JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Syllabus
Applied Mathematics and Statistics 553.488/553.688
Financial Computing I
Fall, 2019
(3 credits, EQ)

Description

This course is aimed for student who need to develop the ability to use programming languages to solve data analytic problems that are likely to arise in financial applications. The Python programming language will be introduced, and students will be exposed to libraries that are commonly used.

In the past, this course introduced C and C++, but demand for these, especially for those who do not plan to become professional programmers, apppears to be dwindling. So we might not cover much or any C/C++ in this course.

Prerequisites

Examples and exercises will be used to introduce programming concepts and particulars of the languages. The applications will use ideas from probability and statistics. For this reason, it is strongly recommended that students be familiar with material from a calculus-based probability and statistics course such as EN.550.310 OR EN.550.311 OR (EN.550.420 AND EN.550.430).

Students should be comfortable using computers but no prior programming background is required.

Some finance background is helpful but not necessary.

Instructor

Professor Daniel Q. Naiman, daniel.naiman@jhu.edu, www.ams.jhu.edu/~dan

Office: Whitehead 202-C, 410-516-7203

Office hours: Monday 3-4:30, Wednesday 3-4:30 and by appointment

Teaching Assistants

Mingzhong Gao – mgao24@jhu.edu

Yitian Li – yli312@jhu.edu

Ke Chen – kchen78@jhu.edu

Office hours: See TA Info in Blackboard

If you expect that you will need to make use of instructor or TA office hours, and none of the posted times work for you, please let me know and we will try to make an accommodation.

TA office hours are held in Room 212 Whitehead Hall

Lectures

Monday, Wednesday 1:30-2:45, Olin 305

Textbook and online resources

Since there are many fine materials available online, you are not expected to purchase a textbook for this course.

There are many introductory Python tutorials available on the web, including these:

Python software foundation: https://docs.python.org/3/tutorial/

W3Schools.com: https://www.w3schools.com/python/

Tutorials point: https://www.tutorialspoint.com/python/index.htm

At the outset, when we focus on the foundations of the Python language and its standard library, the Python software foundation's documentation will be an important resource:

Python standard library: https://docs.python.org/3/library/index.html

Python language reference: https://docs.python.org/3/reference/index.html

This is a nice book that covers a lot of material in the data science area that we will touch upon as the course progresses:

Python Data Science Handbook by Jake VanderPlas: https://jakevdp.github.io/PythonDataScienceHandbook/

This site has some nice notes on specific topics:

Python tips: http://book.pythontips.com/en/latest/index.html

Please log in to Blackboard for all materials, including

- lecture notes
- handouts
- links
- homework assignments
- assessments

related to this course.

Important Software Requirement

Students are expected to work on exercises and submit work for specific platform: Anaconda Python 3.7 using Jupyter notebooks.

For students who do not have laptops, either for the duration of the course, or temporarily, this platform is available in the computer lab on the 3rd floor of Whitehead Hall (Room 302). There are 10 computers in the lab and they use JHED authentication.

Course Objectives

Broadly speaking, students should

1) Develop an understanding of some basic programming concepts.

2) Develop the ability to program in Python and understand nuances of the language.

3) Learn good programming habits.

4) Develop the capability of using programming to solve mathematical problems related to finance.

Python Topics

- Jupyter notebooks
- Programming style
    - Creating readable code
    - Improving code organization
    - Commenting code
- General programming concepts
    - Data types
        - scalars
        - strings
        - lists
        - dictionaries
        - tuples
        - sets
    - Assignment
    - Comparison, Boolean expressions/operations
    - Aliases/Copies
    - Sequence types, operations
    - Ranges
    - File input/output
    - Properties of objects (e.g. mutability, methods, attributes)
    - Control flow
    - Object oriented programming, Classes
    - Memory addresses
    - User-defined Functions
    - Scope
    - Use of packages & environments
    - Containers, Iterators
    - Handling exceptions
- Special topics

    - string methods
    - opening, closing, writing to, reading from files
    - with, as
    - lambda functions
    - iterators, next
    - map
    - zip
    - sorting
    - generators
    - byte arrays, byte objects
    - unicode
    - json
    - pickle & serialization
    - requests
    - api's
    - hashes
    - debugger
    - timing
- Specific packages
    - numpy
    - matplotlib
    - pandas
    - math
    - multiprocessing
    - regular expressions
    - scikitlearn
    - tensorflow (time- and resource- permitting)

Financial topics

- analysis of financial data
    - stochastic process/stochastic differential equation models for price evolution
    - simulation and the Monte-Carlo method
    - exploring statistical relationships
    - graphical representations
- order book implementation
- analysis of loan performance data
    - logistic regression
    - decision trees
    - additional machine learning methods
- crypto-currencies

Course Expectations & Grading

Student evaluation will be based on 2 components:

1) periodic homework assignments/projects (50%)
   - there will be a 5% of total score per day penalty for homework handed in late
2) quizzes (50%)

Assignments & Readings

All assignments (homework/reading) will be posted in Blackboard

The last assignment in the course will be due on the scheduled date of the final exam for this course (Friday May 11th).

Ethics

The strength of the university depends on academic and personal integrity. In this course, you must be honest and truthful. Ethical violations include cheating on exams, plagiarism, reuse of assignments, improper use of the Internet and electronic devices, unauthorized collaboration, alteration of graded assignments, forgery and falsification, lying, facilitating academic dishonesty, and unfair competition.

The following is an important policy statement concerning the manner in which you are expected to complete homework assignments.

> You are allowed, indeed encouraged, to discuss homework problems with other students, and share algorithmic ideas. However, when you proceed to writing a computer program, you must carry out this step completely on your own. Copying code produced by another person, or copied from the internet is considered an ethical violation. Code submitted in assignments will be checked for plagiarism.

Report any violations you witness to the instructor.

You can find more information about university misconduct policies on the web at these sites:

- For undergraduates: https://studentaffairs.jhu.edu/policies-guidelines/undergrad-ethics/
- For graduate students: https://krieger.jhu.edu/hwgradaffairs/wp-content/uploads/sites/35/2017/10/Homewood-and-EP-Graduate-Academic-Misconduct-Policy-October-2017.pdf

Students with Disabilities

Any student with a disability who may need accommodations in this class must obtain an accommodation letter from Student Disability Services, 385 Garland, (410) 516-4720, studentdisabilityservices@jhu.edu .

ABET Outcomes

- Ability to apply mathematics, science and engineering principles.
- Ability to design and conduct experiments, analyze and interpret data.
- Ability to design a system, component, or process to meet desired needs.
- Ability to identify, formulate and solve engineering problems.
- Ability to use the techniques, skills and modern engineering tools necessary for engineering practice.