

HMGP 7620, CPBS 7792, MBGN 7620

# Computational Genomics Laboratory Syllabus

Section Director: Jason de Koning  
Assistant Director: Jianbin Wang

Version 0.7

January 15, 2010

## 1 Meeting time and place

Wednesdays 1-3pm  
Ed-II North, Room 2201AB (computer lab)

## 2 Instructor contact information

Jason de Koning, Ph.D.  
Research Complex 1 South, Room #10114  
Email: *Jason.DeKoning@UCDenver.edu*  
Office hours: By appointment

Jianbin Wang, Ph.D.  
Research Complex 1 South, Room #10401G  
Email: *Jianbin.Wang@UCDenver.edu*  
Office hours: By appointment

## 3 Teaching assistants

Alex Poole, M.S.  
Email: *Alex.Poole@UCDenver.edu*

Vijetha Vemulapalli, M.S.  
Email: *Vijetha.Vemulapalli.edu*

## 4 Course description

The practice of genomics research is highly dependent on computational tools and techniques, from the initial generation and storage of data, to its processing, analysis, and ultimate understanding. This laboratory course is designed to give students with little programming or Linux/UNIX proficiency enough background and hands-on experience to be able to: write simple scripts to obtain, reformat, and process large genomic data sets (*data preparation*), perform basic *data manipulation* including sequence alignment, mapping, assembly, and homology-based search, and to extract meaningful information from patterns of intra- and inter-specific genomic variation using phylogenetic methods, simulation, and advanced statistical techniques (*data analysis*). In later sections, an emphasis will be placed on evolutionary methods that properly account for the non-independence of sequence data due to phylogenetic relatedness and population structure. As this is an introductory course, we will focus more on practice than theory, although optional readings will be provided, giving inroads to the rich theoretical literature that underlies various elements of computational genomics.

## 5 Computer environment

As in most fields that rely extensively on computational science, we will be working in a UNIX/Linux environment for all computer labs. Because the available computers are setup to run Windows XP, each user will run a Linux-based “virtual machine” (VM) through Windows. Each user will be responsible for booting up and shutting down their virtual machine at the beginning and end of each class. Because failure to properly shut down your VM may result in significant problems for the next session, **demerit points will be assigned for failing to shutdown your VM properly.**

## 6 Organization and philosophy

Each class will begin with a short lesson on programming in Perl, which is a very flexible, interpreted scripting language. An emphasis on building proficiency in Perl will be made throughout, since it provides an indispensable mechanism for routine manipulation of genomic data and for automating basic tasks. The Perl lesson will be followed by an outline for an exercise that you will be responsible for turning in at the next session for grading. Some exercises will be designed to be finished during class time, while others will require some take-home work. The exercises and programming assignments are designed to be challenging to someone with little programming or computer science background. You are highly recommended to do extra reading or practice exercises in Perl if you're having trouble keeping up. If you find yourself struggling keep up, talk to the TAs and organizers with feedback on pacing as frequently as possible.

## 7 Preliminary schedule

Subject to minor revision, the schedule will be as follows. Please note suggested pre-requisite reading assignments.

### **Week 1 (February 3): Introduction to UNIX and Perl**

- Lab introduction and welcome
- Working at the "command line"
- UNIX/Linux basic commands
- Choosing a text editor
- Installing and running software from source-code "tar-balls"

Perl lesson: Hello world! Variables. Iteration with for loops.

Assignment: Environment setup. Reading assignment (BLAST).

### **Week 2 (February 10): Genome browsers and BLAST**

- Quick intro to flavors of BLAST (homology search)
- Genome browser paradigms; UCSC browser, EnsEMBL
- Functional-data tracks (UCSC)
- Comparative sequence information and orthology (EnsEMBL)
- Data extraction exercises

Perl lesson: Conditional statements with if..else. File input/output (I/O) pt. 1

Assignment: Complete three of the provided data extraction worksheets. BLAST YFG against genome browser, collect information.

### **Week 3 (February 17): Dealing with Next-Gen sequence data**

- Data file formats; quality score encoding
- Mapping short reads versus assembly tasks
- Effect of read length on ability to assemble
- Generate short read libraries with simulator script (provided); assemble with Velvet, and record observations on success under different experimental designs

Perl lesson: File input/output (I/O) pt. 2; Search and replace with regular expressions pt. 1

Assignment: Write a Perl script to reformat Illumina/Solexa data files to FASTQ format. Assemble simulated datasets into contigs using Velvet; collect statistics.

**Week 4 (February 24): Genome browser advanced topics: programmatic extraction, visualization in the UCSC browser**

- The EnsEMBL Perl API
- Setting up for EnsEMBL access
- UCSC data track upload and display

Perl lesson: Review. Post-mortem on the Perl assignment from last week.  
Assignments: Data extraction exercise (following the EnsEMBL tutorials). Visualization of your data on a genome assembly (using cDNA short read data from last week). Locate introns based on cDNA to genomic DNA mapping.

**Week 5 (March 3): And now for something completely different: R (the amazing, statistical programming language)**

- Introduction to R: syntax for basic calculations, descriptive statistics
- Data input with read.table()
- Random number generation
- Subsetting data
- Visualization: xy-plots, histograms
- Outputting figures to pdf format

Perl lesson: Arrays. Subroutines / functions. Flex your muscles.  
Assignment: Data manipulation, graphing exercises, statistics based on data re-sampling.

**Week 6 (March 10): Transcriptomics workshop**

- Gene Ontology (GO) for functional summaries
- Generate contigs for provided dataset (next-gen format)
- Use corrected hit-counts as measure of expected abundance
- Compare across tissues: GO-term abundance

Perl lesson: Hashes (associative arrays)  
Assignment: Count the number of times a given oligo is observed in a file (Perl). Which GO categories are enriched in a treatment versus control sample?

**Week 7 (March 17): March break**

- Recommended reading: Review Perl notes; Perl supplemental readings.

**Week 8 (March 24): Transcriptomics analysis with R and BioConductor (guest lecture)**

- Tzu Phang, Ph.D. Center for Computational Pharmacology (UC Denver)

**Week 9 (March 31): Phylogenetics I: Phylogenetic inference**

- PHYLIP
- MrBayes
- bootstrap analysis versus posterior credibility
- consensus trees

Assignment: Resolve history of gene duplications for a multi-gene family. Obtain support values for a given bipartition; speculate on differences between bootstrap and posterior support.

**Week 12 (April 7): microRNA Bioinformatics**

- TBD
- Assignment: TBD

**Week 10 (April 14): Phylogenetics II: Analysis and hypothesis testing**

- hypothesis tests for tree topologies: KH and SH tests
- likelihood ratio testing on molecular evolutionary mechanism: rate variation
- phylogenetic shadowing / footprinting for detecting conserved, functional regions

Assignment: Phylogenetic footprinting/shadowing using likelihood ratio statistics.

**Week 11 (April 21): Population genetics and genomics I**

- descriptive statistics on population structure
- estimation of ancestral population sizes and demography
- Structure program (Pritchard)
- Nielsen and Beerli software

Assignment: TBD

**Week 13 (April 28): Metagenomics workshop**

- TBD

Assignment: TBD

**Week 14 (May 5): Population genetics and genomics II**

- 1,000 genomes project data retrieval
- MCMC and likelihood analysis
- rare variants

Assignment: Analysis of 1,000 genomes data.

**Week 15 (May 12): Functional inference from coding data**

- $d_N/d_S$  analysis using codon substitution models
- inference of selection; PAML, PIDAMEbL
- comparative vertebrate genomics of coding regions

Assignment: Detect selection using branch-site type methods. Explain functional significance of positively-selected residues.

## 8 Course readings

Several free, online resources for Perl and UNIX are provided below. Several books are also being evaluated for use as a course text, and one will be chosen in the next few weeks. Some good candidates are listed below.

### 8.1 Concise online resources

**Learn Perl Fast:** Short PERL tutorial. *Highly recommended.*

**Perl Reference:** Beginning PERL

**Learn UNIX Fast:** Introduction to UNIX and Linux

**UNIX Reference:** UNIX quick reference sheet

### 8.2 Candidate texts

**Learning Perl for bioinformatics:** Beginning Perl for Bioinformatics. (Tisdall, 2001). O'Reilly Media.

### 8.3 Optional resource books

**Basic UNIX and computer background for bioinformatics:** Developing Bioinformatics Computer Skills (Gibas and Jambeck, 2001). O'Reilly Media.

**Sequence analysis:** Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. (Durbin, Eddy, Krogh, and Mitchison, 1998)

## 9 Grading

Assignments will vary in difficulty and time-required to complete. Students will have the remaining class time to work on assignments each week, and will have until the beginning of the following week's class to turn in assignments. Students are required to submit assignment solutions by e-mail before the beginning of the next week's class. Late submissions will not be accepted unless arrangements have been made with Jason or Jianbin.

## 10 Policy on Collaboration

Students are expected to work together to solve problems as necessary, but each student is required to submit their own unique solution for assignments. Programming assignments, in particular, should be done independently so that everyone builds proficiency in the assigned areas.