

Computational Evolutionary Biology

Class Meeting

Lectures:

Mondays and Wednesdays 12:30-1:45, Dirac Science Library Room 156

Lab session:

Wednesdays 2:00-3:00, Dirac Science Library Room 156

Instructor

Peter Beerli
150-T DSL
beerli@scs.fsu.edu
(850) 645-1324

Assistant

Michal Palczewski
150-J DSL
michal@scs.fsu.edu
(206) 930-5082

Office Hours

Monday after class or by appointment.

Objectives

This course will introduce students to methods used in phylogenetics and population genetics and writing computer programs using such methods. Primary objectives of the course are:

1. to expose students to a large set of modern methods used in the field of theoretical evolutionary biology
2. introduce students to the programming aspects of the field
3. enable students to develop software that is a base for further exploration or incorporation of new methods.

Content

Advanced computational methods are becoming increasingly important in biology. A wide range of applications — including, for instance, identifying pathogens, tracing viral transmission pathways, and reconstructing the geographic expansion of humans out of Africa — rely on evolutionary inference. This course will cover the methods currently used for evolutionary inference, the stochastic models and inference principles they are based on, and how they are implemented in practice. In a separate lab session, the students will get hands-on experience in developing computational software implementing these methods. We expect that the students leave the course with the necessary skills to develop their own ideas and methods.

Textbook

Felsenstein, Joseph. 2004. *Inferring Phylogenies*. Sinauer, Sunderland MA (we did not preorder the books through the bookstore, we recommend ordering it through amazon.com, much faster than ordering directly from Sinauer)

The textbook includes material for almost all of the major topics that we expect to cover. There is a small amount of material that will be presented in class that is not covered by the textbook. This textbook is a great starting point for further exploration of the subject and we certainly will not have time to explore all chapters.

Grading

- Grades will be based on students' execution of the 5 programming assignments, each of which involves understanding the algorithms, code design, and program documentation (Points for working code and beauty of code (OOD principles, clarity in comments) [5 points each])
- Each student will do a project on their own during the last 5 weeks of the semester and also give a short presentation of their work in the last regular lab-meeting. [20+10 points]
- We will have a theory test on October 19th (midterm) and a final test in finals week. [10 points, each]

A student who accumulates 90% or more of the possible points will receive a grade of "A", a student who accumulates between 80% and 89% of the possible points will receive a grade of "B", a student who accumulates between 70% and 79% of the possible points will receive a grade of "C", a student who accumulates between 60% and 69% of the possible points will receive a grade of "D", and a student who accumulates less than 60% of the possible points will receive a grade of "F".

Attendance

Attendance in class is strongly recommended. While there is no penalty for failure to attend, the instructor may not be able to review with you any material that you miss by nonattendance unless you have a valid reason for absence (illness, death of a family member, professional conflict, subpoena to court, varsity athletics, religious holiday) or make a prior arrangement.

Missed/Late Assignments

Deadlines for assignments will be announced in class; late assignments will be accepted for full grade only in cases of illness or death in the family. 50% of the points are deducted for late assignments.

Florida State Honor Code

You are expected to know and adhere to the FSU Honor Code.

ADA Policy

If you need alternative accommodations for any reason, please notify the instructor.

Lectures

1. Introduction. Trees and tree representation (Aug. 27; Peter)
2. Parsimony. Counting evolutionary change (Aug 29; Peter)
 - Fitch parsimony
 - Wagner parsimony
 - Other parsimony variants
 - Sankoff cost matrices
3. Searching for the best tree(s) (Sept. 5; Michal)
 - How many trees are there
 - Explicit enumeration
 - Branch and bound
 - Heuristic methods
4. Stochastic models of evolution I (Sept. 10; Michal)
 - Nucleotide models: JC, F81, HKY, GTR
 - Rate variation across sites: Gamma
5. Stochastic models of evolution II (Sept. 12; Fredrik Ronquist?)
 - Protein models
 - Codon models
 - Doublet models
6. Stochastic models of evolution III (Sept. 17; Peter)
 - Restriction sites
 - Microsats
 - AFLP
 - Time-irreversible models
7. Stochastic models of evolution IV (Sept. 19; Peter)
 - Quantitative characters
 - Morphology
8. Hidden Markov models (Sept. 24; Peter)
 - General introduction
 - Rate variation across sites: the autocorrelated gamma
 - Mixture models

9. Maximum likelihood inference I (Sept. 26; Peter)
 - General principles
 - Coin-tossing
 - Statistical consistency and efficiency
10. Maximum likelihood inference II (Oct. 1; Peter)
 - Conditional likelihoods
 - Calculating likelihoods of evolutionary models
11. Bayesian inference (Oct. 3; Peter)
 - General principles
 - Coin tossing
 - Statistical consistency and efficiency
 - Priors
12. Markov chain Monte Carlo I (Oct. 8; Peter)
 - Gibbs and Metropolis samplers
 - Calculating Hastings ratios
13. Markov chain Monte Carlo II (Oct. 10; Peter)
 - Convergence and mixing
 - Metropolis-coupling
 - Splice-sampling
14. Review (Oct. 15; Peter and students)
15. Mid-term exam (Oct. 17; students)
16. Population models (Oct. 22; Michal)
17. The coalescent I (Oct.24; Michal)
 - The simple n-coalescent
 - Relation to other population genetic models
18. Trees and tree models I (Oct. 29; Michal)
 - Molecular clocks
 - Relaxed clock models
19. Trees and tree models II (Oct. 31; Michal)
 - Birth-death process

- Covarion models
20. The coalescent II (Nov. 5; Peter)
 - The structured coalescent
 - Coalescent and selection and other complications
 21. Implementing the coalescent (Nov. 7; Peter)
 - ML implementations
 - Bayesian implementations
 22. Phylogeography (Nov. 12; Peter)
 - Haplotype networks
 - Nested clade analysis
 23. Historical biogeography (Nov. 14; Peter)
 - General overview of problems
 - Parsimony reconstruction of vicariance scenarios
 - Parsimony study of dispersal events
 - Statistical approaches
 24. Gene trees and species trees; hosts and parasites (Nov. 19; Peter)
 - Overview of the field
 - Parsimony methods
 - Statistical methods
 25. Model selection and model averaging I (Nov. 21; Peter)
 - Hierarchical and nonhierarchical models
 - Hierarchical likelihood ratio test
 - Akaike information criterion
 - Bayes information criterion
 26. Model selection and model averaging II (Nov. 26; Peter)
 - The Bayesian approach: Bayes factors
 - Model averaging
 27. Bootstrapping and jackknifing (Nov. 28; Peter)
 - Nonparametric and parametric approaches to confidence
 - Phylogenetic implementations
 - Bootstrap corrections

- Bootstraps and posterior probabilities
28. Statistical multiple sequence alignment (Dec. 3; Peter)
 - Heuristic dynamic programming approaches
 - Bayesian MCMC sampling of multiple alignments
 29. Review (Dec. 5; Peter) and student presentations.
 30. Final exam (Tuesday Dec. 11, 10-12, Dirac 156 ; students)

Lab-sessions

Assignments need to be turned in as an electronic attachment, I will get you feedback that I received it, without feedback the next day, our spam or virus filter might have killed and you need to talk to me. I prefer that the assignments are compacted using either the tar.gz or the .zip format. Each assignment must contain one folder named "yourlastnameAssignmentnumber" (for example beerli1), the folder contains

- Source code (preferably in a directory *src*)
- "Executable" (a java jar file, or a compiled binary for other languages)
- Example data set (sometimes we supply an example, but you can use your own)
- User-documentation (about one page: what is the program doing, how do I run the example, what is the output)

Assignments

- Introduction to Java and NetBeans.
- Assignment 1: due September 11, 11:59pm
Parsing and printing trees (newick-format)
- Assignment 2: due September 25, 11:59pm
Calculating the parsimony score of a tree and display a histogram of scores for all trees (exhaustive search).
- Assignment 3: due October 9, 11:59pm
Calculate the likelihood of a tree using a simple mutation model.
- Assignment 4: due October 23, 11:59pm
Generate a coalescent tree and simulate data on it.
- Assignment 5: due November 6, 11:59pm
ABC sampler to estimate population size using the coalescent tree simulation parts and the likelihood calculation parts.

- Project: due December 4, 11:59pm
Write a program to solve a problem involving phylogenetic or population genetic data.
- Oral presentation of individual project (Dec. 5; students)