Ensamblaje de representación reducida de genomas (RAD-seq, GBS y similes)

Introducción a la bioinformática e investigación reproducible para análisis genéticos

Introducción a la Bioinformática para Biólogos y Genetistas

Alicia Mastretta Yanes

Metodologías de representación reducida de genomas

(RAD, GBS, etc)

Restriction site-associated DNA sequencing

Rapid SNP Discovery and Genetic Mapping Using Sequenced RAD Markers

Nathan A. Baird, Paul D. Etter, Tressa S. Atwood, Mark C. Currey, Anthony L. Shiver, Zachary A. Lewis, Eric U. Selker, William A. Cresko, Eric A. Johnson

Published: October 13, 2008 • DOI: 10.1371/journal.pone.0003376

Proportion of the genome sequenced

Sanger sequencing

Whole genome sequencing

Reduced representation genome sequencing

non model species revolution

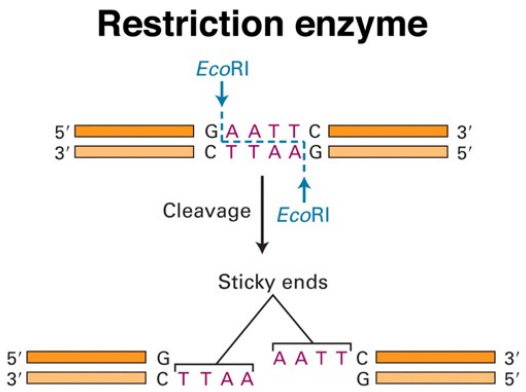
OPEN ACCESS Freely available online

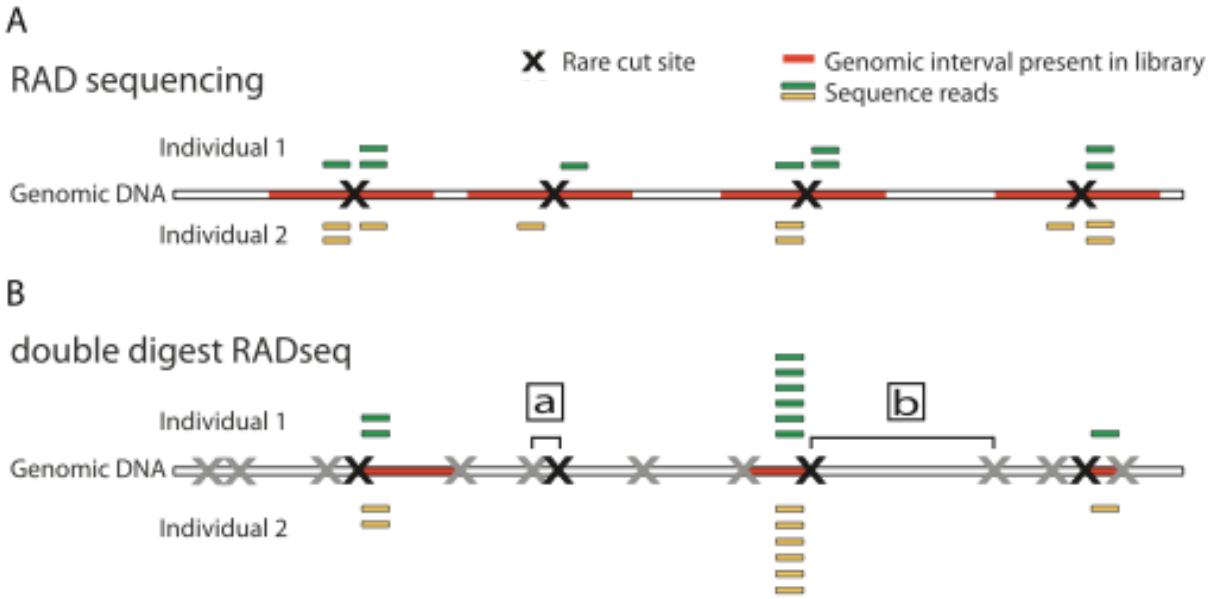
PLoS one

Rapid SNP Discovery and Genetic Mapping Using Sequenced RAD Markers

Nathan A. Baird[1☯], Paul D. Etter[1☯], Tressa S. Atwood[2], Mark C. Currey[3], Anthony L. Shiver[1], Zachary A. Lewis[1], Eric U. Selker[1], William A. Cresko[3], Eric A. Johnson[1*]

1 Institute of Molecular Biology, University of Oregon, Eugene, Oregon, United States of America, **2** Floragenex, Eugene, Oregon, United States of America, **3** The Center for Ecology and Evolutionary Biology, University of Oregon, Eugene, Oregon, United States of America





Peterson *et al.* 2012. PLoS ONE

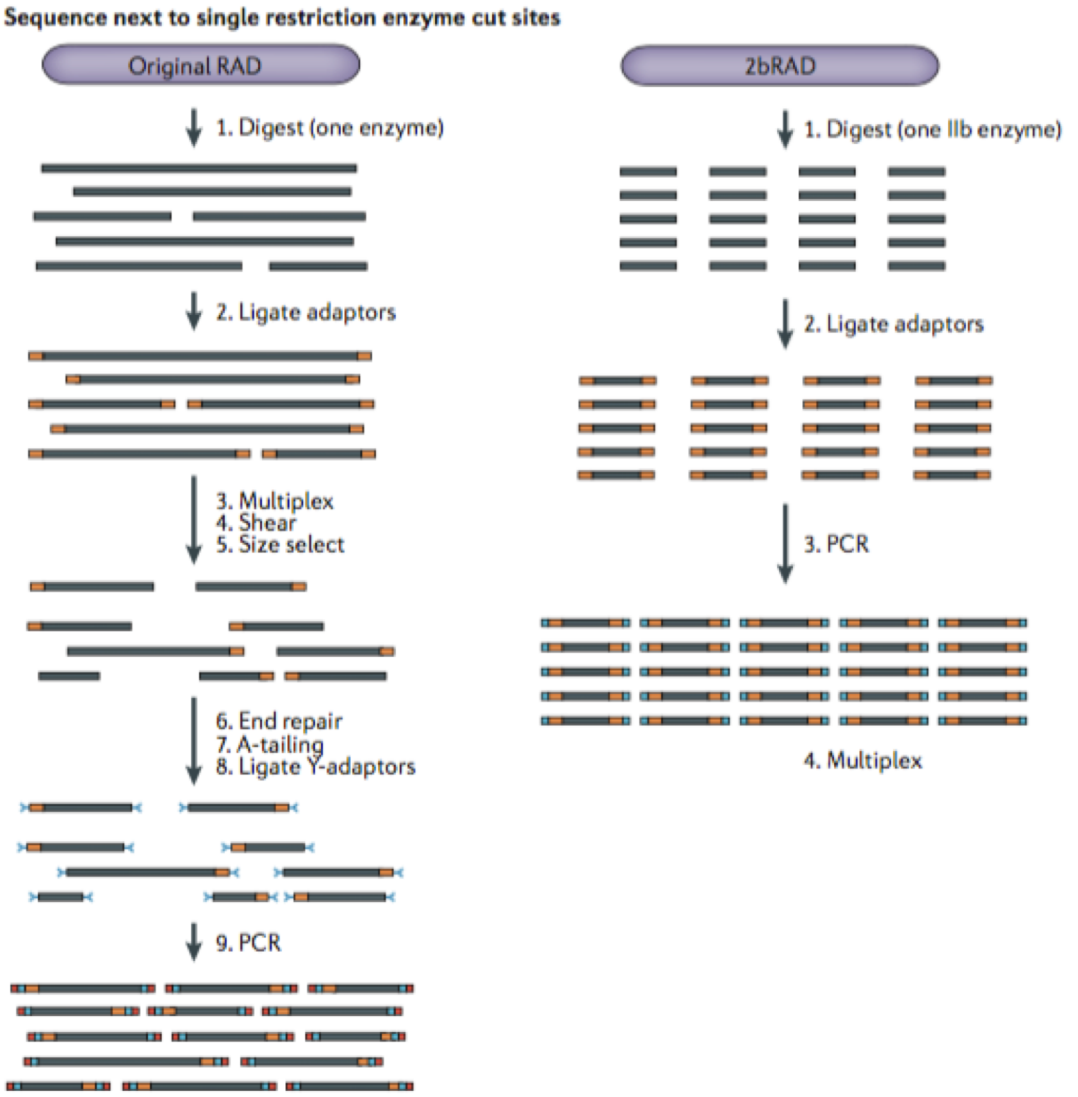
nature REVIEWS GENETICS

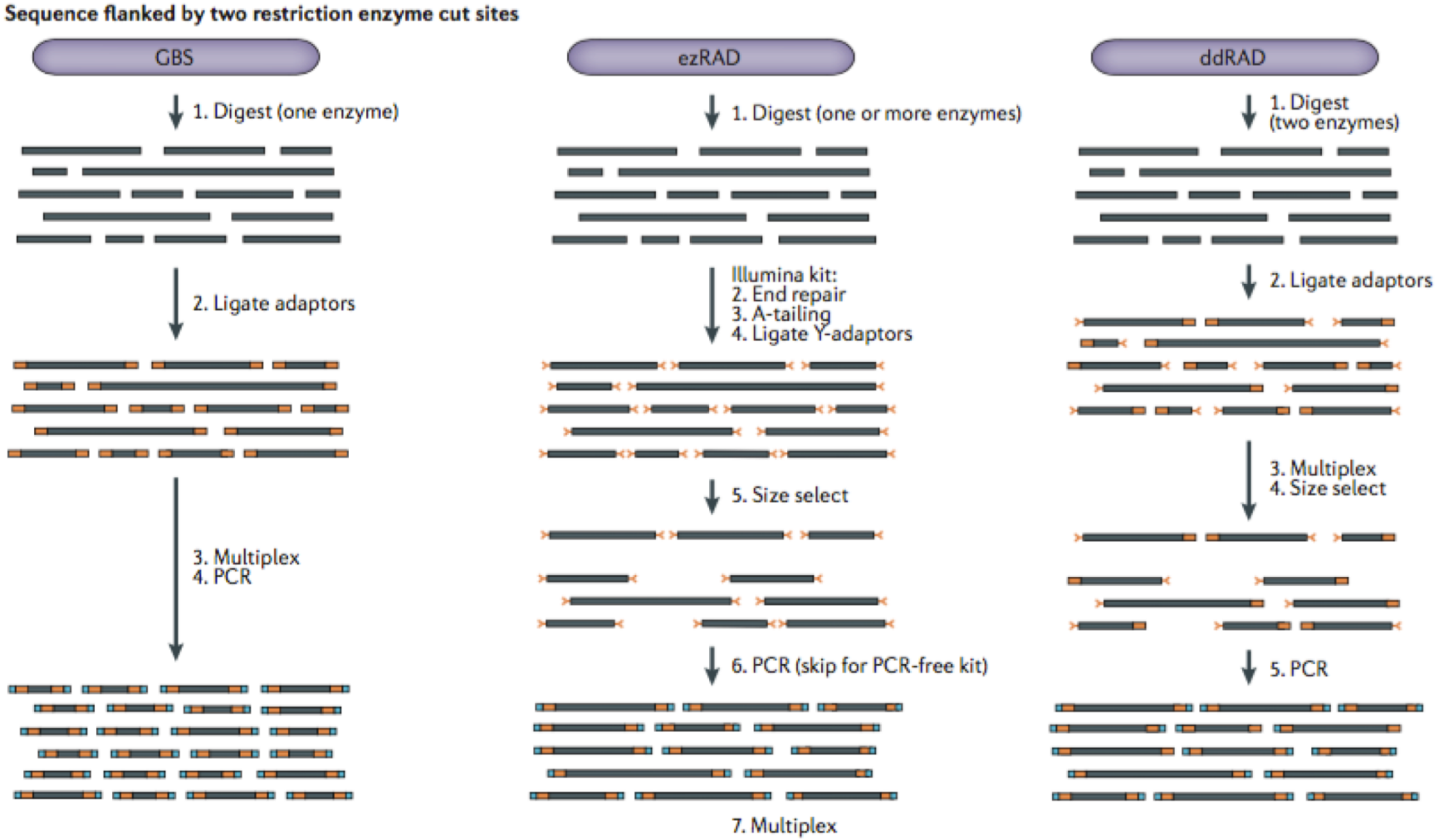
Review Article | Published: 05 January 2016

Harnessing the power of RADseq for ecological and evolutionary genomics

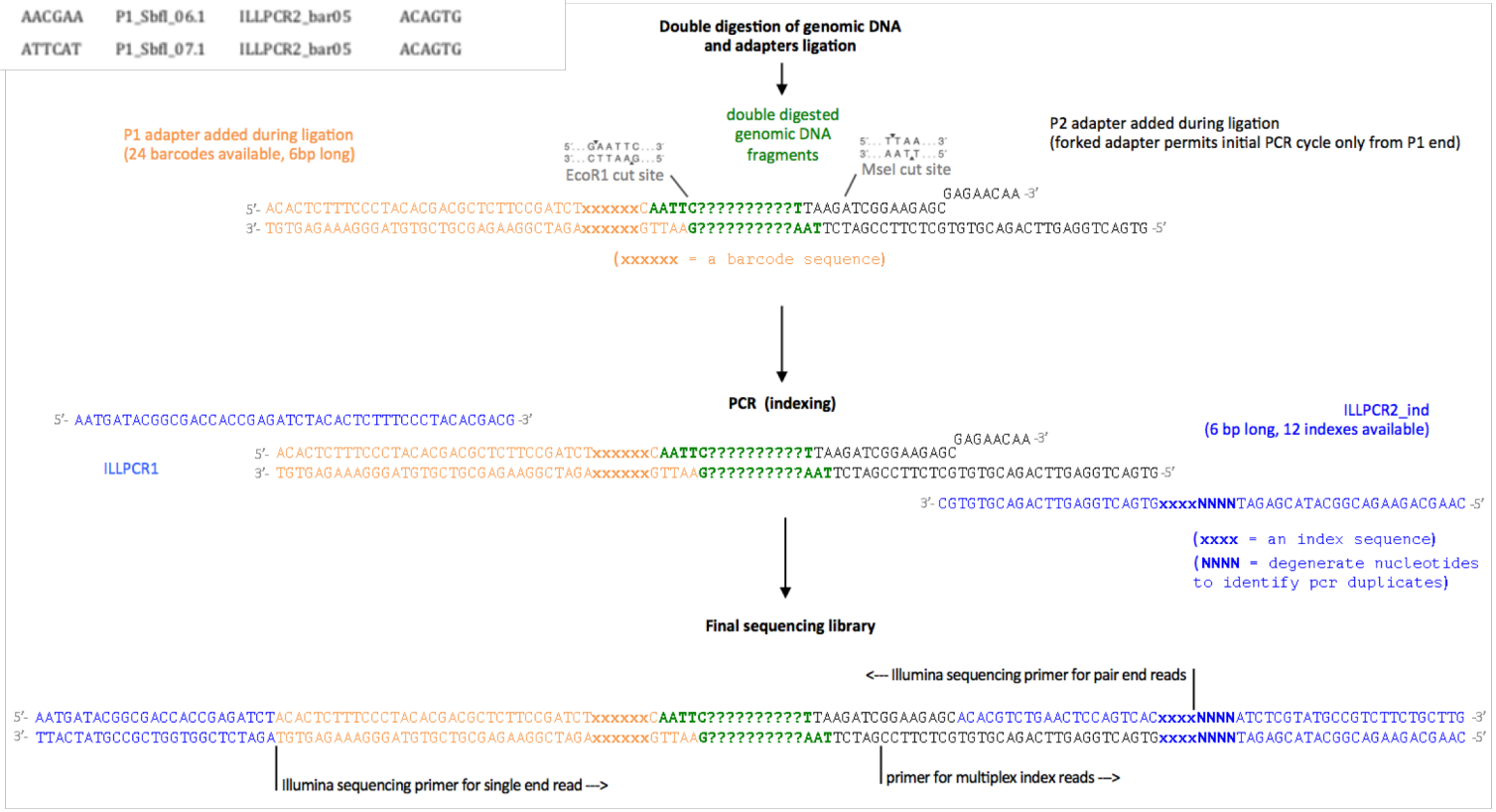
Kimberly R. Andrews, Jeffrey M. Good, Michael R. Miller, Gordon Luikart & Paul A. Hohenlohe

Nature Reviews Genetics **17**, 81–92 (2016) | Download Citation




Sequence flanked by two restriction enzyme cut sites
GBS
1. Digest (one enzyme)
2. Ligate adaptors
3. Multiplex
4. PCR
ezRAD
1. Digest (one or more enzymes)
Illumina kit:
2. End repair
3. A-tailing
4. Ligate Y-adaptors
5. Size select
6. PCR (skip for PCR-free kit)
7. Multiplex
ddRAD
1. Digest
(two enzymes)
2. Ligate adaptors
3. Multiplex
4. Size select
5. PCR

SampleSEQ.ID*	Library	Plate	well	Barcode	ID.adaptor	Index(ILLPCR2)	Index_sequence
JbYc02_ir	Ju01	A	A1	GGTCTT	P1_SbfI_01.1	ILLPCR2_bar05	ACAGTG
JbBk22	Ju01	A	B1	CTGGTT	P1_SbfI_02.1	ILLPCR2_bar05	ACAGTG
JbMh12	Ju01	A	C1	AAGATA	P1_SbfI_03.1	ILLPCR2_bar05	ACAGTG
JmPpJ16	Ju01	A	D1	ACTTCC	P1_SbfI_04.1	ILLPCR2_bar05	ACAGTG
JmTaJ01	Ju01	A	E1	TTACGG	P1_SbfI_05.1	ILLPCR2_bar05	ACAGTG
JbPr13	Ju01	A	F1	AACGAA	P1_SbfI_06.1	ILLPCR2_bar05	ACAGTG
NegCtrL01	Ju01	A	G1	ATTCAT	P1_SbfI_07.1	ILLPCR2_bar05	ACAGTG

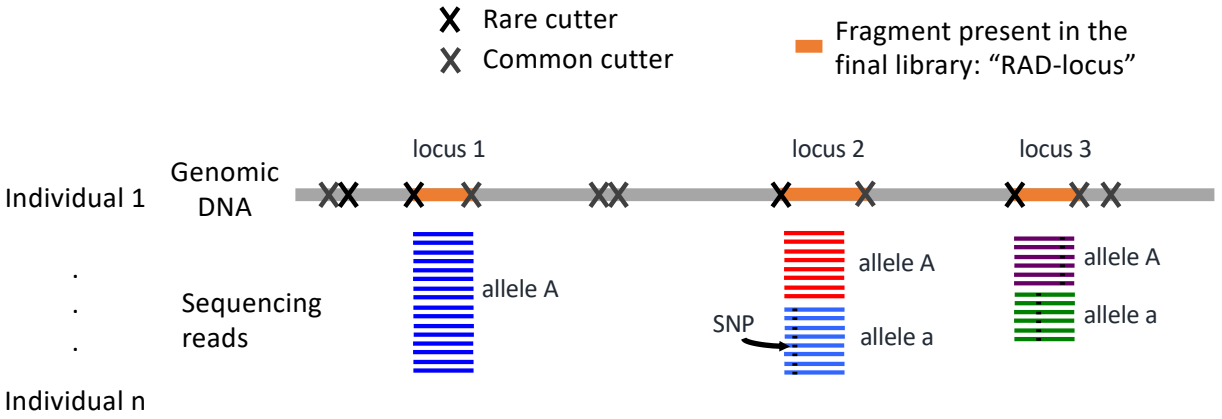


Mastretta-Yanes et al (2015) *Mol. Ecol. Res.*

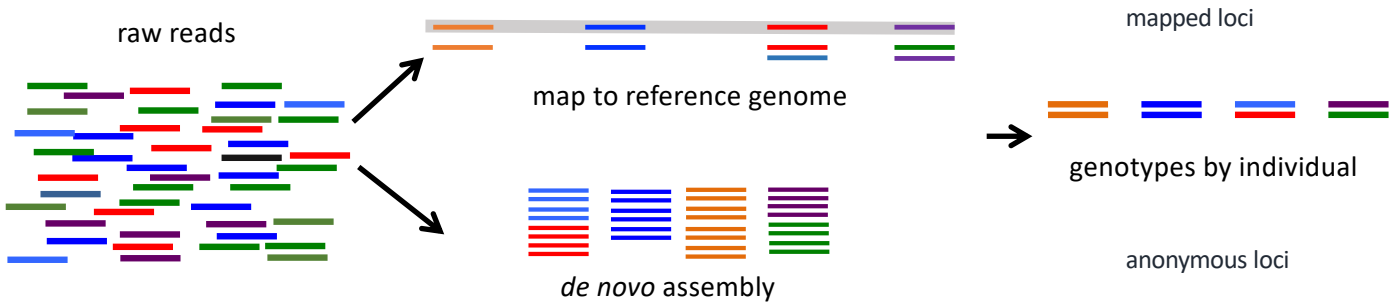
Ensamblaje *de novo* vs. sobre una referencia

From lab to genotypes

ddRAD library preparation



Bioinformatics



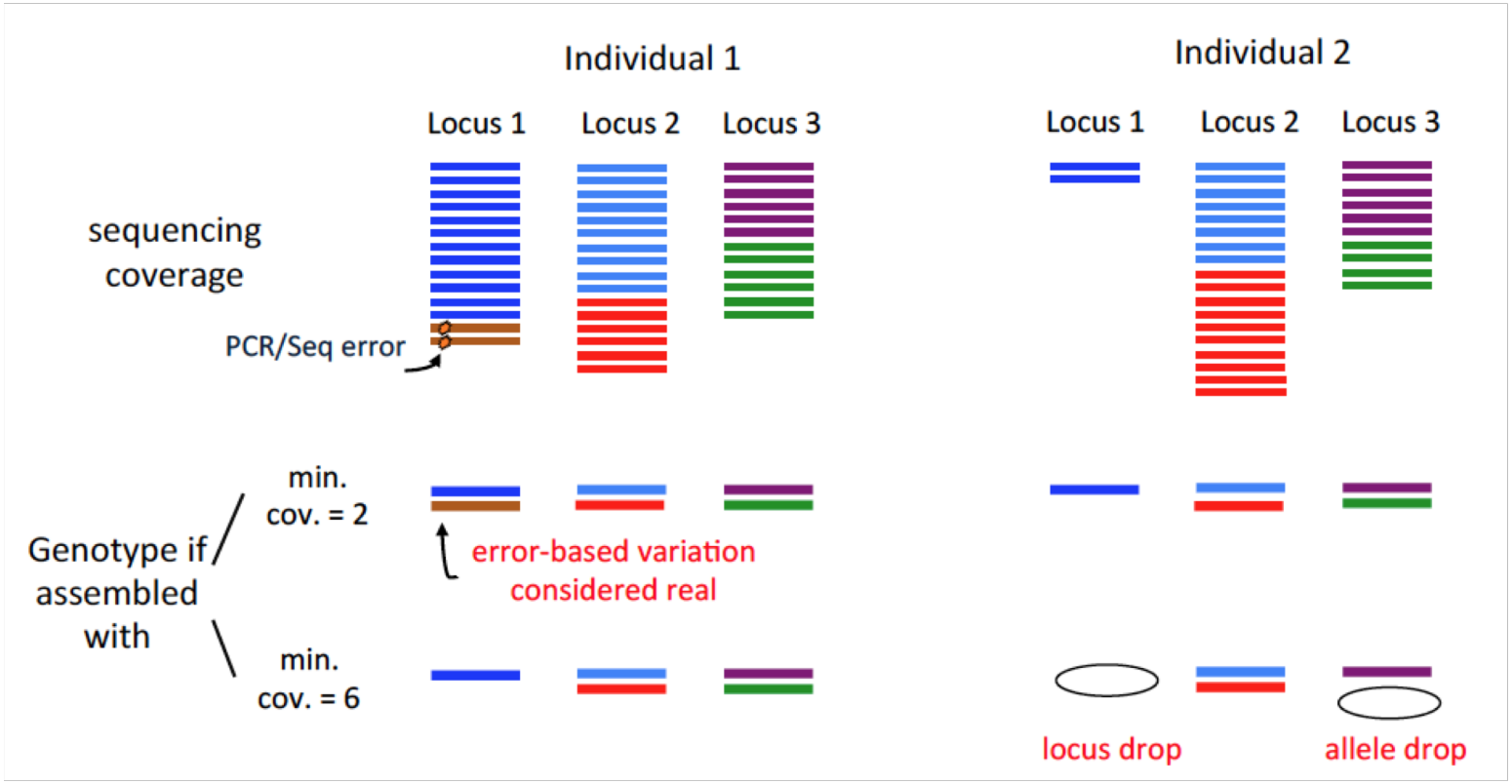
de novo assembly parameters

- Several parameters to consider
- They differ among software
- Optimun parameter values depend on the data set (taxa, read quality, etc).
- Most sensitive parameters:
  - minimun coverage
  - level of sequence dissimilarity

In general terms, these parameters should be relaxed enough to account for genetic variation and sequencing errors, but strict enough to discriminate between paralogous loci.

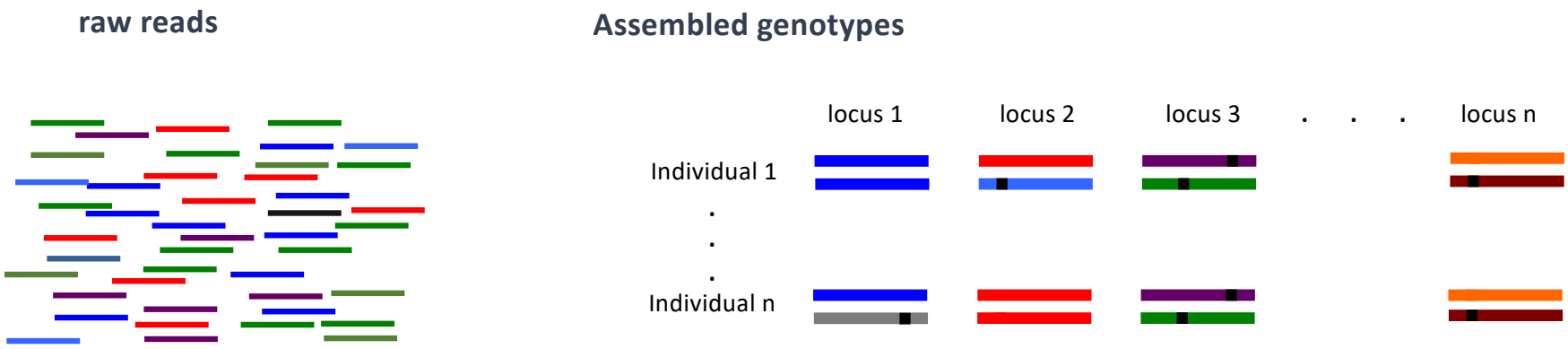
Rochette & Catchen et al (2017) Nature Protocols

The role of min coverage allowed during *de novo* assembly

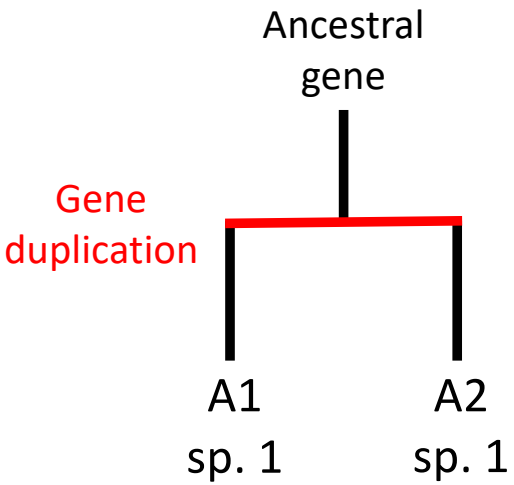


Sequence similarity & paralogous loci during *de novo* assembly

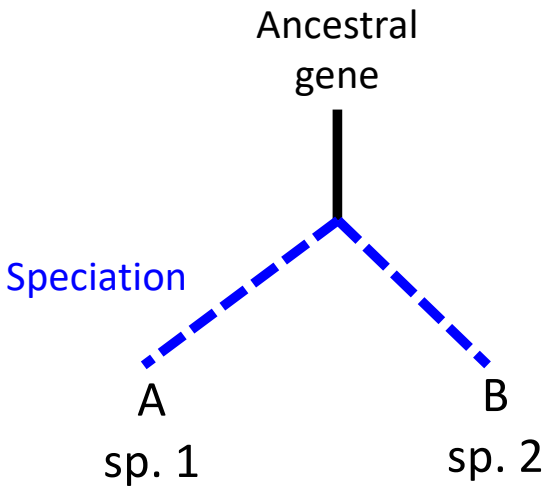
reads for the same locus must be regrouped across alleles and across samples based on sequence similarity.



Paralogy



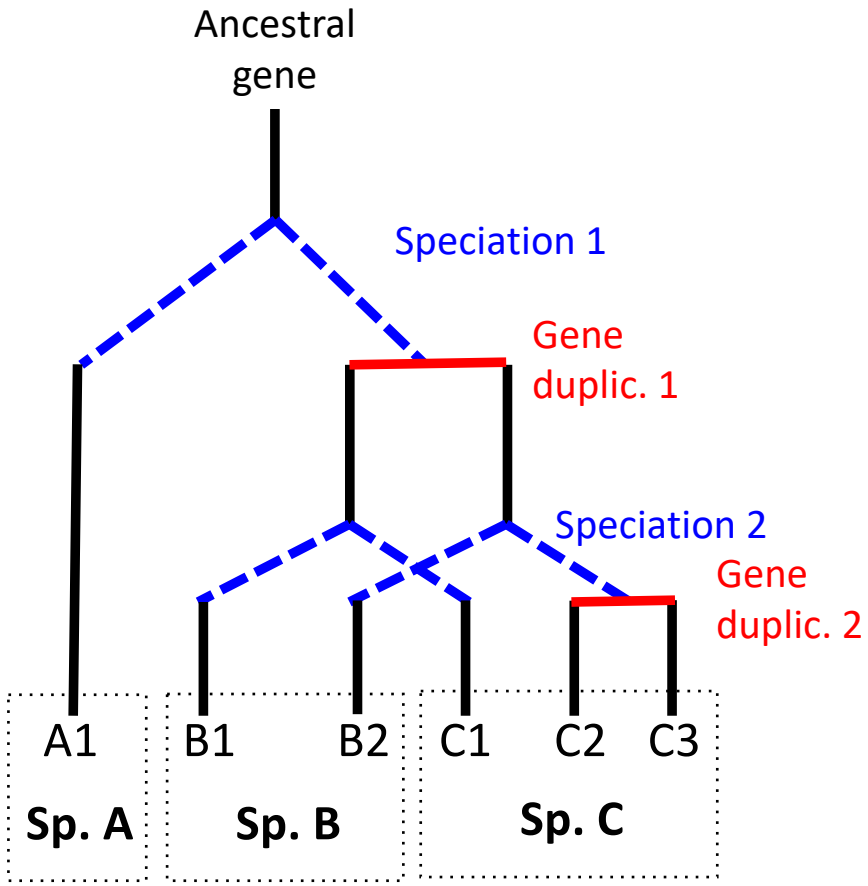
paralogous loci



orthologous loci

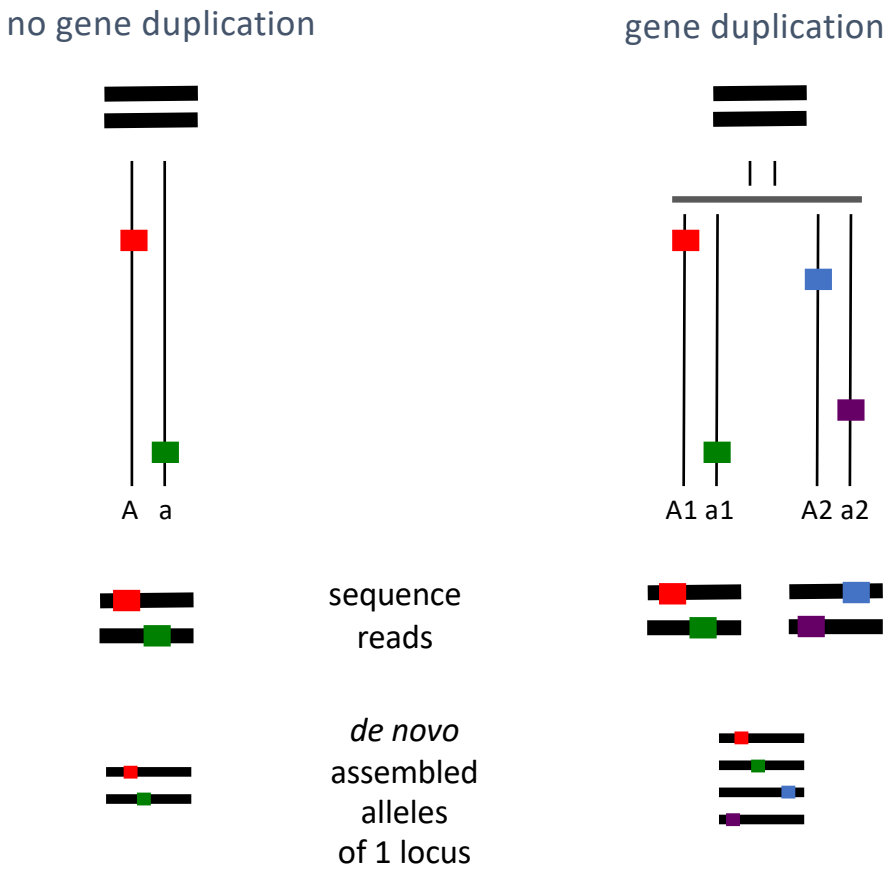
- Whole genome
- Chromosome segments
- Single genes

Paralogy



get rid of paralogous loci for pop. genetics analyses

The problem of paralogy and *de novo* assembly



Principales algoritmos y software (Stacks, pyRAD, Tassel)

	Stacks Catchen et al 2013	ipyRAD Eaton et al 2014	Tassel Bradbury et al 2007
Better for	Population genetics non-model spp.	Phylogenies non-model spp.	Population genetics GBS of crops
Type of variation	SNPs, indels	SNPs, indels	SNPs, indels
Method	*de novo* reference genome	*de novo*, reference, reference addition and reference subtraction	*de novo* (not supported anymore) reference genome
Wet lab Protocols	RAD-seq, ddRAD, DaRT, 2bRAD and GBS	RAD, ddRAD, GBS, NextRAD, RApture	GBS
Pair-end?	Yes	Yes	No in de novo
output	**VCF, plink**, Stacks custom, structure, **GenPop**, phylip, PHASE, Beagle	**VCF**, ipyrad custom, **phylip**, geno, structure, **nexus**,	VCF, plink, h5
Comments	Estimates Fst, π, He	Jupyter notebooks	Graphical interface useful to “visualize” SNPs

Stacks

- Webpage
- Manual
- Main parameters

Pipeline components

The Stacks pipeline is designed modularly to perform several different types of analyses. Programs listed under *Raw Reads* are used to clean and filter raw sequence data. Programs under *Core* represent the main Stacks pipeline — building loci (ustacks), creating a catalog of loci (cstacks, and matching samples back against the catalog (sstacks), transposing the data (tsv2bam), adding paired-end reads to the analysis and calling genotypes, and population genomics analysis. Programs under *Execution Control* will run the whole pipeline.

Raw Reads

```
process_radtags
process_shortreads
clone_filter
kmer_filter
```

Core

```
ustacks
cstacks
sstacks
tsv2bam
gstacks
populations
```

Execution control

```
denovo_map.pl
ref_map.pl
```

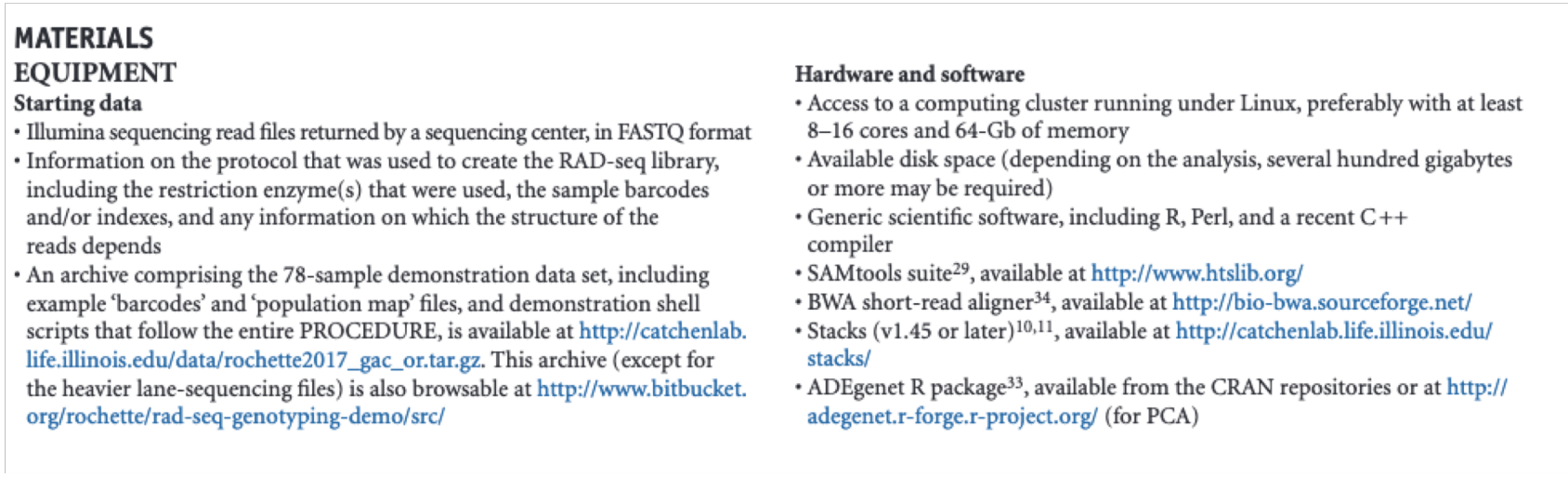
Example pipeline

```
#!/bin/bash

src=$HOME/research/project

files="sample_01
sample_02
sample_03"

#
# Build loci de novo in each sample for the single-end reads only. If paired-end reads are available,
# they will be integrated in a later stage (tsv2bam stage).
# This loop will run ustacks on each sample, e.g.
#   ustacks -f ./samples/sample_01.1.fq.gz -o ./stacks -i 1 --name sample_01 -M 4 --gapped -p 8
#
id=1
for sample in $files
do
    ustacks -f $src/samples/${sample}.1.fq.gz -o $src/stacks -i $id --name $sample -M 4 --gapped -p 8
    let "id+=1"
done

#
# Build the catalog of loci available in the metapopulation from the samples contained
# in the population map. To build the catalog from a subset of individuals, supply
# a separate population map only containing those samples.
#
cstacks --gapped -n 6 -P $src/stacks/ -M $src/popmaps/popmap -p 8

#
# Run sstacks. Match all samples supplied in the population map against the catalog.
#
sstacks --gapped -P $src/stacks/ -M $src/popmaps/popmap -p 8

#
# Run tsv2bam to transpose the data so it is stored by locus, instead of by sample. We will include
# paired-end reads using tsv2bam. tsv2bam expects the paired read files to be in the samples
# directory and they should be named consistently with the single-end reads,
# e.g. sample_01.1.fq.gz and sample_01.2.fq.gz, which is how process_radtags will output them.
#
tsv2bam -P $src/stacks/ -M $src/popmaps/popmap --pe-reads-dir $src/samples -t 8

#
# Run gstacks: build a paired-end contig from the metapopulation data (if paired-reads provided),
# align reads per sample, call variant sites in the population, genotypes in each individual.
#
gstacks -P $src/stacks/ -M $src/popmaps/popmap -t 8

#
# Run populations. Calculate Hardy-Weinberg deviation, population statistics, f-statistics
# export several output files.
#
populations -P $src/stacks/ -M $src/popmaps/popmap -r 0.65 --vcf --genepop --structure --fstats --hwe -t 8
```

Tutorial

Rochette, N. C. & Catchen, J. M. Deriving genotypes from RAD-seq short-read data using Stacks. *Nature Protocols* **12**, 2640–2659 (2017).

Demo scripts: https://bitbucket.org/rochette/rad-seq-genotyping-demo/src/default/
Demo data: http://catchenlab.life.illinois.edu/data/rochette2017_gac_or.tar.gz



MATERIALS

EQUIPMENT

Starting data

- Illumina sequencing read files returned by a sequencing center, in FASTQ format
- Information on the protocol that was used to create the RAD-seq library, including the restriction enzyme(s) that were used, the sample barcodes and/or indexes, and any information on which the structure of the reads depends
- An archive comprising the 78-sample demonstration data set, including example 'barcodes' and 'population map' files, and demonstration shell scripts that follow the entire PROCEDURE, is available at http://catchenlab.life.illinois.edu/data/rochette2017_gac_or.tar.gz. This archive (except for the heavier lane-sequencing files) is also browsable at http://www.bitbucket.org/rochette/rad-seq-genotyping-demo/src/

Hardware and software

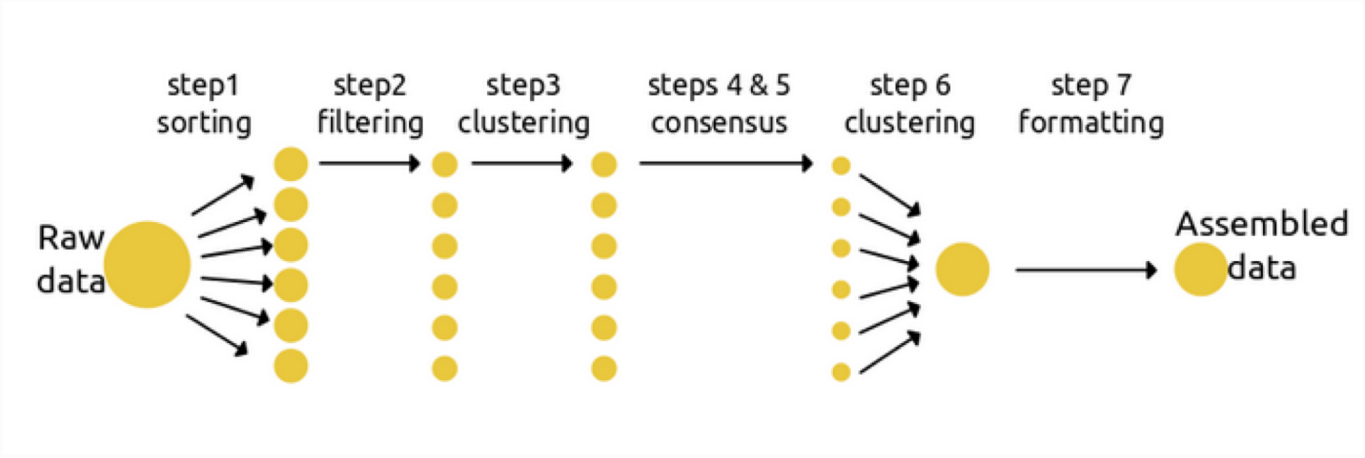
- Access to a computing cluster running under Linux, preferably with at least 8–16 cores and 64-Gb of memory
- Available disk space (depending on the analysis, several hundred gigabytes or more may be required)
- Generic scientific software, including R, Perl, and a recent C++ compiler
- SAMtools suite[29], available at http://www.htslib.org/
- BWA short-read aligner[34], available at http://bio-bwa.sourceforge.net/
- Stacks (v1.45 or later)[10,11], available at http://catchenlab.life.illinois.edu/stacks/
- ADEgenet R package[33], available from the CRAN repositories or at http://adegenet.r-forge.r-project.org/ (for PCA)

Erratum: There is a typo at step 11 in the protocol's PDF: the >> operator should be used instead of the > one. The faulty code replaces the output file at each command instead of appending to it, so that the resulting file will be identical to sample_name.rem.2.fq.gz (which is likely to contain very few reads).

ipyrad

[Webpage]
[Manual]
[Main parameters]

The simplest use of ipyrad is to assemble a data set under a single set of parameters defined in a params file. Step 1 assigns data to each of the Samples, Steps 2-5 process data for each Sample, Step 6 clusters data across Samples, and Step 7 filters these data and formats them for downstream analyses.



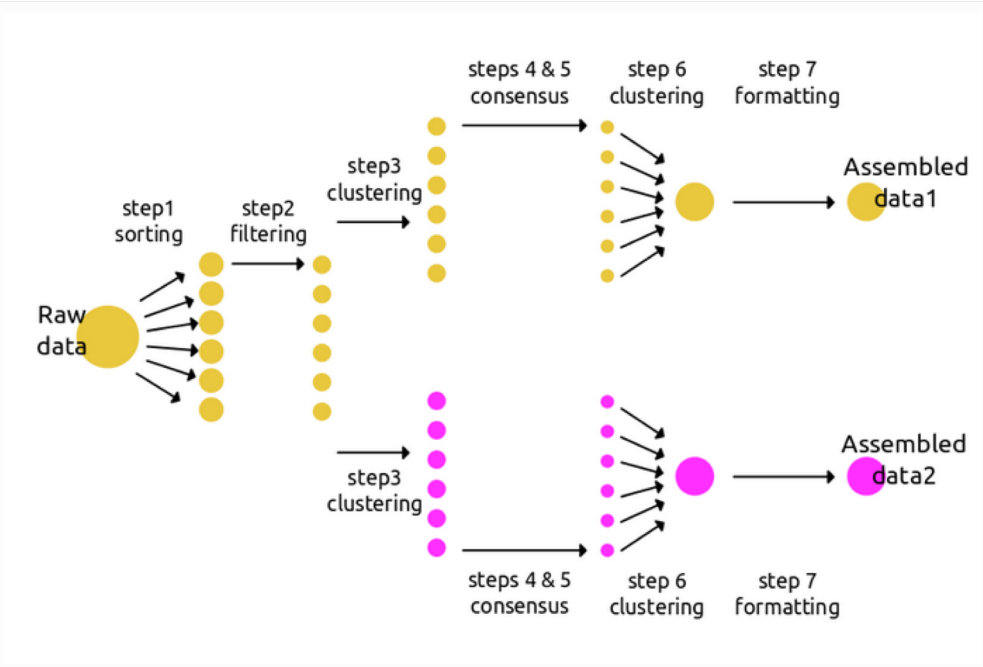
Example pipeline

```
## create an initial Assembly params file
>>> ipyrad -n data1

## fill in the new params file with your text editor
## ... editing params-data1.txt

## run steps 1-7 with the params file for this Assembly
>>> ipyrad -p params-data1.txt -s 1234567
```

```
------- ipyrad params file (v.0.5.15)-----------------------------------------
iptest                         ## [0] [assembly_name]: Assembly name. Used to name output direc
./                             ## [1] [project_dir]: Project dir (made in curdir if not present
                               ## [2] [raw_fastq_path]: Location of raw non-demultiplexed fastq
                               ## [3] [barcodes_path]: Location of barcodes file
                               ## [4] [sorted_fastq_path]: Location of demultiplexed/sorted fas
denovo                         ## [5] [assembly_method]: Assembly method (denovo, reference, de
                               ## [6] [reference_sequence]: Location of reference sequence file
rad                            ## [7] [datatype]: Datatype (see docs): rad, gbs, ddrad, etc.
TGCAG,                         ## [8] [restriction_overhang]: Restriction overhang (cut1,) or (
5                              ## [9] [max_low_qual_bases]: Max low quality base calls (Q<20) i
33                             ## [10] [phred_Qscore_offset]: phred Q score offset (33 is defau
6                              ## [11] [mindepth_statistical]: Min depth for statistical base c
6                              ## [12] [mindepth_majrule]: Min depth for majority-rule base cal
10000                          ## [13] [maxdepth]: Max cluster depth within samples
0.85                           ## [14] [clust_threshold]: Clustering threshold for de novo asse
0                              ## [15] [max_barcode_mismatch]: Max number of allowable mismatch
0                              ## [16] [filter_adapters]: Filter for adapters/primers (1 or 2=s
35                             ## [17] [filter_min_trim_len]: Min length of reads after adapter
2                              ## [18] [max_alleles_consens]: Max alleles per site in consensus
5, 5                           ## [19] [max_Ns_consens]: Max N's (uncalled bases) in consensus
8, 8                           ## [20] [max_Hs_consens]: Max Hs (heterozygotes) in consensus (H
4                              ## [21] [min_samples_locus]: Min # samples per locus for output
20, 20                         ## [22] [max_SNPs_locus]: Max # SNPs per locus (R1, R2)
8, 8                           ## [23] [max_Indels_locus]: Max # of indels per locus (R1, R2)
0.5                            ## [24] [max_shared_Hs_locus]: Max # heterozygous sites per locu
0, 0                           ## [25] [trim_reads]: Trim raw read edges (5'>, <3') applies sam
0, 0, 0, 0                     ## [26] [trim_loci]: Trim locus edges (see docs) (R1>, <R1, R2>,
p, s, v                        ## [27] [output_formats]: Output formats (see docs)
                               ## [28] [pop_assign_file]: Path to population assignment file
```

Tutorial

Introductory tutorial: https://ipyrad.readthedocs.io/tutorial_intro_cli.html#create-an-ipyrad-params-file
Advanced tutorial: https://ipyrad.readthedocs.io/tutorial_advanced_cli.html#tutorial-advanced-cli
Jupyter and more tutorials: https://ipyrad.readthedocs.io/userguide.html

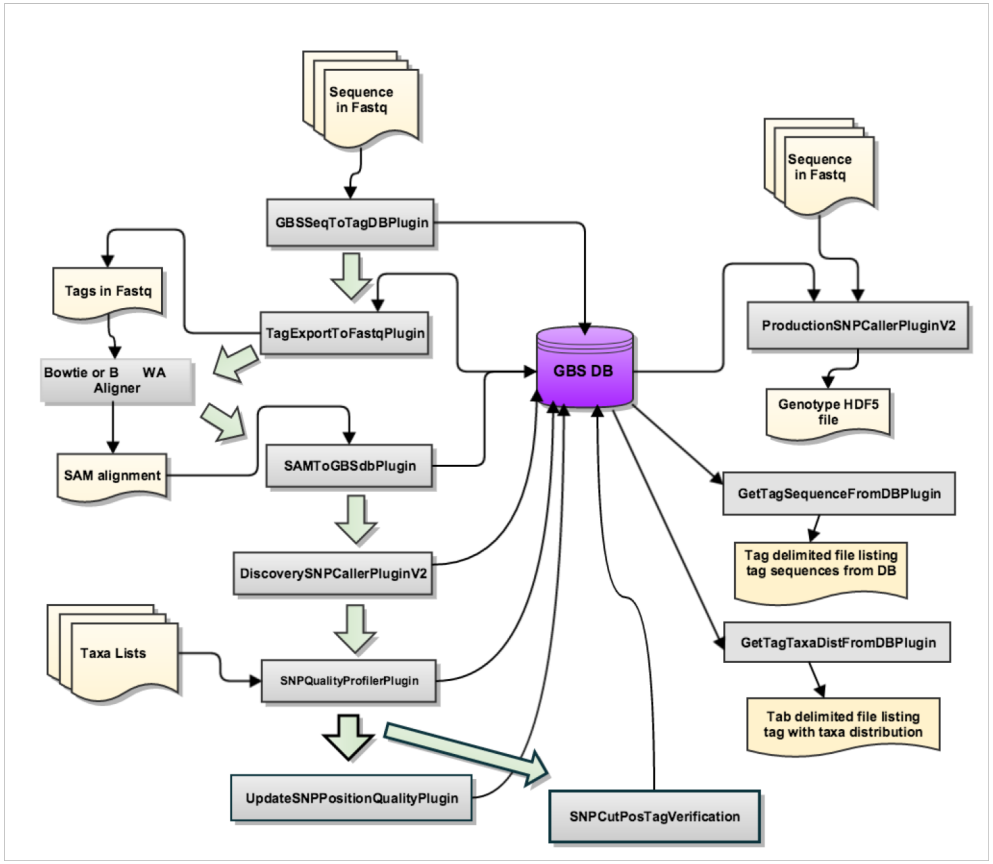


The code to run a branching workflow is only a bit more complex than the basic workflow. You can find more branching examples in the advanced tutorial and cookbook sections.

TASSEL - Trait Analysis by aSSociation, Evolution and Linkage

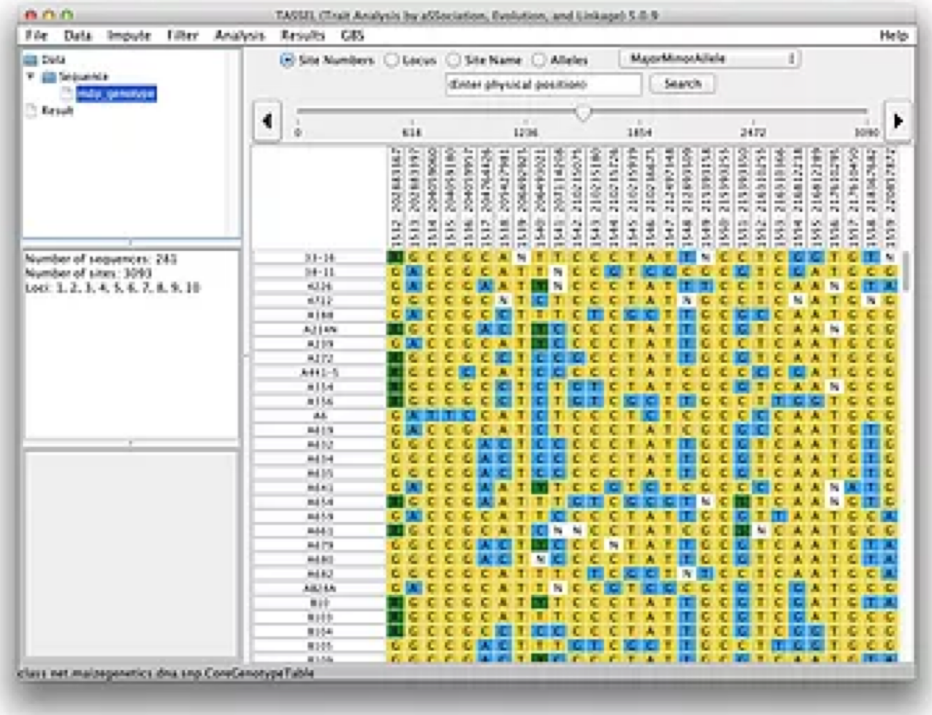
- Webpage
- Manual
- Rtassel (conection to R!)

Useful for species with reference genomes because it can take advantage of diversity pannels

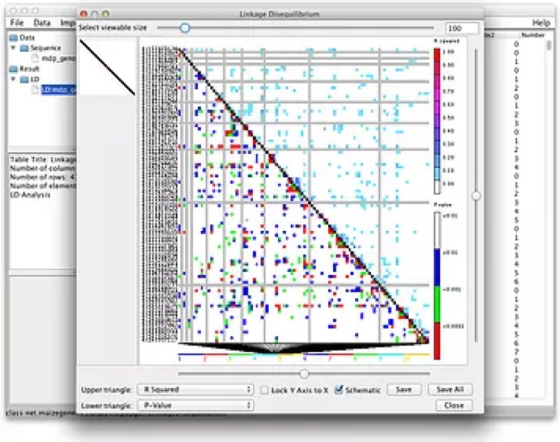


GBSv2 Discovery/Production Pipeline Overview

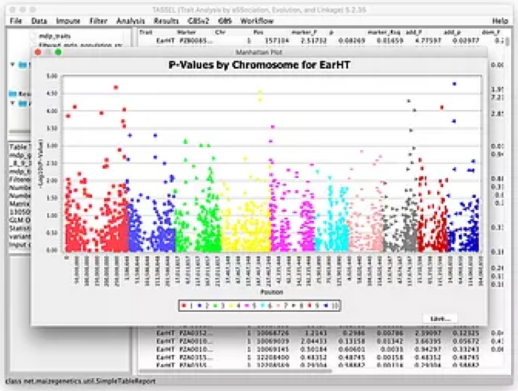
Alignment Viewer



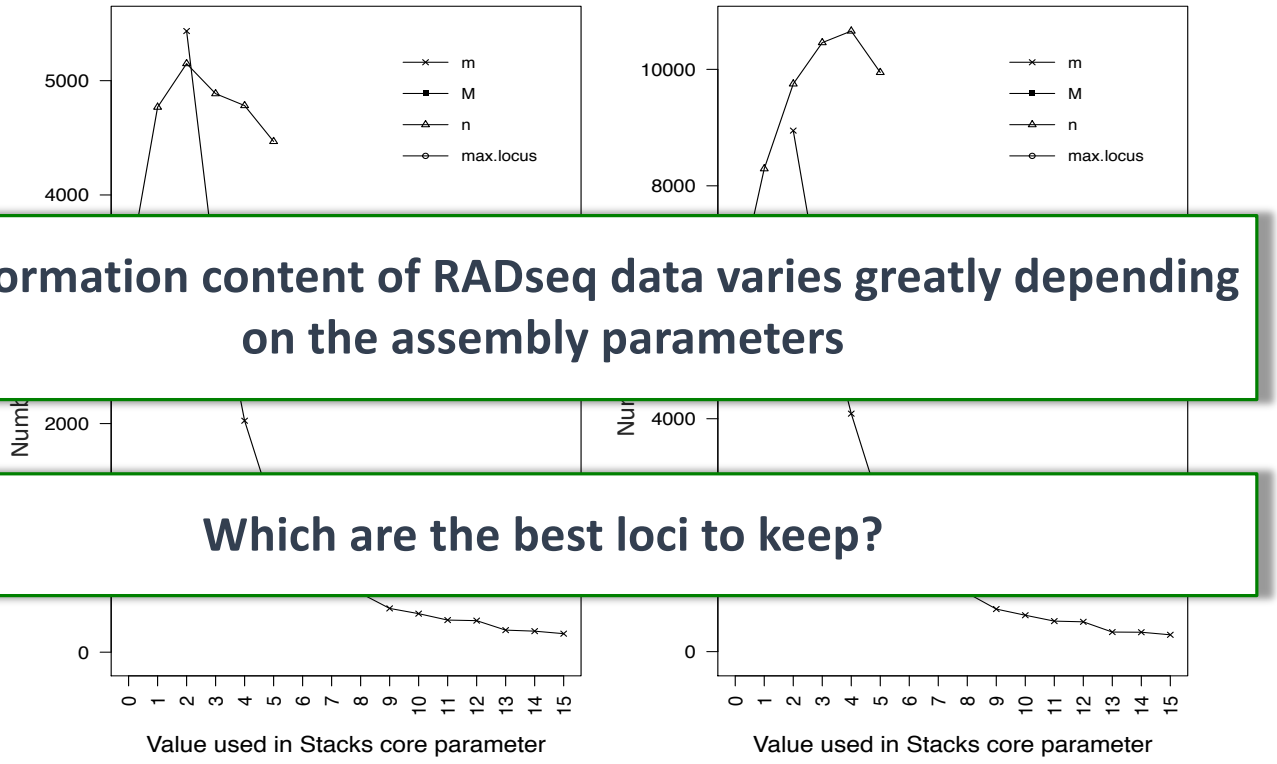
Linkage Disequilibrium Display



Manhattan Plot



¿Cómo optimizer los parámetros de un ensamble *de novo*?



The information content of RADseq data varies greatly depending on the assembly parameters

Which are the best loci to keep?

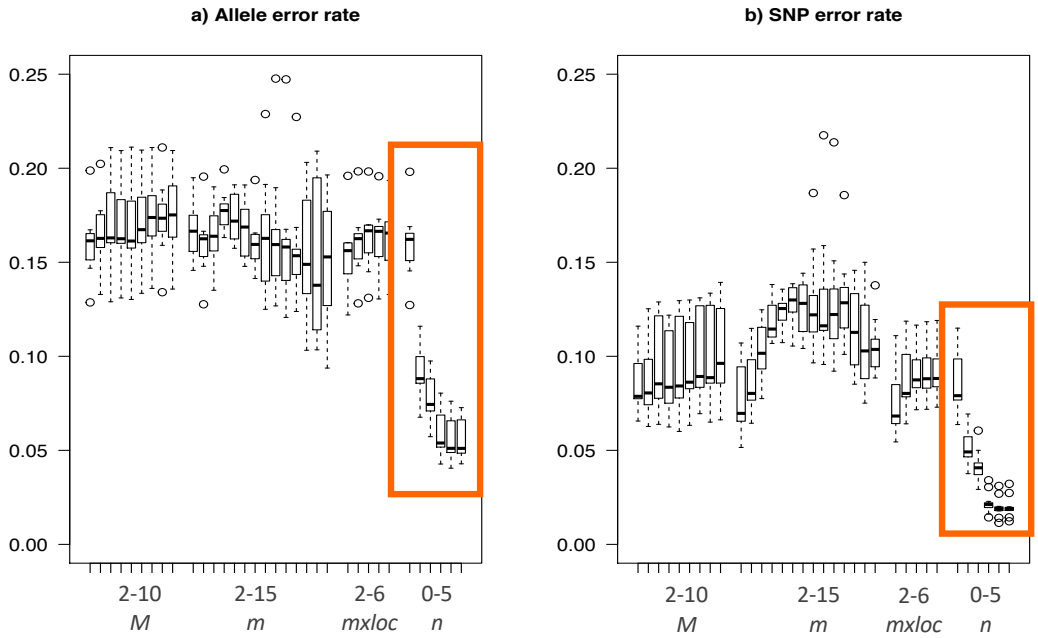
-m min. raw reads to form a stack ("minimal coverage")

-M mismatches between stacks when processing an individual

-max_locus maximum stacks allowed per locus

-n mismatches between loci when building the catalog

!!!



- **RADseq data has error, and it can be very high (specially in some protocols)**
- **You won't know it without replicates**
- **Error can be decreased by tuning the assembly parameter values**

-m min. raw reads to form a stack ("minimal coverage")
-M mismatches between stacks when processing an individual

-mxloc maximum stacks allowed per locus
-n mismatches between loci when building the catalog

Optimal parameter values depend on:

- Polymorphism of the genome
- Sequencing error
- Depth of sequencing

Explore parameter values for each data set

r80 method.

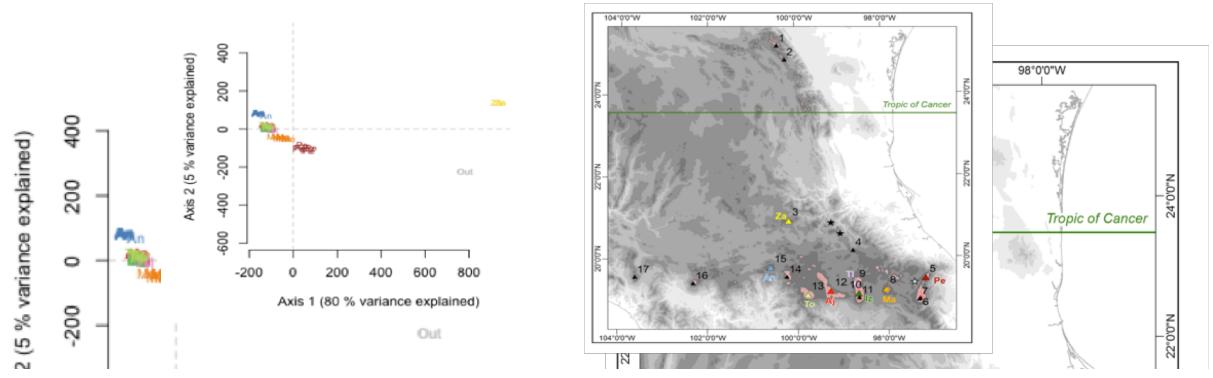
- Vary parameters and select those values which maximize the number of polymorphic loci found in 80% of the individuals in your study.
- J. Paris, J. Stevens, & J. Catchen (2017). *Lost in parameter space: a road map for Stacks*. **Methods in Ecology and Evolution**.

Replicates method

- Vary parameters in a set of replicated samples (same DNA, sequenced independently), and select those values which minimize error rates and maximize number of loci.
- Mastretta-Yanes, A., Arrigo, N., Alvarez, N., Jorgensen, T. H., Piñero, D., & Emerson, B. C. (2015). *Restriction site-associated DNA sequencing, genotyping error estimation and de novo assembly optimization for population genetic inference.* **Molecular Ecology Resources**, *15*(1), 28–41. doi: 10.1111/1755-0998.12291

Stacks parameters optimization example

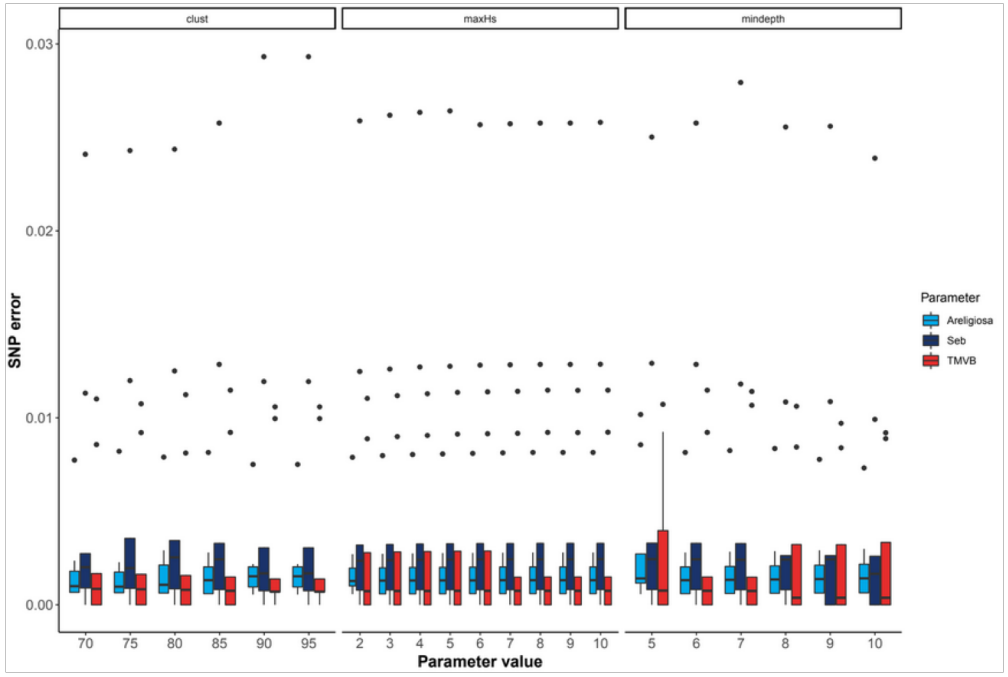
	optimal	near optimal	high coverage	default
*Variation explained by first two axes of PCoA**	80%	82%	47%	57%
*Mean of F_{ST} pairwise matrix**	0.19	0.15	0.03	0.07

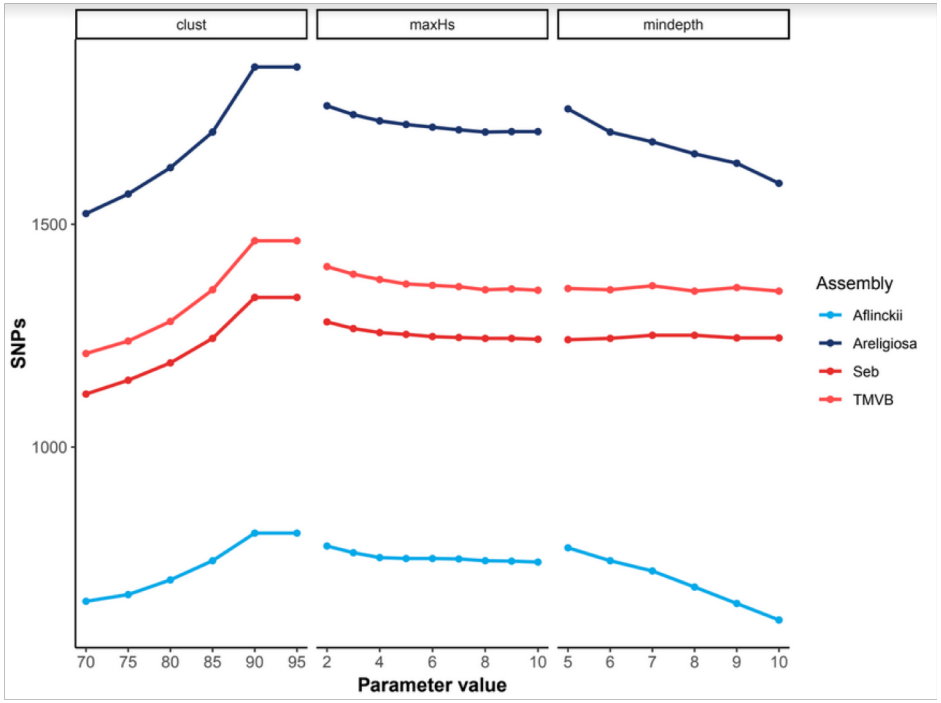


	optimal	near optimal	high coverage	default
Number of **RAD-loci**	6,292	2,449	292	4,554
Number **SNPs**	11,057	4,353	502	7,736
Mean **coverage**	10.32 (SD 4.16)	15.30 (SD 5.9)	58.92 (SD 21.9)	11.50 (SD 4.65)

Mastretta-Yanes et al (2015)

ipyrad parameters optimization example

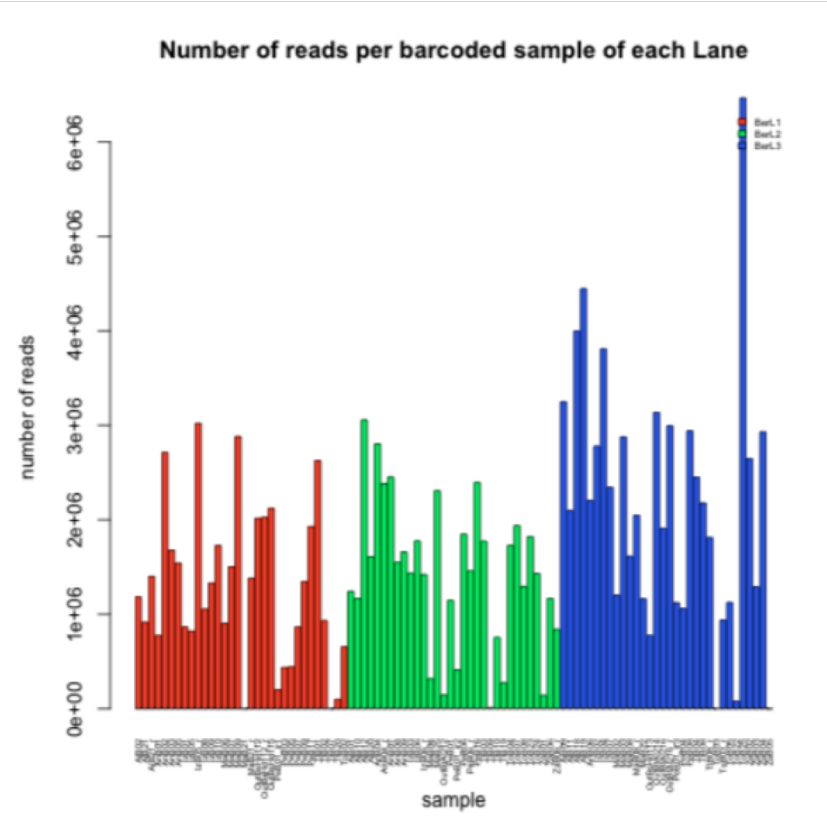




Giles et al (in prep)

Primera inspección de los datos y recomendaciones

Basic plots can tell you a lot



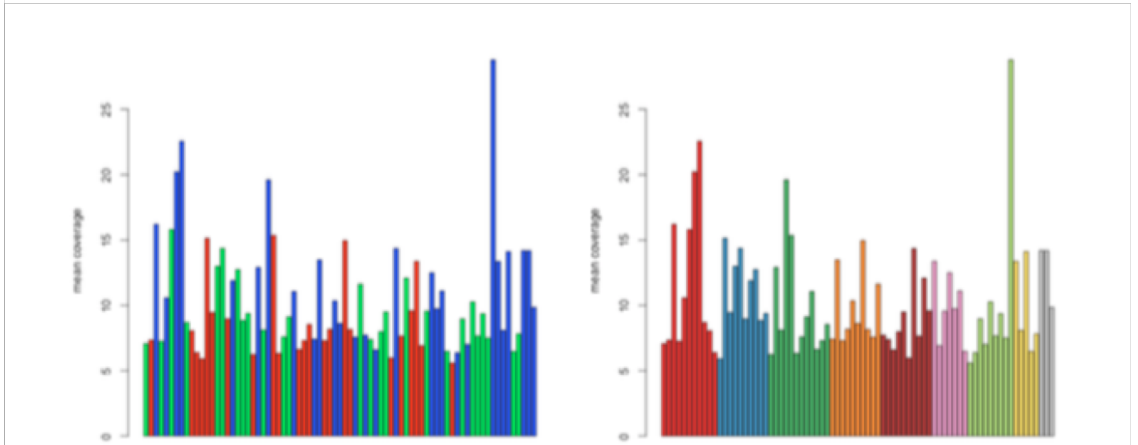
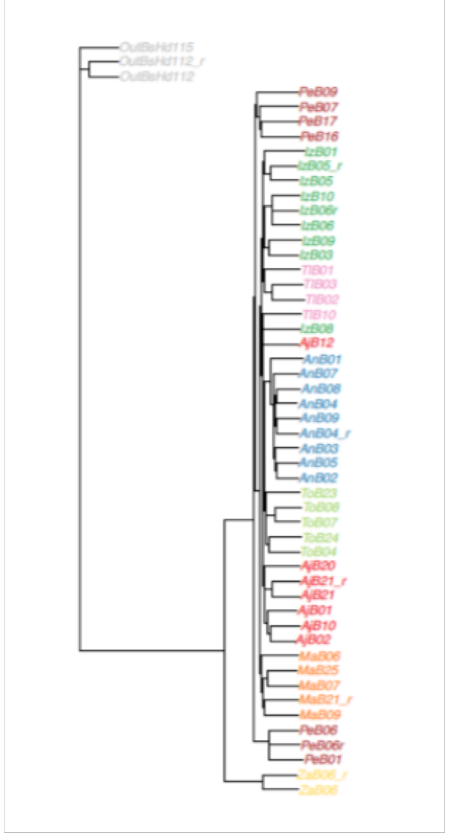
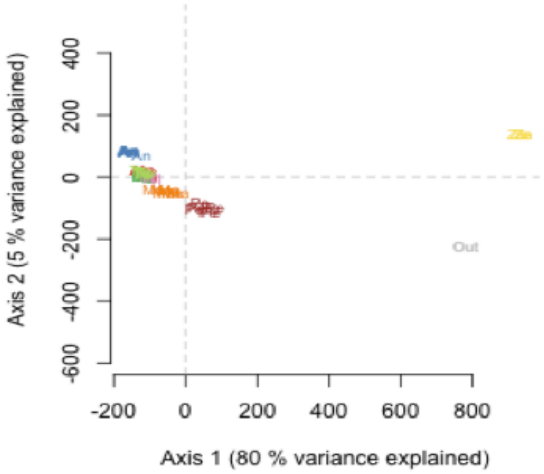
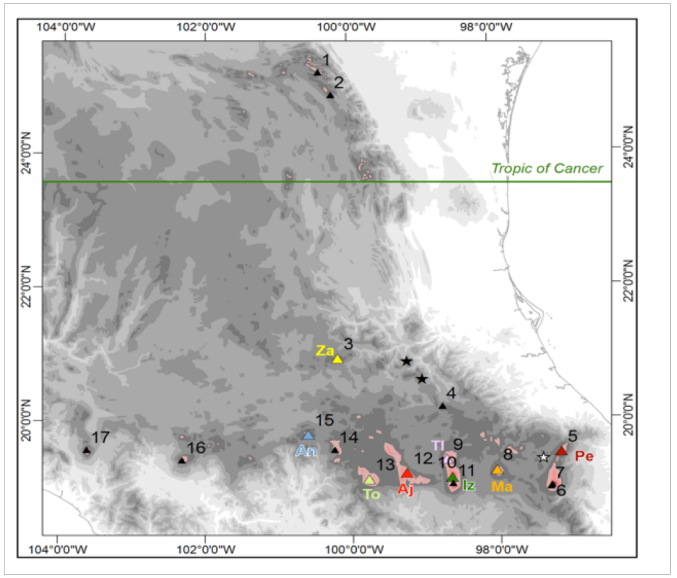


Figure S1.3. Mean coverage per sample after processing the data with *Stacks* optimal profile settings (see results). Left: color key corresponding to sequencing lanes as in Fig. S1.1. Right: color key corresponding to geographic origin of samples as in Fig. 1 (main text).

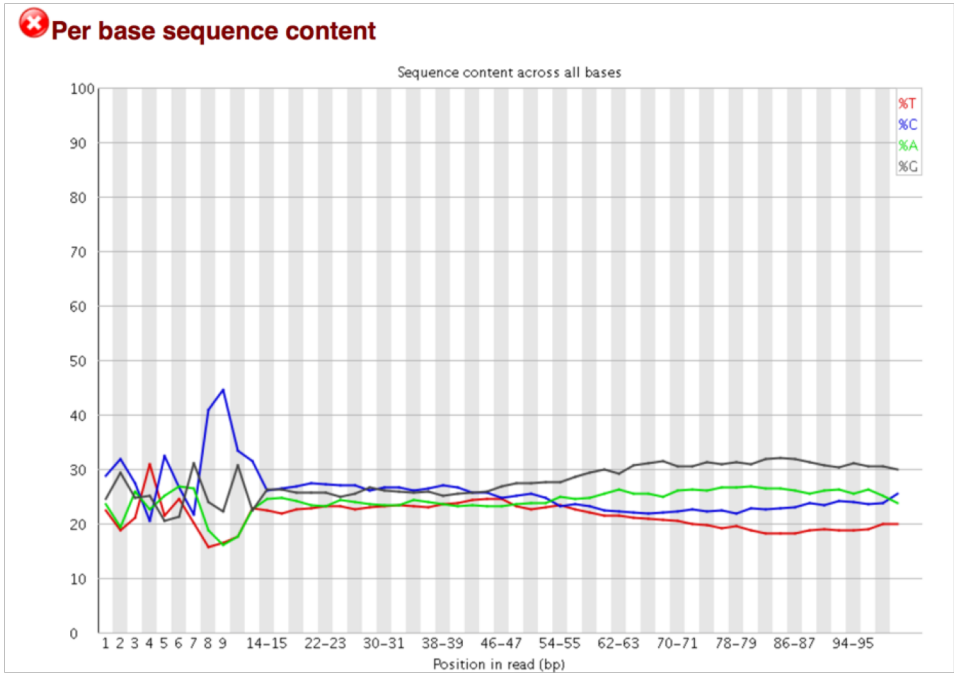
Basic plots can tell you a lot





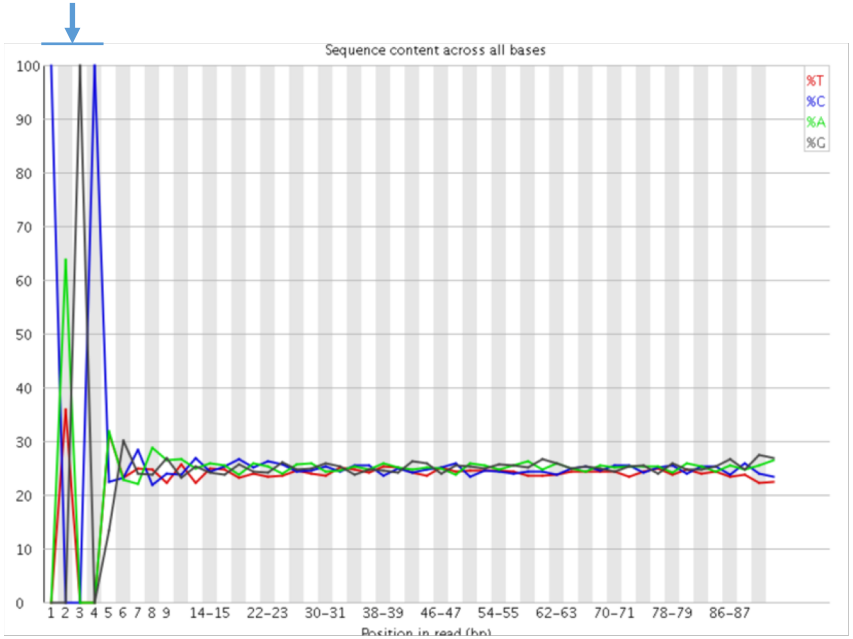


Fastqc reports will detect a bias in the sequence content, but it is fine, it’s the restriction site ;)



Raw reads with adaptor

apeKI = GWGC (W=A o T)
Restriction site



Reads after filtering adaptor still showing weird sequence content -> it’s the restriction site

Other bioinformatic recommendations:

Continue filtering

Once you have a **base** assembly, you can continue filtering the data set by:

- MAF
- Missing data (individual / loci)

VCF and plink are useful for this.

Some time it is better to discard an individual with too much missing data to increase the number of loci.

Assemble different sets of samples if you have several taxa

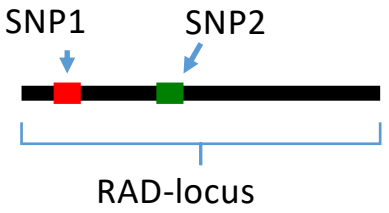
The more distant the taxa are, the more likely a restriction site mutated (thus the enzyme can't cut it) in one taxa and not another. This increases missing data in the individuals with the mutated site. As a consequence:

The more phylogenetically distant the individuals, the LESS loci the assembly will produce

You can re-assembly the same raw sequences data with different sets of samples (individuals) for different analyses: ie "all taxa" for a phylogenetic analysis and "in group samples" for population genetics.

In your **methods** section report:

- Enzymes and protocol used in wet lab
- Number of initial samples
- Parameters used for the assembly/mapping
- Whether you exported **all SNPs** of the **first/random SNP per locus**
- Any posterior filtering (MAF, missing data by individual / locus)



In your **results** section report:

- Number of samples successfully assembled and that passed the post- filters
- Number of RAD-locus and/or SNPs produced
- % missing data
- Error rates (if you included replicates to estimate them)

Una nota sobre parálogos

Gene duplication importance

Evolutionary novelty

- Neofunctionalization
- Subfunctionalization
- others

Ecologically significant polymorphisms

Passive genomic divergence

- Genomic incompatibilities

Postzygotic barriers

Examining gene duplication with ddRAD data

GENOME BIOLOGY AND EVOLUTION

GBE

Gene Duplication, Population Genomics, and Species-Level Differentiation within a Tropical Mountain Shrub

Alicia Mastretta-Yanes[1,*], Sergio Zamudio[2], Tove H. Jorgensen[3], Nils Arrigo[4], Nadir Alvarez[4], Daniel Piñero[5], and Brent C. Emerson[1,6]

[1]Centre for Ecology, Evolution and Conservation, School of Biological Sciences, University of East Anglia, Norwich Research Park, Norwich, United Kingdom

[2]Centro Regional del Bajío, Instituto de Ecología A. C., Pátzcuaro, Michoacán, México

[3]Department of Bioscience, Aarhus University, Denmark

[4]Department of Ecology and Evolution, Biophore Building, University of Lausanne, Switzerland

[5]Departamento de Ecología Evolutiva, Instituto de Ecología, Universidad Nacional Autónoma de México, Mexico

[6]Island Ecology and Evolution Research Group, Instituto de Productos Naturales y Agrobiología (IPNA-CSIC), San Cristóbal de La Laguna, Santa Cruz de Tenerife, Spain

*Corresponding author: E-mail: a.yanes@uea.ac.uk.

Accepted: September 8, 2014

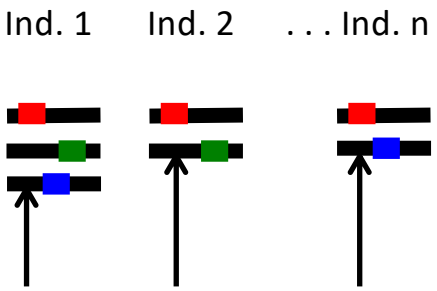
de novo assembly and paralogs identification

During assembly

- Keep paralogs of putative recent origin

Post assembly

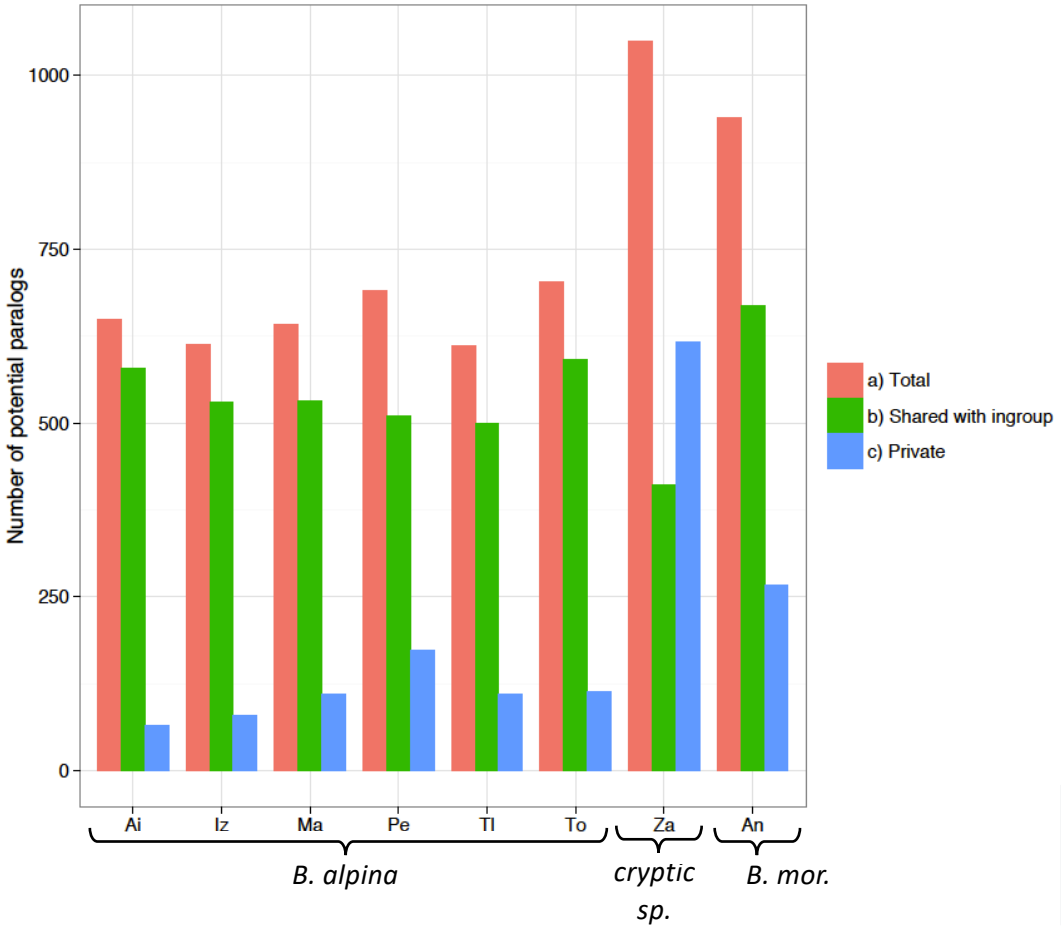
- Site Frequency Spectrum (SFS)
- Identify potential paralogous loci per population / spp.
- Examine distribution of *shared* and *private* potential paralogs

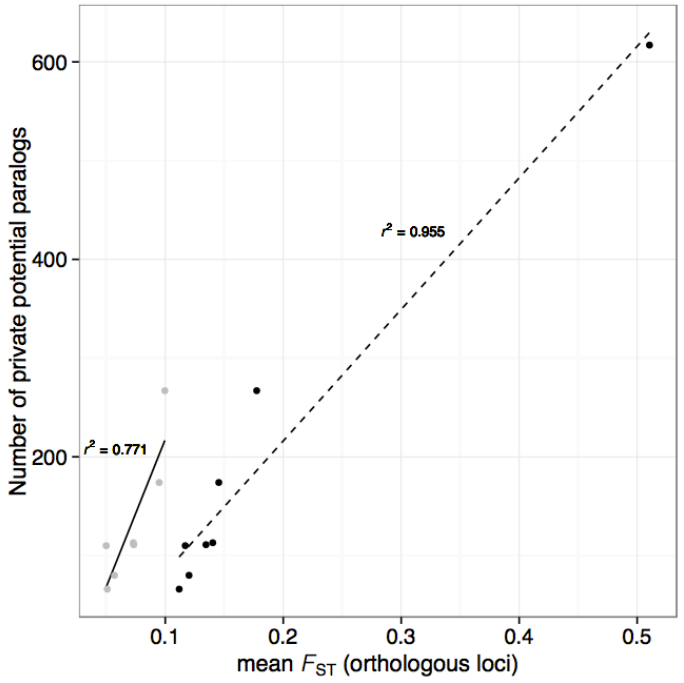


Spurious polymorphic positions at which all individuals are heterozygous

Bias towards heterozygosity & excess of p=0.5

RESULTS: Paralogs – distribution among pops. & spp.





Paralogous loci as a source of population differentiation