ISC-4304-01: Programming for Scientific Applications

based on slides from Gordon Erlebacher and Xiaquiang Wang

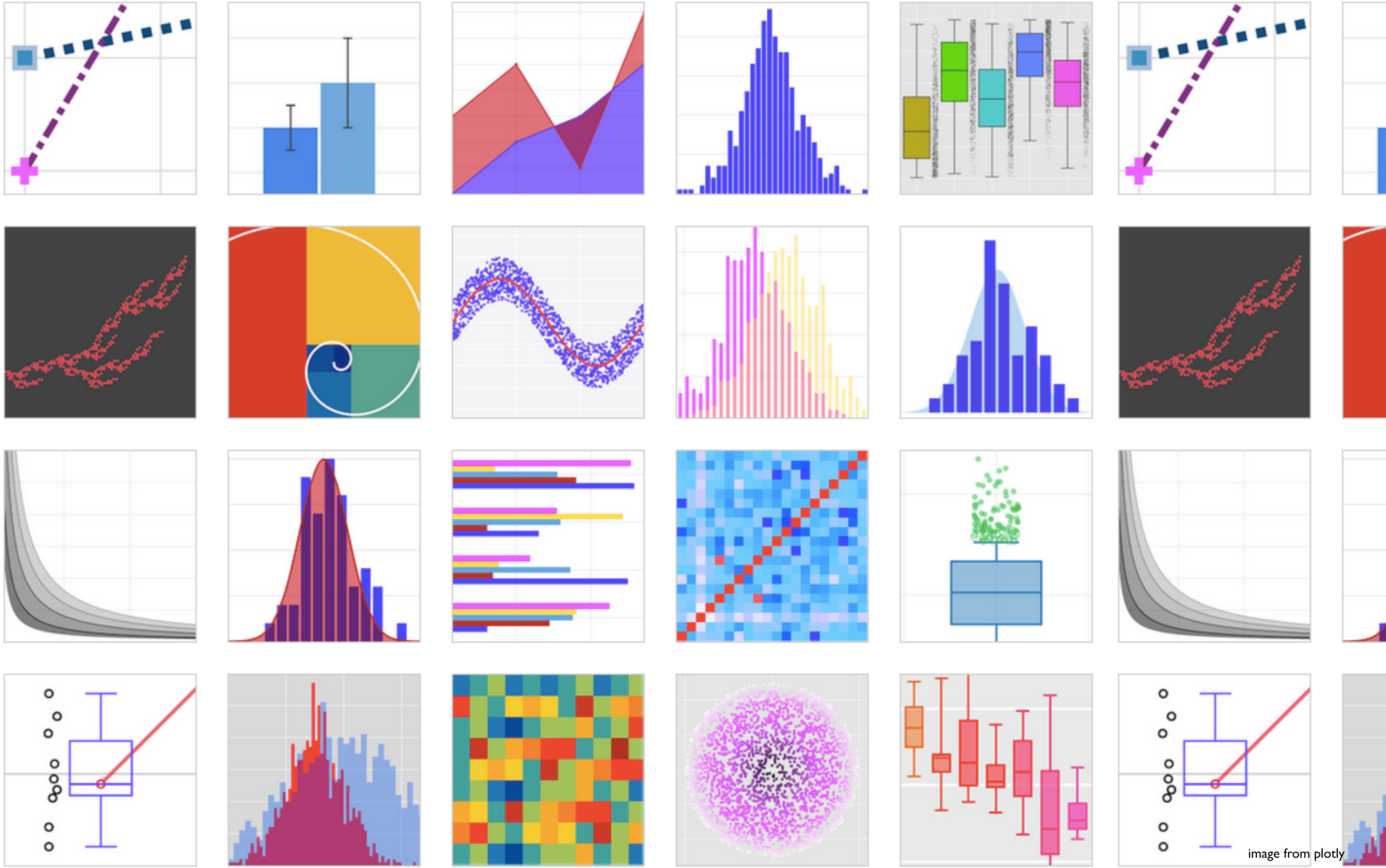

image from plotly

who are you?

tell us your name, your major, and one of your interests

Peter Beerli

$$P(G,E,S) = P(G)\,P(E|G)\,P(S|E,G)$$

$$P(G) = \exp\left(-u\left(\sum_i \frac{k_i(k_i-1)}{\theta_i} + \sum_j \sum_i k_i M_{ji}\right)\right)\left(\sum_i \frac{k_i(k_i-1)}{\theta_i} + \sum_j \sum_i k_i M_{ji}\right) = \lambda \exp(-u\lambda)$$

M_{ji} = migration rate scaled from j to i

θ_i = Population size

$$P(E|G) = \frac{\eta}{\lambda}\sum_j \partial_j + \frac{\circ}{\lambda}\left(1 - \sum_j \partial_j\right)$$

∂_j $\begin{cases} 0 & \text{if there is a migration} \\ 1 & \text{if there is a coalescent event} \end{cases}$!

$$P(S|G,E) = \frac{k_j(k_j-1)}{\theta_j}\partial_j + \frac{\sum k_i M_{ji}(1-\partial_j)}{\upsilon}$$

There are $\frac{2}{k_j(k_j-1)}$ possible coalescent events on each j

who are you?

tell us your name, your major, and one of your interests

ISC4304-1 PROGRAMMING for SCIENTIFIC APPLICATIONS Spring 2018

Instructor

Peter Beerli
Office: 150-T DSL
Email: beerli@fsu.edu
Phone: (850) 559-9664

Teaching Assistant

Kyle Shaw
Office: DSL
Email: kjs16c@my.fsu.edu
Phone: TBA

Lectures (Beerli):

Monday, Wednesday, Friday 10:10am-11:00am
DSL152

Lab-session (Shaw):

Monday 2:30pm - 5:00pm
DSL 152

Office Hours

Monday 1:30pm-2:30pm or (preferred) by appointment (Beerli)
TBA (Shaw)

Textbook

No textbook required; but you may need to read short articles supplied to you.

Overview

This course provides knowledge of a scripting language (python) that serves as a front-end to popular packages and frameworks, along with a compiled language (C++). Students will study and practice object-oriented scientific programming with the scripting and compiled language. In the laboratory component of the course students will apply the concepts learned in several science applications. Prerequisite: MAC 2312.

Learning objectives

The students will be able to ...

- evaluate the benefits of interpreted and compiled languages and know when to use each one to best advantage
- practice the conversion of mathematical formulae to computer algorithms and these to functional programs.
- understand the Python sufficiently well to program small applications with confidence
- incorporate basic algorithms coded in C++ into modules that are used in python
- through lab work, develop the skills to apply Python and C++ to a range of practical scientific applications, ranging from processing of scientific data to visualization.

Goal/Rational for the course

The students will study a scripting language, python, and will practice its use and will be able to write small programs to solve problems that can only be solved numerically. These programs are often very slow, but can be improved by incorporating other computer languages such as C++, students will practice such incorporations and also will evaluate how such changes improve the speed of the computer program.

Grading

The grade for the course will be based upon labs, homework, a midterm and a final exam. This work is weighted as follows:

- Midterm Exam (March 19 Thursday 10:10am-11:00pm)- 25%
- Final Exam (May 3th Thursday 10am-12am) - 20%
- Homework (weekly with lapses) - 15%
- Quizzes (irregular) - 10%
- Labs (weekly) - 30%
- Attendance – will be used to round up/down in cases when the grade will be **near [2 points]** a change point.

Final grades will be given as:
A: 100-93, A-: 92-90, B+: 89-87, B: 86-83, B-:82-80, C+: 79-77, C: 76-73, C-: 72-70, D+: 69-67, D:66-63,D-:62-60, F 59-0); rounding as usual at 0.5.

Attendance

You should attend class. If you fail to attend, I would like to know why; but I will not penalize explicitely for occasional/rare missed classes [but see under grading]. Exemptions are accepted for sickness (please stay home if have a fever) and the attendance of scientific conferences or sports events (if you are on the FSU team). Students, not the professor, are then responsible for bringing themselves up to date both on subject matter covered during class, as well as completing homework assignments in a timely manner. Information given in class supplants information provided on the course web site.

Courtesy

Keep your cell phone in your pocket!

You must get to class on time because often informations and instructions given at the beginning class; late arrivals will be noted. If you must leave class early, please let the instructor know **before** class begins.

Late Assignments

You can turn in ONE laboratory assignment and ONE homework late with no questions asked and no penalty; however, the assignment must be turned in no later than 1 week after its due date. Additional late assignments will be penalized by applying a 10% reduction for the first two days late, 20% for days 3-7, 50% for days 8-14, after that the homework or assignment will get 0 points. Exceptions to these rules are made only if extenuating circumstances (such as illness, etc.) arise which can be documented.

Student responsibilities

Learning a new programming language is difficult, you will need to prepare for every class by reviewing the past material because new material almost always builds up on knowledge already learned. Assignments can be discussed with other students but never use code from your friend, write it yourself! If you have problems see Kyle or me for help.

Lab 2: Lists

Due date: Thursday January 23, 11:59pm

1. The main task in this lab is to separate text into lists and do do some statistics on these lists.
   The data for this lab needs to be downloaded from

   `https://www.gutenberg.org/ebooks/search/?sort_order=downloads`

   pick a few books you like and report (you will need to download the textfile (plain text UTF-8)

   - The average sentence length.
   - The average word length.
   - How many 1-letter words, 2-letter words, ... 20-letter words are present.
   - What are the 100 most common words [use a dictionary] (20 points extra credit)

2. Send the python script and the output of your experiment to Philip (pb12c@my.fsu.edu) by email before the deadline. Your output should contain the book titles and the statistics.


```
#!/usr/bin/env python
# (c) yourname

#
def readbook(filename):
    '''
    Reads a textfile into a single string, ignoring line breaks.
    The function has a single argument: the infile name
    and it returns a soingle string containing the whole file
    '''
     with open(filename, 'r') as myfile:
         data=myfile.read().replace('\n', '').replace('\r','')
     return data

if __name__ == '__main__':
     data = readbook('testbook.txt')
# more stuff to follow
```

Class Objectives

- to understand the benefits of interpreted and compiled languages and know when to use each one to best advantage
- to understand Python sufficiently to program applications with confidence
- to understand C++ sufficiently to program applications
- learn to interface C++ and Python to each other, to take advantage of the best features of both languages
- through lab work, develop the skills to apply Python and C++ to a range of practical scientific applications, ranging from graphical user interfaces, web-based display of results, processing of scientific data, and visualization

Class Contents

- Overview and your first python program (Wednesday)
- basic syntax
- functions
- lists, dictionaries, sequences
- modules
- files, I/O
- interaction with the operating system
- text processing
- numpy for arrays
- classes (object orientation)
- numpy for statistics
- graphical user interfaces
- C/C++ and wrapping (speed up python for science applications)