Syllabus: HRT 891A: Foundation in Computational Plant Sciences

Fall 2019, MW 3:00-4:20 pm A186 PSSB, 3 credit hours

Instructors: Dr. Robert VanBuren and Dr. Dan Chitwood

Emails: bobvanburen@gmail.com; dhchitwood@gmail.com

Office hours:

Tuesday 2-4 Dr. VanBuren, A328 PSSB

Friday 11-1 Dr. Chitwood, A322 PSSB

If you cannot make scheduled office hours, please email the instructors to set up a time convenient for you to meet.

Course description

This course will bring together plant biologists and computational/data scientists to address grand challenges in plant biology. This project-based course will teach fundamental concepts in plant science through a computational lens. The course will begin at the organismal level with plant architecture, development, and morphometrics, followed by macroevolution of photosynthetic organisms. We will then transition into genetics and genomics and link the biochemistry and molecular biology of the cell to complex multi-level networks. Finally, these topics will be integrated to explore complex and emergent biological processes such as crop domestication and improvement, interactions of genotype and the environment, and global plant biology. The course will consist of a mix of lectures and thought experiments and hands on activities in Python implemented in Jupyter notebooks. Throughout the course, students will learn basic programing concepts implemented using cutting edge datasets and computational approaches. Foundation in Computational Plant Sciences is open to graduate students in any program and no prerequisite experience in coding or computational biology is required. Foundation in Computational Plant Sciences is the first course in the NSF-IMPACTS training program and is required for the new graduate certificate in Computational and data sciences. This course will be taught using a flipped classroom approach.

Course philosophy

This course assumes no prior experience in plant biology or coding. It is impossible to teach the breadth of introductory plant biology and data science in a single semester. Rather, this course selects specific examples in which mathematical and modeling approaches intersect with the biology of plants.

Coding and plant biology can be intimidating to those with no prior experience. This course is designed to make these disciplines welcoming. **A diversity of perspectives is vital to combat the grandest challenges of our time and there is no room for gate-keeping and shaming: Your contributions to the computational plant science community are needed and precious.** This course is meant as an

introduction, for you to **develop *your* skills** to **pursue *your* intended impacts**. There are no wrong answers: use this material in whatever way is useful for you to further your education and career. Share this material with others in the same spirit and spread your knowledge.

Above all, find joy in the way mathematics and modeling can reveal the underlying beauty of plants!

Plant biology students with no prior coding experience and data scientists with no knowledge of plant biology might find some aspects of the exercises tedious for which they are familiar, especially for the first lessons. This is not the intended spirit of this course. Rather, if familiar with presented topics, always look deeper, to the synthesis between plant biology and data science. What is the underlying biology of the unsolved problem? What mathematical concepts remain to be explored? What is the grand challenge? You can always take this material to the next level: if you have already mastered what is being presented, research further and push the boundary, hone your teaching skills and help others.

Course goals

The course will synthesize concepts from computational and data science with plant biology principles. The goals for this course are for students to:

(1) acquire foundational knowledge and skillsets in plant biology and data sciences

(2) formulate hypothesis-driven questions stemming from grand challenges at the interface of plant biology and data sciences

(3) synthesize the acquired skills to solve questions in collaborative teams

(4) communicate persuasively across disciplines with peers and the public

By the end of the course, students will be able to identify and create strategies to address current problems in plant biology using cutting edge computational tools and approaches. Through interactive lectures and 'thought experiments' students will learn to communicate across disciplines to explore emergent biological processes.

Required materials for class

In-class programming assignments are a critical part of the learning process in this course. To that end, you are expected to bring the following every day:

- Your laptop and its power cord
- An adapter for connecting your laptop to a VGA or HDMI cable (if you don't have one, you can borrow one from a classmate or the instructors)

There are no required textbooks for this course. Occasionally, there will be assigned readings from *The Algorithmic Beauty of Plants* related to the in-class activities for you to read.

This course uses Jupyter notebooks that will be uploaded to D2L for you to use. These notebooks are publicly available at the following links:

GitHub: https://github.com/DanChitwood/PlantsAndPython

Jupyter notebook viewer:
https://nbviewer.jupyter.org/github/DanChitwood/PlantsAndPython/tree/master/

Course activities

Class participation: Active class participation (led both by the instructor and by students) is critical to the success of this course. As such, you are expected to attend class every period, bring the required materials (most importantly, your computer and power cord) and to actively participate in the in-class discussion and programming activity. **Attendance will be taken every class period and is mandatory for earning attendance and participation points.**

In-class notebooks: A major feature of this course is the use of Jupyter notebooks. These notebooks will be made available on D2L before each class. It is important that you read over the notebooks and watch the video tutorials before class. Then, during class, within your assigned group you will complete the exercises. In addition to physical attendance, **submitting a completed notebook to D2L by 5:00pm of the day the notebook is due is necessary for full attendance and participation points.** The notebooks are graded on effort and completion. You are expected to work collaboratively with your group on completing the notebooks, but **it is expected that you alone complete the individual notebook that you upload to D2L for credit.** Academic dishonesty will be dealt with according to MSU policy, as stated at the end of this syllabus.

Midterm and Final

The midterm and final projects will test the application of the skills you have learned in this course and your ability to use computational skills to address real world questions in plant science. You will work within a group to analyze either a provided dataset or previously published data. There will be individual and group presentations for the midterm and final and you will turn in a documented Jupyter notebook as a record of your work. Both presentations and Jupyter notebooks for the midterm and final will be graded.

The dates of class periods to work on the midterm and final can be found at the end of the syllabus.

The midterm and final projects may be linked to an optional publication, in which case for authorship you will be expected to contribute and participate. If there is a publication linked to the midterm and final for this course, it will not be linked to your grade and it will be optional to participate. Authorship on such a publication will require remaining in communication with the instructors and co-authors after this course, contributing to writing and analysis if needed, and reading over draft manuscripts.

Assessment and grading

Graded assignments include a midterm and end of the semester final project as well as weekly attendance, participation, and completion of the Jupyter notebooks. The breakdown is as follows:

50% Attendance, participation, and completion of the in-class Jupyter notebooks

25% Midterm project

25% Final project

Grading scale:

4.0 >= 90%, 3.5 >= 85%, 3.0 >= 80%, 2.5 >= 75%, 2.0 >= 70%, 1.5 >= 65%, 1.0 >= 60%, 0.0 < 60%

Grades are not curved. **You are to collaborate with your classmates, not compete with them.**

There is no reason each and every one of you cannot earn a 4.0! If you show up to each class (and communicate with the instructors about excused absences), put in the time to go over and learn each notebook, and put in effort on the midterm and final projects, you can receive 100% of the points for this class.

Other important information

Course website: This course uses a Desire2Learn page for course organization, which can be found at http://d2l.msu.edu. Accompanying course information and Jupyter notebooks can be found at this website. All assignments will be handed in via Desire2Learn. Consult the class website for instructions.

Email: At times, we will send out important course information via email. This email is sent to your MSU email address (the one that ends in "@msu.edu"). You are responsible for all information sent out to your University email account, and for checking this account on a regular (daily) basis.

Class attendance: This class is heavily based on material presented and worked on in class, and it is critical that you attend and participate fully every week! Therefore, class attendance is absolutely required. **An unexcused absence will result in zero points for the day, which includes the in-class programming assignment points.** *Arriving late or leaving early without prior arrangement with the instructor of your session counts as an unexcused absence.* Note that if you have a legitimate reason to miss class you must arrange this ahead of time to be excused from class. **Three unexcused absences will result in the reduction of your grade by one step (e.g., from 4.0 to 3.5), with additional absences reducing your grade further at the discretion of the course instructor.**

Inclusive classroom behavior: Respectful and responsible behavior is expected at all times, which includes not interrupting other students, turning your cell phone off, refraining from non-course-related use of electronic devices, and not using offensive or demeaning language in our discussions. Flagrant or repeated violations of this expectation may result in ejection from the classroom, grade-related penalties, and/or involvement of the university Ombudsperson. In particular, behaviors that could be considered discriminatory or harassing, or unwanted sexual attention, will not be tolerated and will be immediately reported to the appropriate MSU office (which may include the MSU Police Department).

In addition, MSU welcomes a full spectrum of experiences, viewpoints, and intellectual approaches because they enrich the conversation, even as they challenge us to think differently and grow. However, we believe that expressions and actions that demean individuals or groups comprise the environment for intellectual growth and undermine the social fabric on which the community is based. These demeaning behaviors are not welcome in this classroom.

"Michigan State University is committed to fostering a culture of caring and respect that is free of relationship violence and sexual misconduct, and to ensuring that all affected individuals have access to services. For information on reporting options, confidential advocacy and support resources, university policies and procedures, or how to make a difference on campus, visit the Title IX website at www.titleix.msu.edu."

Academic Honesty: Intellectual integrity is the foundation of the scientific enterprise. In all instances, you must do your own work and give proper credit to all sources that you use in your papers and oral presentations – any instance of submitting another person's work, ideas, or wording as your own counts as plagiarism. This includes failing to cite any direct quotations (including code) in your notebooks, class debate, or written and oral presentations, or notebooks. Especially for coding, you are encouraged to google examples and copy and paste relevant code in your assignments; however, it is imperative that you **cite the source of the code you use.** Citing code is easily accomplished using comments or markdown in Jupyter notebooks. Much of your work for this course will be done in a group setting, and we encourage you to work collaboratively with your classmates. **We ask, however, that when working in a group, you focus on the exchange of ideas and use the group for inspiration, but that you write the code in your notebook by yourself.**

Article 2.III.B.2 of the Academic Freedom Report states: "The student shares with the faculty the responsibility for maintaining the integrity of scholarship, grades, and professional standards." In addition, the Department of Sustainability adheres to the policies on academic honesty specified in General Student Regulation 1.0, Protection of Scholarship and Grades; the all-University Policy on Integrity of Scholarship and Grades; and Ordinance 17.00, Examinations.

Therefore, unless authorized by your instructor, you are expected to complete all course assignments, including homework, lab work, quizzes, tests and exams, without assistance from any source. You are expected to develop original work for this course; therefore, you may not

submit course work you completed for another course to satisfy the requirements for this course. Students who violate MSU regulations on Protection of Scholarship and Grades will receive a failing grade in the course or on the assignment.

Faculty are required to report all instances in which a penalty grade is given for academic dishonesty. Students reported for academic dishonesty are required to take an online course about the integrity of scholarship and grades. A hold will be placed on the student's account until such time as the student completes the course. This course is overseen by the Associate Provost for Undergraduate Education.

Bereavement: Michigan State University is committed to ensuring that the bereavement process of a student who loses a family member during a semester does not put the student at an academic disadvantage in their classes. If you require a grief absence, you should complete the "Grief Absence Request" web form (found at https://www.reg.msu.edu/sitemap.aspx?Group=7) no later than one week after knowledge of the circumstance. We will work with you to make appropriate accommodations so that you are not penalized due to a verified grief absence.

Accommodations: Michigan State University is committed to providing equal opportunity for participation in all programs, services and activities. Requests for accommodations by persons with disabilities may be made by contacting the Resource Center for Persons with Disabilities at 517-884-RCPD or on the web at rcpd.msu.edu. Once your eligibility for an accommodation has been determined, you will be issued a Verified Individual Services Accommodation ("VISA") form. Please present this form to me at the start of the term and/or two weeks prior to the accommodation date (test, project, etc.). Requests received after this date may not be honored.

Using Python and Jupyter notebooks for this course

You can watch a video about how to install Python onto your computer and how to access the Jupyter notebook interface. The video also shows how to run simple code and markdown in Jupyter using Python:
https://www.youtube.com/watch?v=CDHRKQl_Pq8

Installing Python for this course:

Go to jupyter.org/install

Follow the instructions to install the Anaconda distribution of Python on your computer. Even if you already have a version of Python installed on your machine, we encourage you to go through this installation process as the assignments will assume that you are working with the same versions of the Anaconda Python packages that the instructors are using. If you already

specifically have Anaconda installed, we encourage you to update all of the Python packages (you may need to look up how to do this).

Instructions for downloading Anaconda (Python 3.7.x):

1. Go to the Anaconda Download webpage: https://www.anaconda.com/download/
2. Select the appropriate operating system (Windows | macOS | Linux) for your computer (it may auto-detect the correct operating system).
3. Download the Python 3.7 version (64 bit recommended).
4. Follow the online documentation to install Python for your specific operating system: https://docs.anaconda.com/anaconda/install/
5. Open the command line program on your computer.
    - On Windows, type CMD in the run box in the Start menu.
    - On Mac, type "terminal" in the spotlight search and run the "Terminal" application
    - On Linux, open up the "Console" application
6. Type "jupyter notebook" in the command line and hit enter.

If everything goes correctly, a browser window should open up with the Jupyter interface running. If things don't work, don't worry, we will help you get started.

Instructions for getting .ipynb notebook files into Jupyter:

iPython notebooks (also referred to as Jupyter notebooks) are files that end with the .ipynb extension. We will give you these files for all of your assignments, you will edit them, and turn in the edited files.

Once you have an .ipynb file you can load it into Jupyter using the "upload" button on the main "Files" tab in the Jupyter web interface. Just navigate to your .ipynb file, select it, and hit the open button.

Once you see your filename in the jupyter window you can just click on that name to start using that file.

Instructions for making a copy of Jupyter notebooks and turning them in:

When you are finished editing your IPython notebook and are ready to turn it in you will need to download the .ipynb file from the Jupyter interface.

1. With the notebook file open in Jupyter, go to the "File" menu, select the "Download as" menu option and then choose "iPython Notebook (.ipynb)"
2. Pick a place to save the file (The desktop is a good choice).
3. Make sure you make a copy of the .ipynb file for your own records.
4. **Name your file with your first and last name, followed by underscore, "NB", and notebook number. For example: DanChitwood_NB1.ipynb**

Detailed course schedule:

Date	Notebook	Title	Activity	Learning objectives
28-Aug-19	0	**Getting Started with Jupyter**	Download, start using, and become familiar with Jupyter notebooks. Write your very first Python code!	Successfully load and use a Jupyter notebook on each student's computer. Become familiar with Jupyter notebooks and run very simple first code in Python. Explain the structure and goals of the course.
2-Sep-19	Holiday			
4-Sep-19	1	**Intro to Plants And Python: Lists & Leaves**	Learn about the macroevolutionary history of plants using lists and different datatypes. Explore matplotlib plotting grapevine leaf shapes.	How to create variables, learn what different data types are, how to create a list, how to index lists and strings, how to modify lists, how to use print(), .append(), len(), str(), float(), int(), how to use matplotlib
9-Sep-19	2	**Calculating the Golden Angle with Loops**	Use loops to calculate the Fibonacci sequence and the golden angle.	How to write a for loop, how to use range(), how to write a while loop, using a counter, how to use a loop to modify a list
11-Sep-19	3	**How to Build a Sunflower**	Model a sunflower disc and its growth.	Use loops and math to model phyllotaxy, create double loops, create animations and model plant development dynamically, create and use models to understand natural phenomena
16-Sep-19	4	**Can a Biologist Fix a Fern?**	Metaphorically use genetic techniques to understand the algorithm for a fractal fern frond.	How to generate pseudo-random numbers, using .shuffle() and .seed(), Boolean logic statements, if/elif/else statements, using 'and', 'or', and 'in' in Boolean logic
18-Sep-19	5	**An Array of Fronds**	Explore the parameter space of the Barnsley fern using numpy arrays.	How to use arrays, how to index arrays, dealing with array dimensionality, how to use numpy functions on arrays, how to use arrays for data analysis
23-Sep-19	6	**Functions and Fractals**	Use L-systems to generate fractals and branching architectures.	How to use a dicitonary, how to retrieve dictionary values using keys, how to define functions, how to use the function outputs return and yield
25-Sep-19	7	**Flowering: Chance and time**	Use L-systems to model plant development and flowering using chance and determinism.	How to use L-systems, functions, and random number generators to model phenomena, how to design models to reflect biological reality

30-Sep-19	8	**Inflorescences Illuminated**	Model the architectures of racemes, cymes, thyrsi, and panicles using monopodial and sympodial growth.	How to use models to make advanced morphological models, use models to create representations of monopodial and sympodial growth, create complex graphical representations of models
2-Oct-19	9	**Grapevines in a Warming World**	Using centuries of European grapevine harvest dates, learn how to fit curves using polynomials or custom equations to model effects of climate change.	How to read in data, how to use dataframes, how to select dataframe columns, how to perform descriptive statistics using pandas, how to mask data, how to use .iloc to isolate data, plotting with seaborn, how to fit polynomial curves, how to fit custom curves
7-Oct-19	10	**A Passion for Passiflora**	Use morphometrics to analyze the shape differences between 40 *Passiflora* species and model heteroblastic shape changes through development	How to create developmental models, dealing with multivariate data, performing Principal Component Analysis (PCA)
9-Oct-19	Midterm project	**(Prep/Presentations)**		
14-Oct-19	Midterm project	**(Prep/Presentations)**		
16-Oct-19	Midterm project	**(Prep/Presentations)**		
21-Oct-19	11	**A Feeling for the Organism**	Brief introduction to the field of molecular biology and genomics. Use single molecule, real time sequencing (NanoPore) to sequence a plant genome.	Students should have a basic understanding of the central dogma of molecular biology (DNA -> RNA -> protein). Students should be able to describe (in simple terms) what a gene is and how this is related to the genome. Students should understand basic features of the genome (i.e. genes, cis-regulatory elements, methylation, chromatin dynamics) and the types of datasets associated with each.

23-Oct-19		**How to Build a Genome**	Connect to the MSU High Performance Computer Cluster (HPCC). Load and manipulate NanoPore genome sequencing data. Assemble a draft genome.	Students should have a basic understanding of the MSU HPCC system architecture and Slurm, job submission, and creating virtual environments. Understand how to visualize and manipulate large-scale genomic data. Understand the principles of overlap and genome assembly.
28-Oct-19	12	**Explore the Genome Within**	Map, annotate, visualize, and compare features of the genome, Compare genome architecture between different species.	Students should understand the different elements of the genome (e.g. genes, regulatory elements, repeats, Students will learn to visualize genome assemblies
30-Oct-19		**Quantifying Activity in the Genome**	Download various sequencing data types from the NCBI-SRA, align data to refernce genomes. Visualize alignments and quantify read depth.	Students should gain a basic understanding of large-scale genomic datasets that quantify activity in the genome (RNAseq, ChIPSeq, ATAC-Seq,) and relate this to biological function. Studnets will learn the principles of short read alignment, visualization, and quantification
4-Nov-19	13	**Expression Dynamics**	Analyze RNA sequencing data, quanitfy differential expression	Students should understand the basic principles of differential gene expression analysis. Students should understand the strengths and limitations of using bulk RNAseq to understand biological principles
6-Nov-19		**Single Cell Expression**	Analyze single cell RNAseq datasets, cluster single cell data into tissues	Sutdents should understand the principles of single cell analyses. Students should understand the importance of surveying expression on a cell level and be able to link this to complex tissues and functions across the organism
11-Nov-19	14	**From Cells to Networks**	Use quanitifed, genome-scale datasets to build gene regulatory networks. Link networks to KEGG pathways and biological functions	Students should understand the principles underlying co-expression and regulatory networks. Students should be able to link individual genes to pathways and networks
13-Nov-19		**The Tangled Field: Integrating Genomic Datasets**	Analyze and integrate diverse genomics datasets to build predictive models of function	Students should be able to link layers of genomic information to biological function
18-Nov-19	15	**Genetic Diversity and Relatedness**	Use genome variants to quantify crop diversity. Use clustering and phylogenetic approaches to assess relatedness and similarity	Students should gain a basic understanding of genetic diversity, relatedness, and phylo genetics. Students should be able to link genetic variability with phenotypes

20-Nov-19		**From Wild to Domesticated**	Use variants to identify loci contributing to traits of interest. Identify signatures of human selection in genomes	Students should gain a basic understanding of quantitative genetics and the principle of genetic association. Students should be able to link patterns of human selection to genetic architecture and agronomically important traits
25-Nov-19	Final project	**Plan Final projects**		
27-Nov-19	Holiday	**(no class)**		
2-Dec-19	Final project	**(Project Presentations)**		
4-Dec-19	Final project	**(Prpject Presentations)**		