Systems Biology – Exercises

Week 1: Introduction

Instructor and Assistant

- **Course Instructor**

Name: Yanlei GU

Email: guyanlei@fc.ritsumei.ac.jp

Courses:

Engineering Mathematics 2, Applied informatics (Deep learning), Computer Architecture, Embedded system, Data Science,

Systems Biology- Exercises

- **Teaching Assistant**

Name: Budi Darma Setiawan

Courses: Systems Biology and Systems Biology – Exercises

- **Systems Biology**

Type: Lecture, more theoretical

Instructor: Prof. MARUTSCHKE D. MORITZ

- **Systems Biology – Exercises**

Type: Exercises

You need knowledge learned from course 35063 to perform the programming Exercises in our class

(Strongly recommend student to take Prof. MORITZ's Lecture: Systems Biology)

Course Outline -1

- Students in this course build software-based models of biological functions.

- Students will learn models that simulate processes in systems biology, including neural networks, genetic algorithms, and visualization of organisms.

Course Outline -2

- The emphasis is on learning how these models form the basis for innovation and simulation, as well as for education and training.

- Students will learn the differences between the models as well as their use for specific applications.

Student Attainment Objective

1. Students will acquire the practical skills to use each technology with concrete exercises.
2. The fundamentals covered during the exercises are technologies widely used in artificial intelligence, optimization, and crowd behavior.
3. Students will be equipped to implement bio-inspired solutions with their programming language of choice in their studies and research.

Course Schedule

1- Introduction and Overview

Class structure, general information, and course overview.

2- Exercise: Paradigm of Nature and Bio-inspired computing

Systems based on evolution via natural-selection.

3- Exercise: Fractals

Self-similarity in nature and computer models.

Course Schedule

4- Exercise: Emergent Systems

Emergence in physical systems and biological systems.

5- Exercise: Artificial Life

Applications of simulated artificial life, robots, and synthetic biology.

6- Exercise: Cellular Automaton

Practical use of cellular automata.

7- Course review for week 2-6

Course Schedule

8- Exercise: Genetic Algorithm (1)

Simple examples and exercises to show applications and limitations of genetic algorithms.

9- Exercise: Genetic Algorithm (2)

Detailed view on mutation, crossover techniques, and fitness tests.

10- Exercise: Neural Network (1)

Single layer examples of artificial neural networks.

Course Schedule

11- Exercise: Neural Network (2)

Hidden layer and multi-layer artificial neural networks.

12- Exercise: Swarm Intelligence

Swarm behavior in nature and emerging swarm intelligence, following their adoption to computer models.

13- Exercise: Artificial Immune System

Examples of artificial immune systems.

Course Schedule

14- Exercise: Epidemiological Modeling

and Course review

Applications of epidemiological modeling, including viral, bacterial, and non-biological contagion agents.

15- Course review and Q&A

Grade Evaluation Method

- Assignment: 100%

About On-line Class

- In the case of BCP level 1-2:
  - -The Number of face-to-face class sessions: 6
  - -The Number of web-based class sessions: 9
- In the case of BCP level 3-4:
  - -All the classes will be the web-based class sessions

- We plan to use Zoom to conduct the web-based class.

- We are planning to provide live-stream by using Zoom (same link) in face-to-face classes.

- If there are any changes, we will announce on Manaba +R

Important note

- Consultations.

Office Hours: By appointment, e-mail: guyanlei@fc.ritsumei.ac.jp.
Note: Contact me if you are having any difficulties with the material. The sooner the better.

- Attendance.

Students are responsible for all material covered in this class.
Students who miss more than 5 classes without a legitimate reason will automatically receive an "F" for the course.

- Professional ethics.

The behavioral and ethical standards of Ritsumeikan University will be observed in all aspects of this course. Specifically, academic dishonesty (e.g. copying assignments or the like) will result in a grade F for the corresponding assignment, and in many cases - in a failing grade (F) for the course.

Other requirements

- **Device**

In each class, students need to use your own PC to program.

- **Programming Language**

We will mainly use Python for programming.

Please review and learn the basic knowledge of python programming language.

Introduction for Systems Biology

What is Systems Biology?

- Systems biology is an emergent field that aims at **system-level** understanding of biological systems.

- What does it mean to understand at "system level"?
  - Unlike molecular biology which focus on molecules, systems biology focus on systems that are composed of molecular components.

Systems Biology – English, http://www.systems-biology.org/000/001.html
H. Kitano, Systems Biology: a brief overview, Science, 295:1662-1664, 2002
H. Kitano, Computational Systems Biology, Nature, 420:206-210, 2002

What is Systems Biology?

Scope:

(1) Structure of the system, such as gene regulatory and biochemical networks, as well as physical structures

(2) Dynamics of the system, both quantitative and qualitative analysis as well as construction of theory/model with powerful prediction capability

(3) Control methods of the system

(4) Design methods of the system

Systems Biology – English, http://www.systems-biology.org/000/001.html
H. Kitano, Systems Biology: a brief overview, Science, 295:1662-1664, 2002
H. Kitano, Computational Systems Biology, Nature, 420:206-210, 2002

What is Systems Biology?

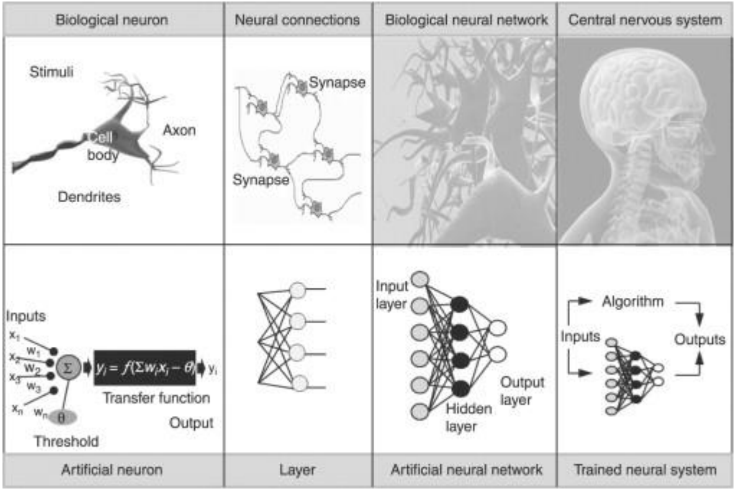
- Systems biology is based on the understanding that the whole is greater than the sum of the parts.

- Systems biology has been responsible for some of the most important developments in the science of human health and environmental sustainability.

Institute for Systems Biology is a nonprofit scientific research organization
https://isbscience.org/about/what-is-systems-biology/

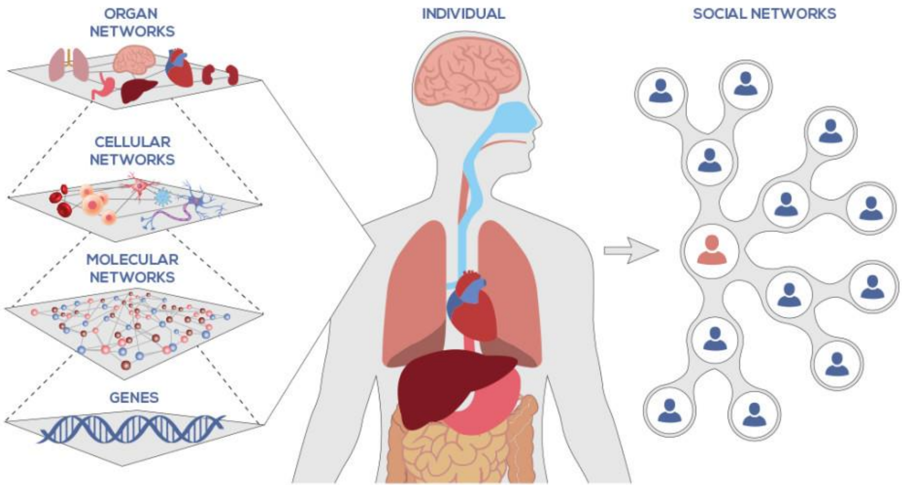
Research on Systems Biology
E.g. Bio-Inspired Artificial Intelligence

- Neural Systems → Deep Neural Networks



Orhan E. Arslan, in Artificial Neural Network for Drug Design, Delivery and Disposition, 2016, https://www.sciencedirect.com/topics/veterinary-science-and-veterinary-medicine/biological-neural-network

Network of Networks



Institute for Systems Biology, What is Systems Biology,
https://isbscience.org/about/what-is-systems-biology/

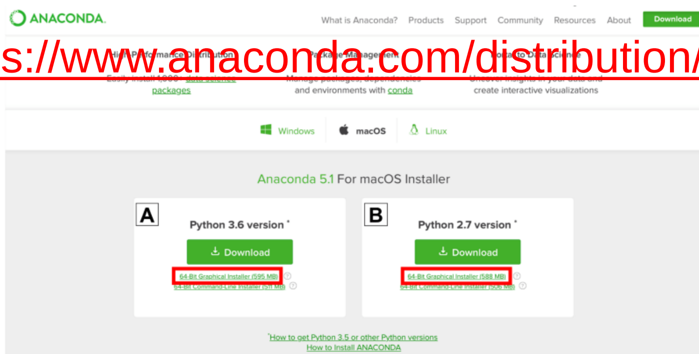
IDE (Anaconda) for Python

Students can use other IDEs.

Installing Anaconda on Mac OS X

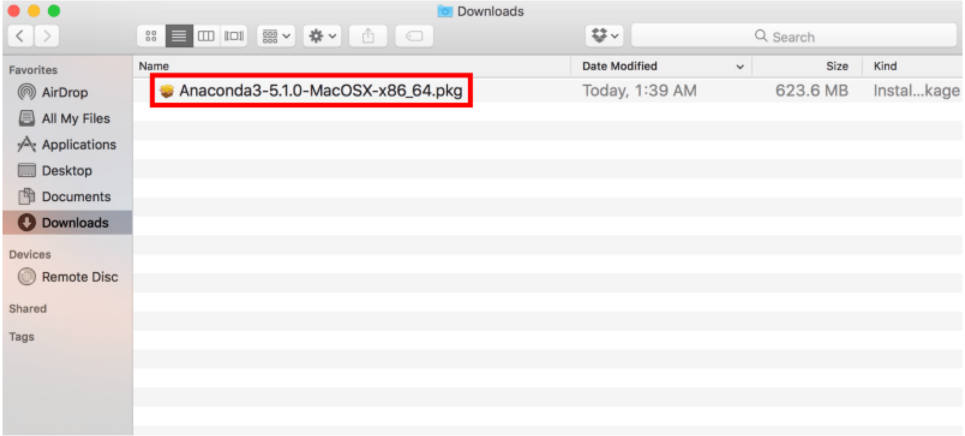
1 – Go to the Anaconda Website and choose a Python 3.x graphical installer (A) or ~~a Python 2.x graphical installer (B)~~. Please choose Python 3.

https://www.anaconda.com/distribution/#macos



Installing Anaconda on Mac OS X

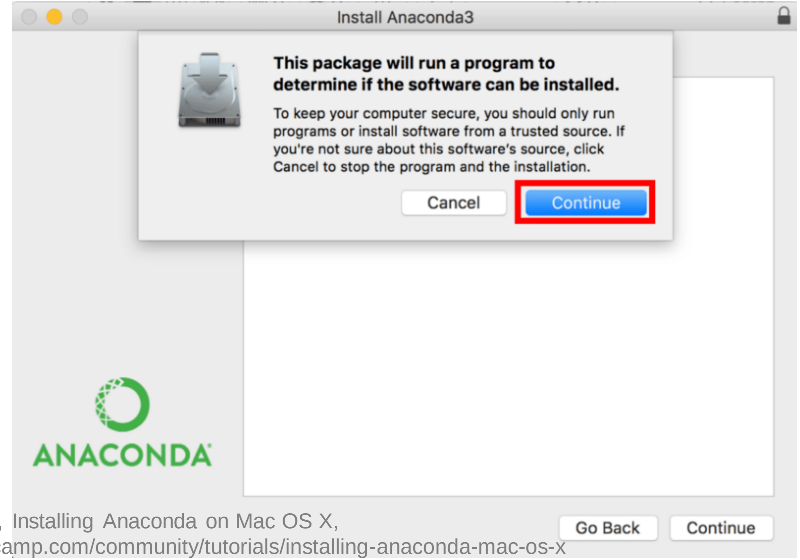
2 - Locate your download and double click it.



Michael Galarnyk, Installing Anaconda on Mac OS X, https://www.datacamp.com/community/tutorials/installing-anaconda-mac-os-x

Installing Anaconda on Mac OS X

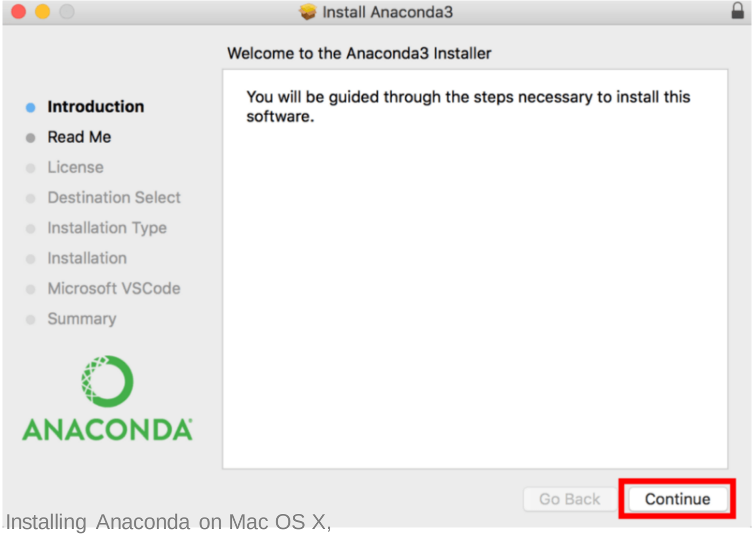
3 - Click on Continue



Michael Galarnyk, Installing Anaconda on Mac OS X, https://www.datacamp.com/community/tutorials/installing-anaconda-mac-os-x

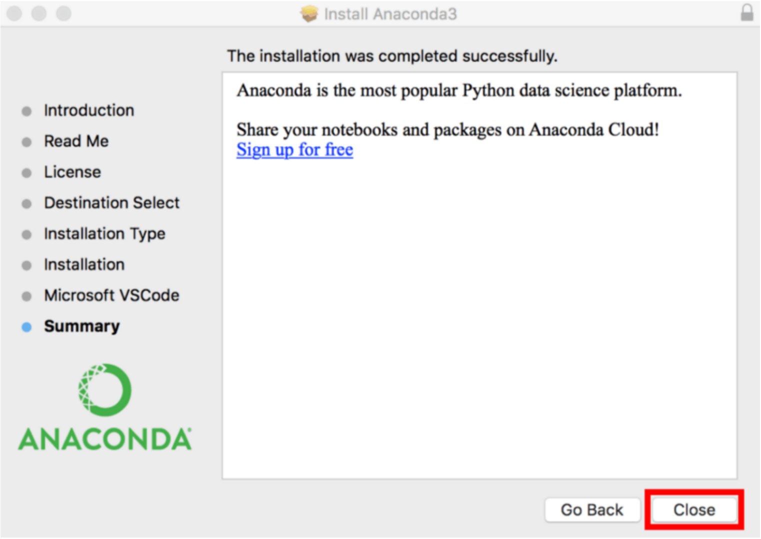
Installing Anaconda on Mac OS X

3 - Click on Continue



Michael Galarnyk, Installing Anaconda on Mac OS X, https://www.datacamp.com/community/tutorials/installing-anaconda-mac-os-x

Installing Anaconda on Mac OS X



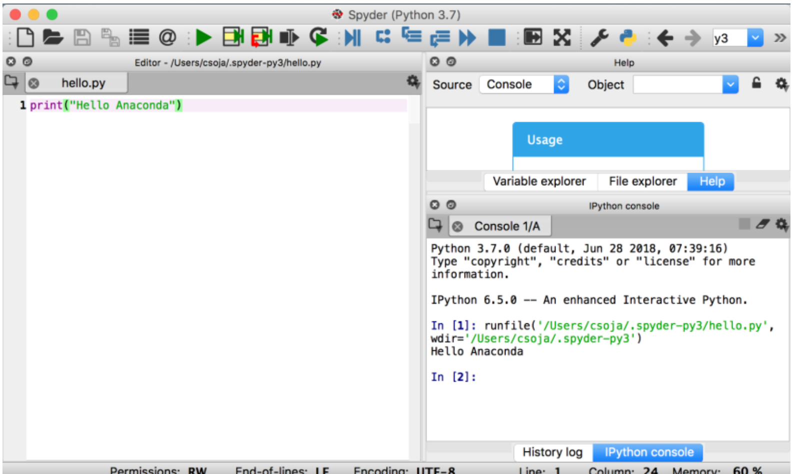
Michael Galarnyk, Installing Anaconda on Mac OS X, https://www.datacamp.com/community/tutorials/installing-anaconda-mac-os-x

Open Navigator

Michael Galarnyk, Installing Anaconda on Mac OS X,
https://www.datacamp.com/community/tutorials/installing-anaconda-mac-os-x

Run Python in Spyder IDE

- Launch Spyder by clicking Spyder’s Launch button.



Anaconda, Getting started with Anaconda,
https://docs.anaconda.com/anaconda/user-guide/getting-started/

Run Python in Spyder IDE

- In the new file on the left, type `print("Hello Anaconda")`.
- In the top menu, click *File - Save As* and name your new program *hello.py*.
- Run your new program by clicking the green triangle Run button.
- You can see your program's output in the bottom right Console pane.

Anaconda, Getting started with Anaconda, https://docs.anaconda.com/anaconda/user-guide/getting-started/

Debugging in spyder

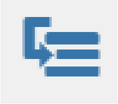
- Start debug execution (with the Debug → Debug menu option or Ctrl-F5) to activate the debugger.
- The Editor pane will then highlight the line that is about to be executed, and the Variable Explorer will display variables in the current context of the point of program execution.

Anaconda, Getting started with Anaconda, https://docs.anaconda.com/anaconda/user-guide/getting-started/

Debugging in spyder

- After entering debug mode, you can execute the code line by line using the Step button of the Debug toolbar:

- You can also inspect how a particular function is working by stepping into it with the Step Into button



Anaconda, Getting started with Anaconda, https://docs.anaconda.com/anaconda/user-guide/getting-started/

Debugging in spyder

- If you prefer to inspect your program at a specific point, you need to insert a *breakpoint* by pressing *F12* on the line on which you want to stop, or double-clicking to the left of the line number.
- A red dot in this position indicates a breakpoint; it can be removed by repeating the same procedure.

Anaconda, Getting started with Anaconda, https://docs.anaconda.com/anaconda/user-guide/getting-started/

Debugging in spyder

- After entering the Debugger, you can press the Continue button 

  to stop the execution at the breakpoint.

Anaconda, Getting started with Anaconda,
https://docs.anaconda.com/anaconda/user-guide/getting-started/

Systems Biology – Exercises

Week 2:

Exercise: Paradigm of Nature and Bio-inspired computing