremely high level. In more detail, after taking this course you will be able to:

- Perform basic scripting tasks in Bash and Python (e.g., redirecting the output of one program to the input of another, setting and reading environment variables)
- Write programs in Python that incorporate: basic control structures; functions; data structures such as lists, tuples and associative arrays; classes and objects; iterators and generators; higher order functions; and regular expressions
- Understand the basics of sampling, estimation, and hypothesis testing
- Write data analyses that perform: visualization; textual analysis; regression; classification; clustering

- Perform more complex analyses using approaches like neural networks.

Course assessment: The achievement of the course objectives will be through 10 programming assignments (covering outcomes 2, 4, 5, and 6), two midterms (covering outcomes 1, 3, and 4), and a final project (covering outcome 7)

Course grading: Grades will be assigned as follows:

50% — Programming assignments (5% per assignment)

20% — Midterms (10% each)

30% — Final project

5% bonus — Class participation

Class participation will be assessed by participation in class as well as on the Piazza discussion board.

Programming assignments: Programming assignments will be due each week, except for the weeks of the midterms and during the final three weeks (which is set aside for project development). They will test the concepts covered in class, both programming and statistical. The following rules apply to *all* programming assignments:

1. All programs should run correctly in the versions of Python available on the Scholar cluster. You will have a Scholar account created for you at the beginning of the semester.
2. Your assignment submissions should be either (a) a Jupyter notebook (and any other accompanying files) with code that will produce the required output *and writeup*; or (b) a python script (or scripts) that will produce the required output when run as well as a *separate* writeup.
3. Unless otherwise specified, assignments are due at 11:59 PM on the deadline.

Programming assignments will be submitted via GitHub Classroom (**https://classroom.github.com**). As such, *you are required to have a GitHub account.* These can be obtained for free at **https://github.com**.

Please fill out the form here: **https://goo.gl/forms/CscnGcV8blMski1z2** to provide your GitHub account information.

Late submission policy: Except for medical and family emergencies (accompanied by verification), there will be *no individual extensions* granted for programming assignments. Late submissions will be scaled according to lateness, docking 10% from your score per day late, up to a maximum of 50%. Submissions more than 5 days late will be assigned a score of 0.

Course Discussion: This term we will be using Piazza for class discussion. If you have questions about the course or the project, we encourage you to post them on Piazza. It's a shared discussion forum, where your question can be answered by the instructors, the TAs or your fellow students!

Find our class’s Piazza page at: **TBD**

Students who are active participants on Piazza may receive class participation bonus points.

Email: Questions about course material or programming assignments *should be posted to Piazza* or raised during lecture or office hours. The professor and TAs will not answer programming questions via email. This is to allow other students who might have similar questions to benefit from our answers. Of course, if you have questions of a personal or confidential nature, we welcome your email.

Course announcements: Course announcements, including changes in due dates, course topics, programming assignment details, etc., will be communicated in three ways:

1. Posts on Piazza.
2. Changes on the shared course calendar.

Course Schedule: Below is a rough schedule of what will be covered in class:

Week	Topic(s) covered
1	Bash scripting (environment variables and input/output redirection); Git basics; Python basics (control structures, functions)
2	Python data structures (lists, tuples, associative arrays)
3	Histograms; probability distributions
4	Higher-order functions; map/reduce
5	Regular expressions; basic text processing
6	More regular expressions; n-gram analysis; **Midterm 1**
7	Sampling, estimation, and hypothesis testing
8	Hypothesis testing (cont.); regression (ordinary least squares; autoregressive models)
9	Classes and objects; clustering (k-means, mixture models)
10	Objects (cont.): iterators and generators; classifiers (kNN, naive Bayes)
11	Objects (cont.): inheritance; perceptrons
12	Neural nets; **Midterm 2**
13	Neural nets (cont.)

Week	Topic(s) covered
14	Project lectures (special topics)
15	Project lectures (special topics)

Academic Honesty: Unless expressly allowed, you are expected to complete all assignments by yourself. However, you are allowed to discuss general issues with other students (programming techniques, clearing up confusion about requirements, etc.). You may discuss particular algorithmic issues on Piazza (but do not copy code!). *We will be using software designed to catch plagiarism in programming assignments, and all students found sharing solutions will be reported to the Dean of students.*

Punishments for academic dishonesty are severe, including receiving an F in the course or being expelled from the University. By departmental rules, all instances of cheating will be reported to the Dean. On the first instance of cheating, students will receive a 0 on the assignment; the second instance of cheating will result in a failure of the course.

Campus Interruptions: In the event of a major campus emergency, course requirements, deadlines and grading percentages are subject to changes that may be necessitated by a revised semester calendar or other circumstances beyond the instructor's control. In such an event, information will be provided through the course website, Piazza, and email.

Week	Topic
1	Bash scripting, basic Python
2-3	Higher order functions, map/reduce, regular expressions, basic text processing
4-5	Regular expressions, N-gram analysis, probability distributions
6-7	Data structures: lists, arrays, associative arrays, sampling and estimation
8	Iterators and generators, regression
9	Objects, Clustering (k-means)
10	Classifiers (kNN, naive bayes)
11-12	Perceptrons, neural nets, PyTorch
13-15	Project