Genetic and genomic analyses using RAD-seq and Stacks

Referenced-aligned RAD tags for genome scanning and identifying signatures of selection

Instructors:

Julian Catchen <jcatchen@illinois.edu>
Department of Animal Biology, University of Illinois at Urbana-Champaign
William Cresko <wcresko@uoregon.edu>
Institute of Ecology and Evolution, University of Oregon

Datasets and Software

- **Data sets - All are produced using an Illumina HiSeq 2500 sequencer**
  - ***Dataset 6 (DS6)*** - This is a set of population genomic data from the threespine stickleback. The dataset comprises 8 individuals from each of two differentiated populations, for a total of 16 barcoded individuals. The RAD data were prepared using the restriction enzyme *SbfI*, and sequenced using an Illumina sequencer. These data are unpublished, but similar to those published in Hohenlohe, et al. 2010.

- **Software - All are open source software**
  - ***Stacks* (http://catchenlab.life.illinois.edu/stacks/*)*** - A set of interconnected open source programs designed initially for the *de novo* assembly of RAD sequences into loci for genetic maps, and extended to be used more flexibly in studies of organisms with and without a reference genome. The pipeline has a Perl wrapper allowing sets of programs to be run. However, the software is modular, allowing it to be applied to many scenarios. You will use the Perl wrapper in class and the modules on your own.
  - ***GSnap* (http://research-pub.gene.com/gmap/)** - *GSnap* is a very fast and efficient software package used for aligning sequences against a reference genome. We will use *GSnap* to align RAD reads against the stickleback reference genome, and then analyze these reads within the *Stacks* pipeline. Although we will use *GSnap* for this exercise, many other algorithms and software exist for aligning against a reference genome, and these could be used in conjunction with *Stacks* as well.
  - ***Samtools* (http://samtools.sourceforge.net)** - A suite of software tools designed to perform a variety of common tasks with next generation sequencing data tools. The SAM and BAM were developed associated.

Exercise III. Population genomics with a reference genome

1. Population genetics is a very old field that has a rich mathematical theory and a core set of statistical approaches for inferring parameters from genetic data. These statistics are such things as nucleotide diversity (π), differentiation statistics (i.e. $\mathbf{F_{st}}$), and measures of genetic covariance such as Linkage Disequilibrium (**D** and **D′**). However, because of methodological limitations, the majority of the theoretical, statistical and empirical work in population genetics has focused on a small number of loci. With the advent of second generation sequencing, tens or hundreds of thousands of genetic markers can now be examined in dozens of individuals, allowing the field of population genomics to truly come to fruition. An exciting new activity in population genomics is the identification of signatures of selection in wild populations. Today you will process RAD data from one oceanic and one freshwater population of threespine stickleback from Middleton Island, which is located off the coast of Alaska. One set of data comes from an ancestral oceanic population, whereas the other is from a derived freshwater population that is likely less than 60 years old. We will align these data to the stickleback reference genome using *GSnap*, and then feed the alignments into *Stacks*. After *Stacks* determines the loci and associated alleles present in each population, we will export the data and calculate several population genomic statistics, including F_{ST}. Performing a study like this was nearly impossible before the advent of next generation sequencing. *For more information on population genomics, see the papers listed in that section at the end of this document.*

2. Acquire and process DS3 (Middleton Island).

   - In your `./working` workspace, create a directory called `scan` to contain all the data for this exercise. Inside that directory, create three directories: `samples`, `aligned`, and `stacks`. To save time, we have already cleaned and demultiplexed this data set and will start from the cleaned samples stage.

   - Unarchive DS3 from

     ```
     ~/workshop_data/stacks/scan/middleton_scan.tar
     ```

     into the `samples` directory.

3. Align the stickleback sequences against the genome with *GSnap*.

   - Run *GSnap* on the first freshwater sample: `samples/s13_fw_01.fa.gz`

     [Running GSnap could take 30-60 minutes total.]

     - Running *GSnap* with the `--help` parameter will give you a list of all options.

     - We only want to keep alignments that have a single, best alignment to the genome. With the right combination of parameters, *GSnap* can do this natively.

       Some hints for command line parameters: set *GSnap* to read gzipped input files; allow a maximum of five mismatches in the alignment; set an indel penalty of two; allow only one path (a.k.a. alignment) for each read and if there is more than one path, suppress that read; make sure you turn on multithreading and output

SAM format. Name the output file the same as the input file, with a ".sam" suffix instead of ".fa.gz".

    - The stickleback *GSnap* database is located within this directory:

      ```
      ~/workshop_data/stacks/gsnap_db
      ```

      the GSnap database is stored in several files, we will specify only the common prefix of those files and GSnap will know how many files to read in.

    - Use Samtools to convert the SAM output file to a BAM file. Delete the SAM file.
        - What is a terminal alignment and why is it important for RAD data?
        - Why do we convert the SAM file to a BAM file, what is the advantage?
        - Why was our `middleton_scan.tar` file not also gzipped?

- Run *GSnap* again with the first oceanic sample: `samples/s13_an_01.fa.gz` and convert it again to a BAM file using Samtools.
- To save time, the remaining 14 alignments can be found here:

  ```
  ~/workshop_data/stacks/scan/scan_gsnap_aligned.tar
  ```

  `Untar` these remaining *GSnap* alignments into the `aligned` directory.

4. Create a new MySQL database called `middleton_radtags` and populate the tables by loading the table definitions from:

```
/usr/local/share/stacks/sql/stacks.sql
```

If you view this file, you will see all the SQL commands necessary to create tables to hold our Stacks data. We need to create the database and then feed these commands to the MySQL server:

```
% mysql --defaults-file=/usr/local/share/stacks/sql/mysql.cnf
        -e "CREATE DATABASE middleton_radtags"

% mysql --defaults-file=/usr/local/share/stacks/sql/mysql.cnf
        middleton_radtags < /usr/local/share/stacks/sql/stacks.sql
```

5. We next want to run *Stacks* on the freshwater and anadromous population.

- Run the *Stacks* `ref_map.pl` pipeline program. This program will run `pstacks`, `cstacks`, and `sstacks` on the members of the population, accounting for the alignments of each read.
    - Information on `ref_map.pl` and its parameters can be found online:
        - http://catchenlab.life.illnois.edu/stacks/comp/ref_map.php
    - Create a file in the working directory called `popmap` that is formatted like this:

      ```
      <sample file prefix><tab><population ID>
      ```

      Include all 16 samples in this file and specify which individuals belong to which populations and specify it to the `ref_map.pl` program. A sensible set of population IDs might be "fw" for the freshwater samples, and "oc" for the oceanic (anadromous) samples.

    - Specify each *GSnap*-aligned individual as a "sample" to `ref_map.pl`. Specify the `stacks` directory as the output location.
- Once *Stacks* has completed running, investigate the output files. Notice that each locus now has a chromosome/base pair specified in each of the `*tags.tsv` files and in the catalog files.
- Examine the Stacks output through the web interface:
    - http://<Amazon Instance>/stacks/
    - Explore the web interface
        - Why are some markers found in more samples?
        - Set the filters so that there are no fewer than 2 SNPs and no more than 3 SNPs per locus and so that there are at least 12 matching individuals per locus.
        - Select a locus that has a reasonable ratio of genotypes (depending on your parameter choices you may have slightly different loci compared with another run of the pipeline). Click on `Allele Depths` to view additional information.
        - Select a polymorphic sample, click on the alleles to see the actual stack that corresponds to the catalog locus.
            - Do any of the columns have a blue background? If so, why?
            - Why do some nucleotides in the stack have a yellow background?
            - What are the different roles played by primary and secondary reads?

6. The program `populations` calculates population genetic statistics for each SNP in the two populations for one level of population subdivision, as we have here. So, it will calculate expected and observed heterozygosity, π, F_{IS}, and it includes F_{ST} as a measure of genetic differentiation between populations. It uses the same method for calculating F_{ST} as was used in the human HapMap project.

- Now look at the output in the file `batch_1.sumstats.tsv`. There are a large number of statistics calculated at each SNP, such as the frequency of the major allele (*P*), and the observed and expected heterozygosity, and F_{IS}. Use UNIX commands like `head`, `cat`, `cut`, `more`, `column`, and `sort` to focus on the minimum and maximum heterozygosity and F_{IS} statistics. Are these statistics in the (roughly) same locations within the genome between the two populations? How are these summary statistics related to Hardy-Weinberg equilibrium?
- Pick a locus from the `batch_1.sumstats.tsv` file and look it up in the web interface. Do the data match?
- What are the population mean and standard error for π in the two populations? (Check the `batch_1.sumstats_summary.tsv` file.)

7. Because RAD produces so many genetic markers, and because we have a reference genome sequence, we can examine population genetic statistics like F_{ST} as continuous distributions along the genome. The populations program does this using a kernel-smoothing sliding window approach.

  - Run the `populations` program again, this time turning on kernel smoothing for F_{ST}. Also, turn on filters so we only include loci available in both populations that are found in 75% of individuals of each population, and use a p-value correction to exclude insignificant F_{ST} measures. Be sure to again specify the population map you created previously and turn on multithreading.

- The output file **`batch_1.fst_summary`** contains the mean F_{ST} measure between populations. What is the mean F_{ST} between the marine and freshwater populations?
- The output file **`batch_1.fst_oc-fw.tsv`** contains F_{ST}, a measure of genetic differentiation between the two populations. What is the maximum value of F_{ST} at any SNP? How many SNPs reach this F_{ST} value?
- Look at the genomic distribution of F_{ST} in the file **`batch_1.fst_oc-fw.tsv`**. Use UNIX commands like **`cut`** and **`sort`** to find the genomic regions that show the highest levels of population differentiation.
  - What does the p-value generated by Fisher's exact test tell you about a particular F_{ST} score? How about the LOD score?
- Now plot F_{ST} over a single linkage group. First use **`grep`** to produce a new F_{ST} file with only data for Linkage Group IV (labeled **groupIV**), call it **`batch_1.fst_oc-fw_lg4.tsv`**. Now plot this file using gnuplot.
- Copy the Gnuplot script to the `stacks` directory:

```
~/workshop_data/stacks/scan/fst_groupIV.gnuplot
```

- Cat the file to see what it does.
- Execute Gnuplot:

```
% gnuplot < fst_groupIV.gnuplot
```

- Download the resulting PDF file and open it. The red crosses represent the raw F_{ST} measures while the green line is the kernel-smoothed average value.