pyRAD - a python pipeline for Restriction-Associated-DNA sequencing

Andrew Gottscho, PhD

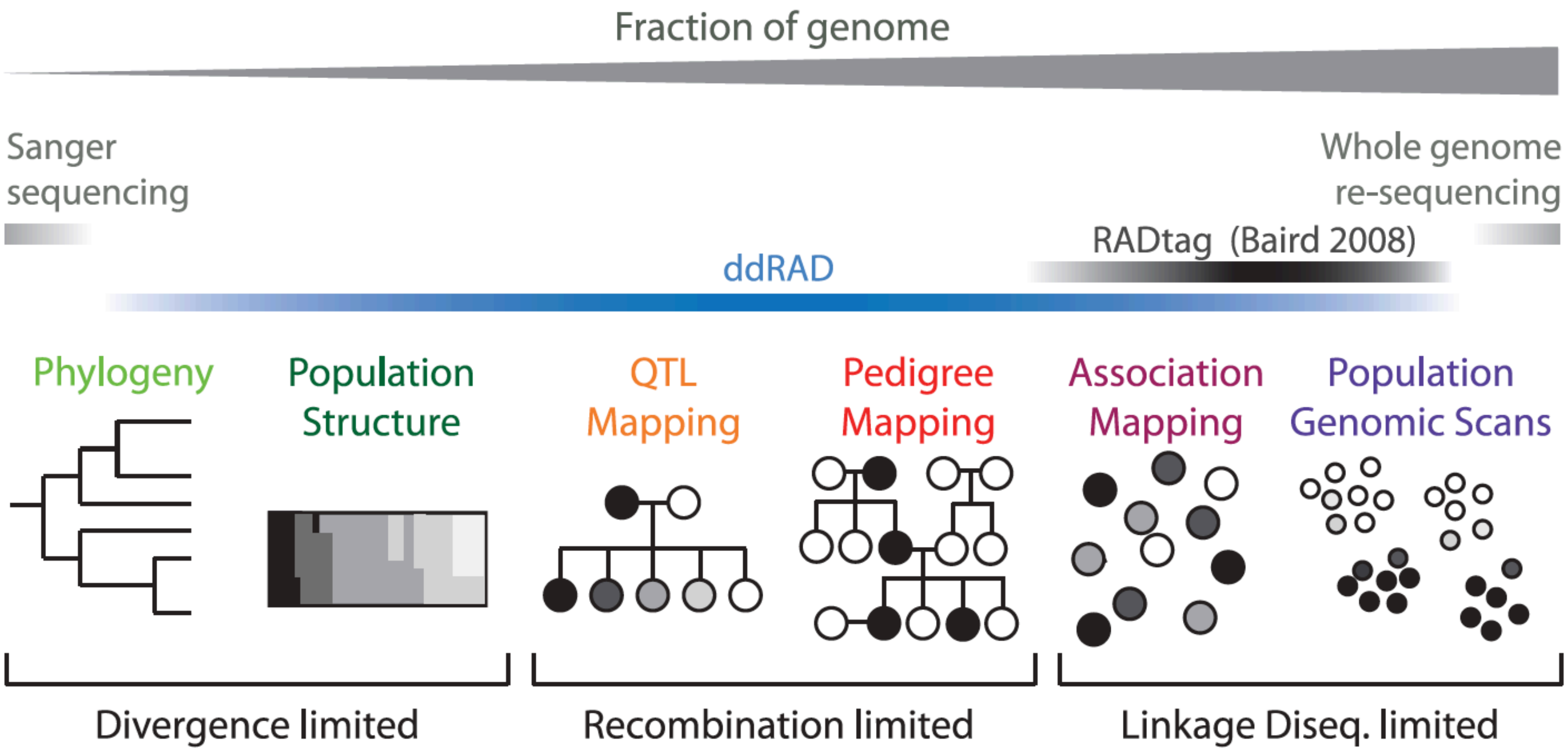
Peter Buck Postdoctoral Fellow

Vertebrate Zoology, Amphibian & Reptile Division

National Museum of Natural History

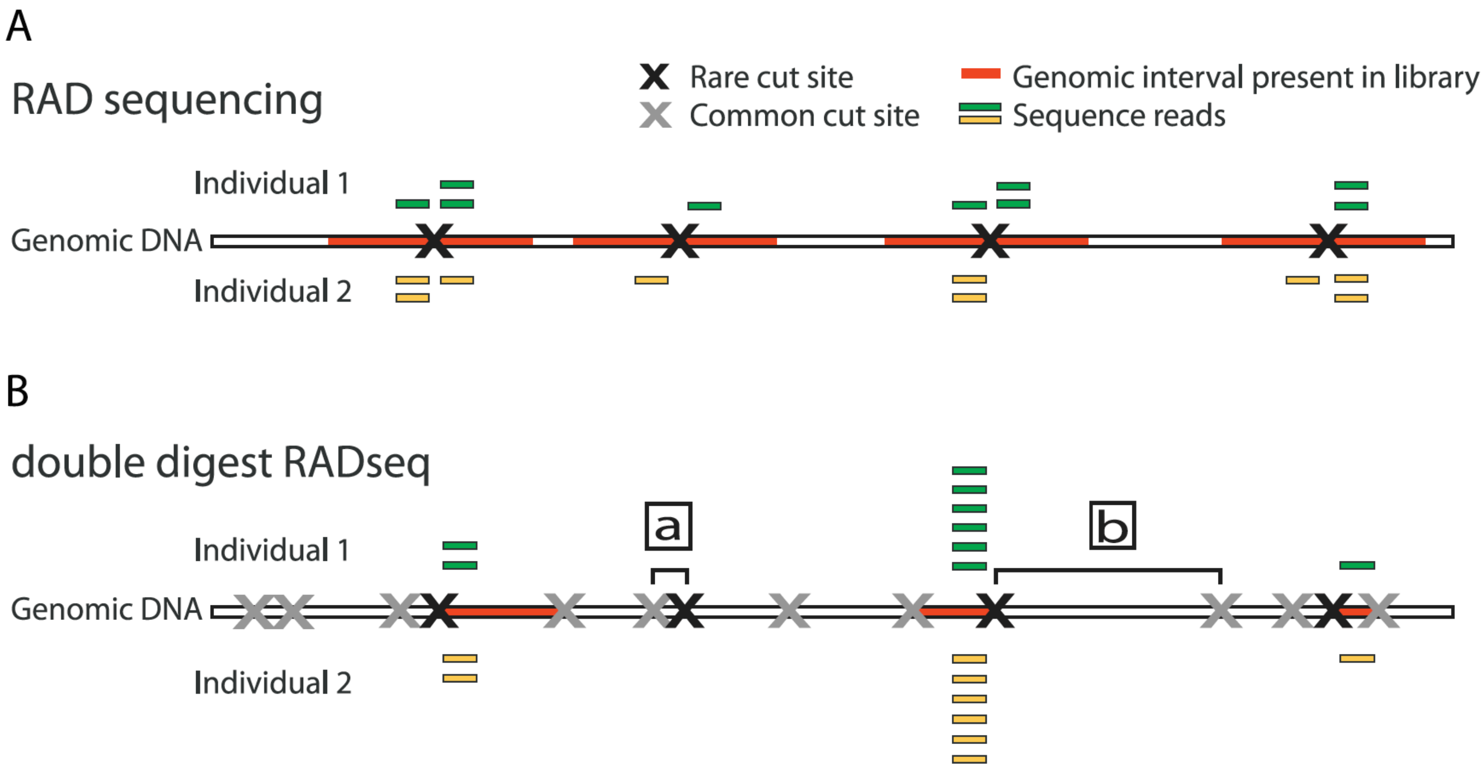
March 17, 2016

Restriction-Associated DNA sequencing (RADseq)



Peterson et al. (2012)

Restriction-Associated DNA sequencing (RADseq)



Peterson, B.K., Weber, J.N., Kay, E.H., Fisher, H.S. & Hoekstra, H.E. (2012) Double digest RADseq: an inexpensive method for de novo SNP discovery and genotyping in model and non-model species. PLoS One, 7, e37135.

pyRAD

- Two popular pipelines for assembling RADseq data: pyRAD and STACKS
- Advantages of pyRAD:
  - Aligns sequences with MUSCLE and USEARCH (or VSEARCH), accommodates indel variation
  - Hierarchical clustering for large datasets
  - D-statistics for gene flow
  - Python-based, open source
  - Supports RAD, ddRAD, GBS, paired-ddRAD, paired-GBS
- http://dereneaton.com/software/pyrad/
- https://groups.google.com/forum/#!forum/pyrad-users
- Eaton, D.A.R. (2014) PyRAD: assembly of de novo RADseq loci for phylogenetic analyses. Bioinformatics (Oxford), 30, 1844-1849

Installation and Dependencies

- Must have python 2.7 or greater (?) installed on laptop or cluster
- Must have numpy and scipy packages installed
  - http://www.numpy.org/
  - http://www.scipy.org
- MUSCLE
  - http://www.drive5.com/muscle/
- USEARCH or VSEARCH
  - http://www.drive5.com/usearch/
  - https://github.com/torognes/vsearch

pyRAD: 7 steps to pipeline

1. De-multiplexing (separate by barcodes)
2. Quality filtering and removal of barcodes, cut sites and adapters
3. Clustering within samples and alignment
4. Joint estimation of error rate and heterozygosity
5. Consensus base calling and paralog detection
6. Clustering across samples
7. Alignment across samples, filtering and formatting

Step 1: De-multiplexing (separate by barcodes)

- Separates raw FASTQ formatted sequence data into separate files for each sample
- Allows a user set maximum number of mismatches (errors) in a barcode
- If samples are already demultiplexed, this step can be skipped

Step 2: Quality filtering

- Removes barcodes and Illumina adapters, if present
- Filters reads by their quality scores, replacing base calls below a user-set limit with an ambiguous base (N)
- Reads with more than a user-defined number of Ns are discarded

Step 3: Clustering within samples and alignment

- First collapses replicate sequences into individual records while retaining their total number of occurrences.
- Sequence order is randomized and clustering is performed using USEARCH with all heuristic options turned off.
- This creates clusters (stacks) by matching each sequential sequence to a ’seed’ sequence that came before it, or else creating a new seed. The resulting stacks are aligned with MUSCLE.

Step 4: Joint estimation of error rate and heterozygosity

- Uses the maximum likelihood equation of Lynch (2008) to jointly estimate the mean heterozygosity and sequencing error rate from the base frequencies at each site across all stacks in an individual (with greater than a set minimum depth of coverage)

Step 5: Consensus base calling and paralog detection

- Uses values from step 4 to calculate the binomial probability a site is homozygous (aa or bb) versus heterozygous (ab) (Li et al., 2008)
- A base call is only made if the depth of coverage is above a user-set minimum, and high enough to make a statistical base call, else it is called undetermined (N)
- Consensus sequences containing more than a maximum number of undetermined sites are discarded
- To filter for paralogs (or repetitive or high copy number DNA regions, hereafter referred to collectively as 'paralogs' for simplicity), consensus sequences are also discarded if they contain more than a maximum number of heterozygous sites or more than the allowed number of haplotypes (two for diploids).

Step 6: Clustering across samples

- Putative orthologs are then identified by clustering consensus loci across samples in USEARCH, using only one allele from each consensus sequence to measure sequence similarity, but retaining data for both (or multiple) alleles

Step 6: Clustering across samples

- Putative orthologs are then identified by clustering consensus loci across samples in USEARCH, using only one allele from each consensus sequence to measure sequence similarity, but retaining data for both (or multiple) alleles

Step 7: Alignment across samples, filtering and formatting

- The resulting stacks are aligned and filtered once again for paralogs before being output in a variety of familiar formats as individual or concatenated loci (e.g. Fasta, Phylip, Nexus), or in several custom formats [e.g. Haplotypes, single nucleotide polymorphisms (SNPs) and unlinked SNPs]
- The filter applied in this step makes use of a user-set maximum for the number of shared heterozygous sites across all samples in the dataset (maxSharedH)
- For a phylogenetic scale dataset the expectation for this number is unknown, but should be fairly low under the assumption that polymorphisms are less likely to be retained over deep divergences than are fixed differences between paralogs
- Loci containing one or more heterozygous sites shared across more than 'maxSharedH' samples are thus discarded as potential paralogs
- Step 7 can be repeated while substituting different subsets of taxa, and requiring different amounts of coverage across them, to construct datasets of varying size and completeness

Lets Get Started!