SPLITTER: a pipeline for RADseq preliminary analysis
version 1.1

Arraiano-Castilho, Michael R. Miller, Albano Beja-Pereira

Disclaimer

Although SPLITTER has been tested and, to the best of our knowledge, works without problems, the authors take no responsibility for any bugs, errors, or any harm or damage caused by the software or the documentation. The software and the documentation come without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. Use the software and the documentation at your own risk.

Contents

1 About SPLITTER

SPLITTER pipeline is written in Python and conducts a preliminary analysis when dealing with large data sets with particular emphasis in complex of species with unresolved taxonomic issues. Here we describe the optimal work flow based in Miller et al. [2012] assuming the output of the alignment program Novoalign (Novocraft Technologies) as described in Figure 1. Users are free to choose the best suited method to get the output of novoalign keeping in mind that this is the crucial step to run SPLITTER. The main goal is to summarize the data set and get rates of similarity of all individuals with a reference. Throughout this manual we used the provided dataset to help understand the pipeline progress across all steps.

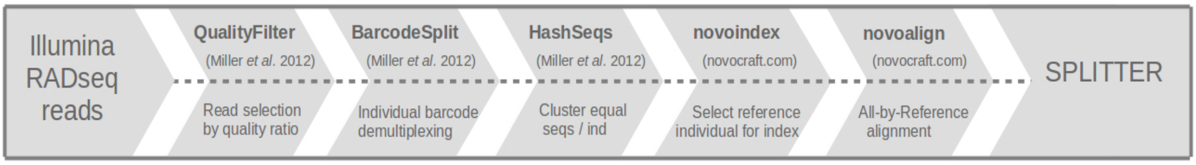


Figure 1: Simplified flowchart depicting the required previous steps for SPLITTER pipeline application. Part of this work flow was described by Miller et al. [2012] and it is written in Perl typically conducted in RADseq analyses. The flow starts with the input file of RADseq reads filtered by a quality filter removing bad reads and sequencing errors. The individually indexed samples from RAD libraries are then demultiplexed by specific barcodes and split by individuals. Hashing sequences clustering and scoring equal sequences per individual is required for further alignment steps. Hereafter it is necessary to choose the individual or group of individuals to act as reference for the alignment. The result of the alignment of all individuals with the reference should be the input to run the SPLITTER

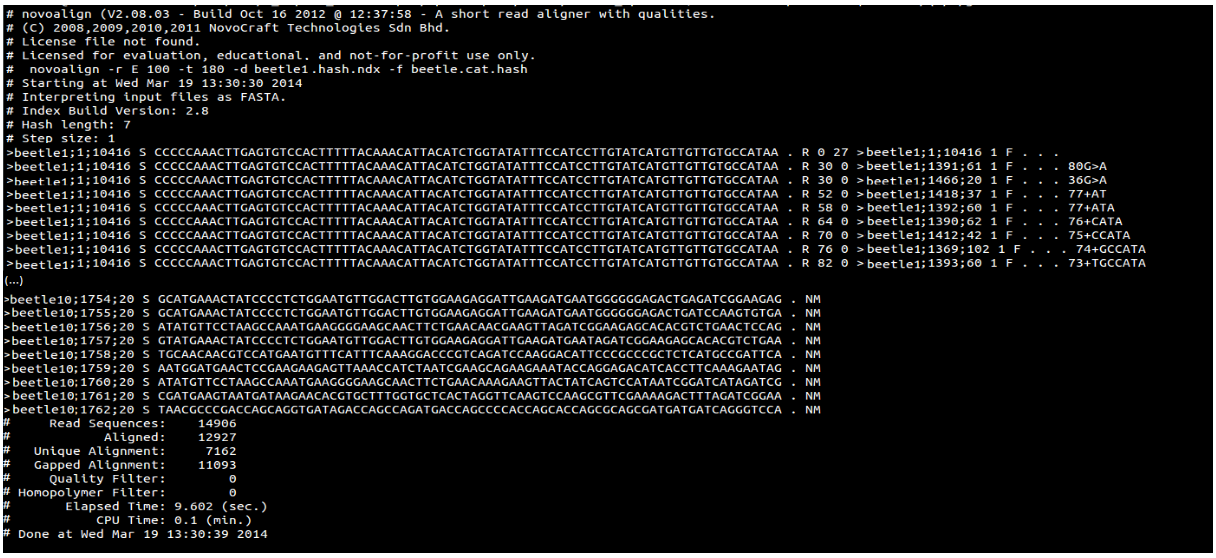
2 Citation

A resource article about SPLITTER has been submitted to the journal Bioinformatics. If you use SPLITTER, the package should be cited as:
Publication status: **submitted**

3 The Input files

In order to run SPLITTER two input files are required; i) the output of novoalign, that can be in native report format (.novo) or in Sequence Alignment/Map (.sam) and; ii) a list of your samples/individuals. A template of each of these files comes with the example data. We strongly recommend reading about the output formats in novoalign user's manual.

3.1 input.novo

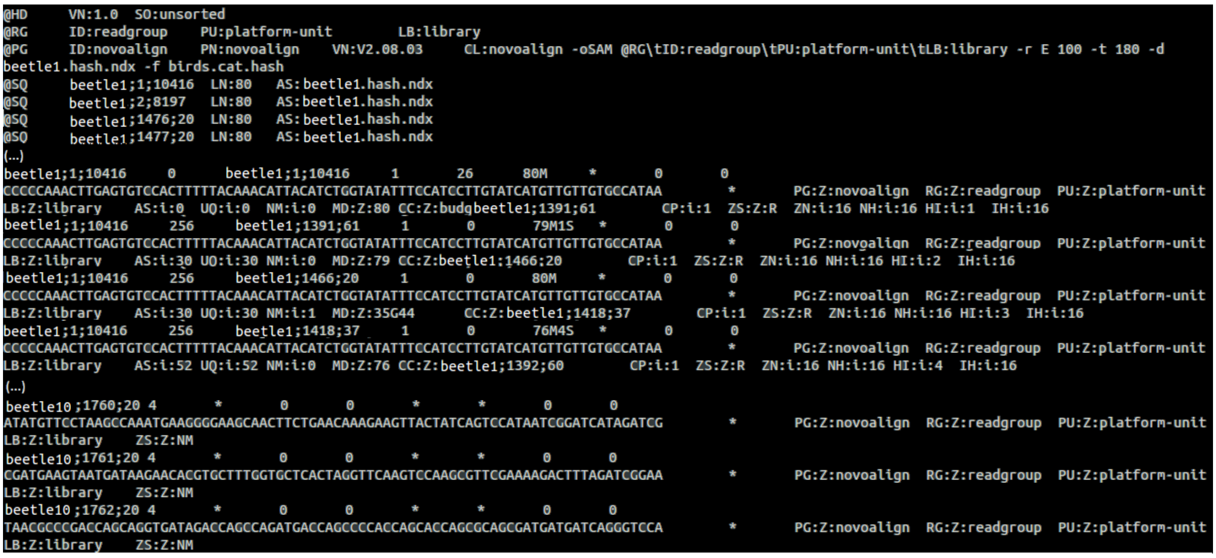


```
# novoalign (V2.08.03 - Build Oct 16 2012 @ 12:37:58 - A short read aligner with qualities.
# (C) 2008,2009,2010,2011 NovoCraft Technologies Sdn Bhd.
# License file not found.
# Licensed for evaluation, educational, and not-for-profit use only.
#  novoalign -r E 100 -t 180 -d beetle1.hash.ndx -f beetle.cat.hash
# Starting at Wed Mar 19 13:30:30 2014
# Interpreting input files as FASTA.
# Index Build Version: 2.8
# Hash length: 7
# Step size: 1
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 0 27 >beetle1;1;10416 1 F . . .
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 30 0 >beetle1;1391;61 1 F . . . 80G>A
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 30 0 >beetle1;1466;20 1 F . . . 36G>A
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 52 0 >beetle1;1418;37 1 F . . . 77+AT
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 58 0 >beetle1;1392;60 1 F . . . 77+ATA
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 64 0 >beetle1;1390;62 1 F . . . 76+CATA
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 70 0 >beetle1;1412;42 1 F . . . 75+CCATA
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 76 0 >beetle1;1369;102 1 F . . . 74+GCCATA
>beetle1;1;10416 S CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA . R 82 0 >beetle1;1393;60 1 F . . . 73+TGCCATA
(...)
>beetle10;1754;20 S GCATGAAACTATCCCCTCTGGAATGTTGGACTTGTGGAAGAGGATTGAAGATGAATGGGGGAGACTGAGATCGGAAGAG . NM
>beetle10;1755;20 S GCATGAAACTATCCCCTCTGGAATGTTGGACTTGTGGAAGAGGATTGAAGATGAATGGGGGAGACTGATCCAAGTGTGA . NM
>beetle10;1756;20 S ATATGTTCCTAAGCCAAATGAAGGGGAAGCAACTTCTGAACAACGAAGTTAGATCGGAAGAGCACACGTCTGAACTCCAG . NM
>beetle10;1757;20 S GTATGAAACTATCCCCTCTGGAATGTTGGACTTGTGGAAGAGGATTGAAGATGAATAGATCGGAAGAGCACACGTCTGAA . NM
>beetle10;1758;20 S TGCAACAACGTCCATGAATGTTTCATTTCAAAGGACCCGTCAGATCCAAGGACATTCCCGCCCGCTCTCATGCCGATTCA . NM
>beetle10;1759;20 S AATGGATGAACTCCGAAGAAGAGTTAAACCATCTAATCGAAGCAGAAGAAATACCAGGAGACATCACCTTCAAAGAATAG . NM
>beetle10;1760;20 S ATATGTTCCTAAGCCAAATGAAGGGGAAGCAACTTCTGAACAAAGAAGTTACTATCAGTCCATAATCGGATCATAGATCG . NM
>beetle10;1761;20 S CGATGAAGTAATGATAAGAACACGTGCTTTGGTGCTCACTAGGTTCAAGTCCAAGCGTTCGAAAAGACTTTAGATCGGAA . NM
>beetle10;1762;20 S TAACGCCCGACCAGCAGGTGATAGACCAGCCAGATGACCAGCCCCACCAGCACCAGCGCAGCGATGATGATCAGGGTCCA . NM
#     Read Sequences:    14906
#            Aligned:    12927
#   Unique Alignment:     7162
#   Gapped Alignment:    11093
#     Quality Filter:        0
# Homopolymer Filter:        0
#       Elapsed Time: 9.602 (sec.)
#           CPU Time: 0.1 (min.)
# Done at Wed Mar 19 13:30:39 2014
```

Figure 2: The native report format: lines started with ">" character are comments and lines started with "#" are aligned reads against the index.

3.2 input.sam

The SAM report format from novoalign is the following:

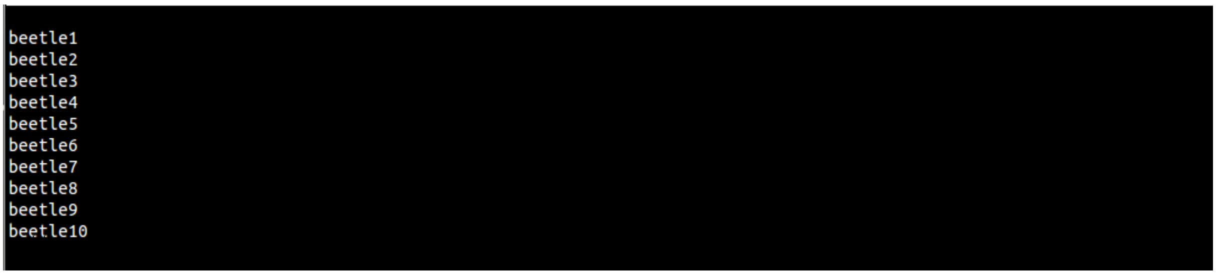


```
@HD     VN:1.0  SO:unsorted
@RG     ID:readgroup    PU:platform-unit        LB:library
@PG     ID:novoalign    PN:novoalign    VN:V2.08.03     CL:novoalign -oSAM @RG\tID:readgroup\tPU:platform-unit\tLB:library -r E 100 -t 180 -d
beetle1.hash.ndx -f birds.cat.hash
@SQ     beetle1;1;10416  LN:80   AS:beetle1.hash.ndx
@SQ     beetle1;2;8197   LN:80   AS:beetle1.hash.ndx
@SQ     beetle1;1476;20  LN:80   AS:beetle1.hash.ndx
@SQ     beetle1;1477;20  LN:80   AS:beetle1.hash.ndx
(...)
beetle1;1;10416     0       beetle1;1;10416     1       26      80M     *       0       0
CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    AS:i:0  UQ:i:0  NM:i:0  MD:Z:80 CC:Z:budgbeetle1;1391;61        CP:i:1  ZS:Z:R  ZN:i:16 NH:i:16 HI:i:1  IH:i:16
beetle1;1;10416     256     beetle1;1391;61     1       0       79M1S   *       0       0
CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    AS:i:30 UQ:i:30 NM:i:0  MD:Z:79 CC:Z:beetle1;1466;20        CP:i:1  ZS:Z:R  ZN:i:16 NH:i:16 HI:i:2  IH:i:16
beetle1;1;10416     256     beetle1;1466;20     1       0       80M     *       0       0
CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    AS:i:30 UQ:i:30 NM:i:1  MD:Z:35G44      CC:Z:beetle1;1418;37        CP:i:1  ZS:Z:R  ZN:i:16 NH:i:16 HI:i:3  IH:i:16
beetle1;1;10416     256     beetle1;1418;37     1       0       76M4S   *       0       0
CCCCCAAACTTGAGTGTCCACTTTTTACAAACATTACATCTGGTATATTTCCATCCTTGTATCATGTTGTTGTGCCATAA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    AS:i:52 UQ:i:52 NM:i:0  MD:Z:76 CC:Z:beetle1;1392;60        CP:i:1  ZS:Z:R  ZN:i:16 NH:i:16 HI:i:4  IH:i:16
(...)
beetle10;1760;20 4      *       0       0       *       *       0       0
ATATGTTCCTAAGCCAAATGAAGGGGAAGCAACTTCTGAACAAAGAAGTTACTATCAGTCCATAATCGGATCATAGATCG        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    ZS:Z:NM
beetle10;1761;20 4      *       0       0       *       *       0       0
CGATGAAGTAATGATAAGAACACGTGCTTTGGTGCTCACTAGGTTCAAGTCCAAGCGTTCGAAAAGACTTTAGATCGGAA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    ZS:Z:NM
beetle10;1762;20 4      *       0       0       *       *       0       0
TAACGCCCGACCAGCAGGTGATAGACCAGCCAGATGACCAGCCCCACCAGCACCAGCGCAGCGATGATGATCAGGGTCCA        *       PG:Z:novoalign  RG:Z:readgroup  PU:Z:platform-unit
LB:Z:library    ZS:Z:NM
```

Figure 3: SAM report format: Header lines start with "@" and the remaining are aligned reads.

3.3 RADlist.txt

This is the list of all samples used in the analysis. It should be one sample per line without spaces or delimiters.

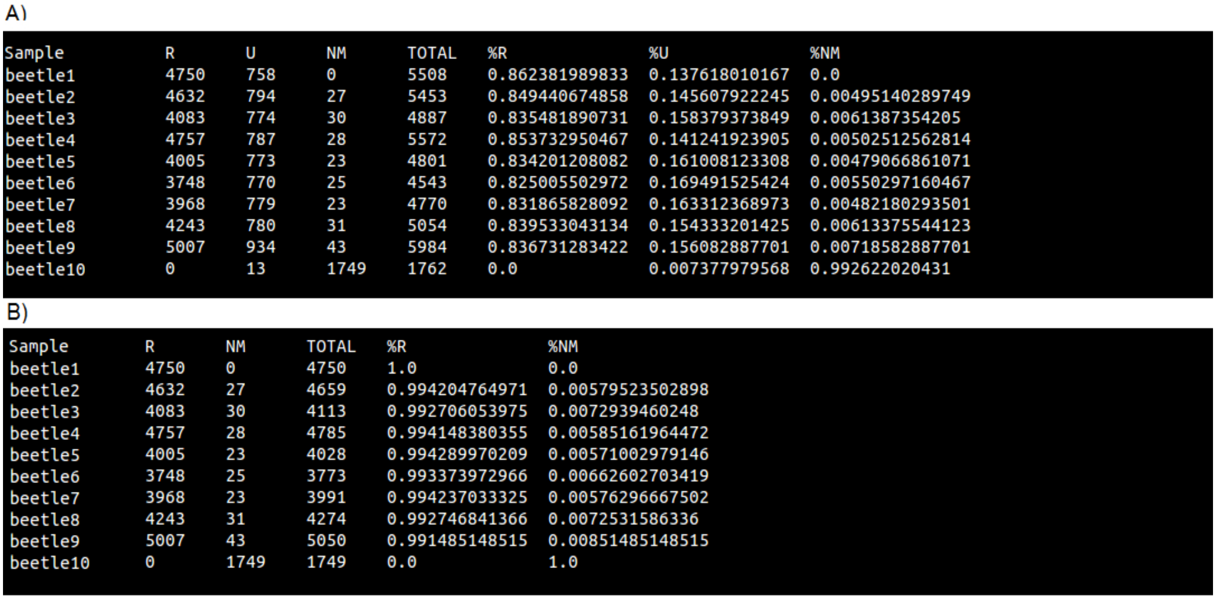


```
beetle1
beetle2
beetle3
beetle4
beetle5
beetle6
beetle7
beetle8
beetle9
beetle10
```

Figure 4: : Sample list with one sample per line, without spaces or delimiters.

4 The output file

Depending on the input format used in SPLITTER the output will be lightly different. The difference is the absence of unique reads information in SAM output but present in novo. The output file is presented in Figure 5. It is tab delimited and it is organized in three or four columns (sam and novo respectively): i) sample name; ; ii) number of multiple alignments with similar score (R) iii) number of single alignments with respective score (U) (only present in .novo) and iv) number of non aligned sequences (NM); v) The total number of reads in the alignment; vi) and the remaining columns are the proportions (%) of each.



A)

```
Sample      R     U     NM    TOTAL  %R              %U              %NM
beetle1     4750  758   0     5508   0.862381989833  0.137618010167  0.0
beetle2     4632  794   27    5453   0.849440674858  0.145607922245  0.00495140289749
beetle3     4083  774   30    4887   0.835481890731  0.158379373849  0.0061387354205
beetle4     4757  787   28    5572   0.853732950467  0.141241923905  0.00502512562814
beetle5     4005  773   23    4801   0.834201208082  0.161008123308  0.00479066861071
beetle6     3748  770   25    4543   0.825005502972  0.169491525424  0.00550297160467
beetle7     3968  779   23    4770   0.831865828092  0.163312368973  0.00482180293501
beetle8     4243  780   31    5054   0.839533043134  0.154333201425  0.00613375544123
beetle9     5007  934   43    5984   0.836731283422  0.156082887701  0.00718582887701
beetle10    0     13    1749  1762   0.0             0.007377979568  0.992622020431
```

B)

```
Sample      R     NM    TOTAL  %R              %NM
beetle1     4750  0     4750   1.0             0.0
beetle2     4632  27    4659   0.994204764971  0.00579523502898
beetle3     4083  30    4113   0.992706053975  0.0072939460248
beetle4     4757  28    4785   0.994148380355  0.00585161964472
beetle5     4005  23    4028   0.994289970209  0.00571002979146
beetle6     3748  25    3773   0.993373972966  0.00662602703419
beetle7     3968  23    3991   0.994237033325  0.00576296667502
beetle8     4243  31    4274   0.992746841366  0.0072531586336
beetle9     5007  43    5050   0.991485148515  0.00851485148515
beetle10    0     1749  1749   0.0             1.0
```

Figure 5: : SPLITTER output files: A) output from native report format and B) output from SAM report format.

The output file present the summary rates of each individual related to the reference previously selected (here was beetle1). The individual beetle10 shows the highest rate of non aligned sequences (NM) and no alignment score (R) which was expected as because it is a different species

5 Installation

SPLITTER pipeline don't need any specific installation procedure. It is ready to use since it is downloaded from the web repository http://radsplitter.github.io/SPLITTER

6 System Requirements

The pipeline was developed and is fully dependent of UNIX/Linux-based operating system. It was designed to be used from command line terminal and requires basic knowledge of command-line syntax. It will run on any machine that has *Python* v2.7.6 and *Perl* v5.18.2 (or compatible versions) installed.
To check *Python* and *Perl* version in your machine type the following command in the terminal:

```
$ python -V
$ perl -V
```

You will also need a simple text editor and office suite for edit the final output (optional).

7 Work Flow

As previously described the work flow described here is based on Miller et al. [2012]. Here we detail the steps taken to get the final output of SPLITTER. All scripts referred here, can be obtained by contacting the corresponding author from Miller et al. [2012] or in SPLITTER repository at http://radsplitter.github.io/SPLITTER/
We will conduct this tutorial assuming that you allocate SPLITTER components in a specific directory of your machine. The preliminary scripts from Miller et al. [2012] can be allocated in your PATH environmental variable. This facilitate to run the scrips in any directory of your machine. The following are an example only, and should not be interpreted as the optimal method for all data sets.
This work flow is just an example usage since you can run SPLITTER pipeline from any novoalign output in Native or SAM format. To start you need to get your data from Illumina RADseq in .fastq format. This will be the input for the scripts described in Miller et al. [2012] to filter the reads, individual demultiplexing and hash equal reads for individual until we get the novoalign output to start with SPLITTER.

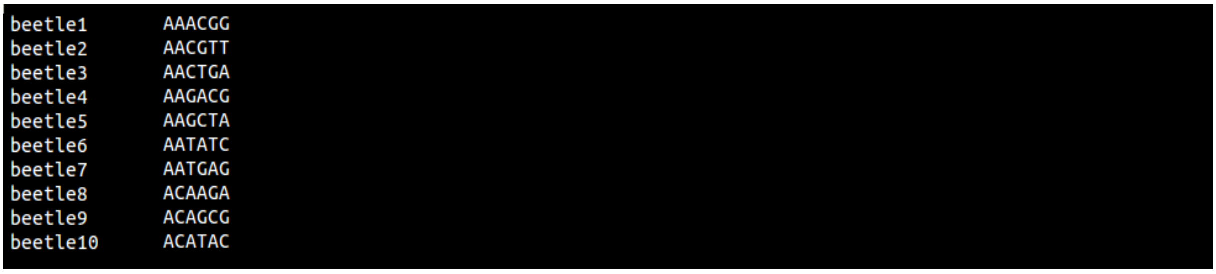
7.1 Step 1: Run the Quality filter

Edit QualityFilter.pl [Miller et al., 2012] parameters: $percent_filter = 80; $length = 80; $phred = 33 We strongly recommend using .fastq quality control software (i.e. FastQC V0.10.1) to determine the optimal trim length of sequence for this step.
Run QualityFilter.pl:

```
$ QualityFilter.pl your_data.fastq >your_data_L80P80.fastq
```

7.2 Step 2: Run Barcode Split

You can demultiplex your individuals in multiple ways using BarcodeSplit.pl [Miller et al., 2012]. Depending on the number of individuals indexed you can call one by one typing the command for each individual (reasonable for a few individuals) or you can generate a script to call the individual barcodes from a text file (e.g. RAD_metadata.txt (Figure 6)) and call all at once (preferable when dealing with many individuals).



```
beetle1     AAACGG
beetle2     AACGTT
beetle3     AACTGA
beetle4     AAGACG
beetle5     AAGCTA
beetle6     AATATC
beetle7     AATGAG
beetle8     ACAAGA
beetle9     ACAGCG
beetle10    ACATAC
```

Figure 6: : RAD_metadata.txt format. One column with sample names and other with RAD multiplex barcode.

The BarcodeSplit syntax is the following:

```
$ BarcodeSplit.pl your_data_L80P80.fastq barcode >your_data _barcode.L80P80.fastq
```

7.3 Step 3: Select a reference

Here we will choose the reference individual against which we will align all the others. The reference should have a good amount of reads. Check the size of the files of each individual, this is proportional to the amount of reads. It is therefore advisable to use several references until a clear grouping pattern is observed.
Type the following command in the directory where you keep your individuals:

```
$ du -h *
```

ATTENTION: if you notice a large discrepancy between individuals you will have to standardize the number of sequences. See Hecht et al. [2013] for more detailed information.

7.4 Step 4: Hash Sequences

The Hashseq.pl [Miller et al., 2012] script identifies all unique sequences within a .fastq of each individual, and outputs a file for each containing a sequence ID for each unique sequence, a count of how many times the sequence occurred and appends a user specified identifier. As in Barcode split here you have to choose again to use this script in multiple ways depending on the number of individuals (See how to do it above).
To hash sequences type the following command:

```
$ HashSeqs.pl individual_barcode.L80P80.fastq barcode >individual_barcode.L80P80.hash
```

7.5 Step 5: Concatenate

In order to use the alignment program *novoalign* we need concatenate all the individuals in the same file. To do this type the following command:

```
$ cat *.hash >beetle.cat.hash
```

ATTENTION: beetle.cat.hash is a fictitious name used in the example data.

7.6 Step 6: Built index

Now we will create an index of the chosen reference individual. See http://www.novocraft.com/main/index.php for software and documentation) and type the following command:

```
$ novoindex beetle1.L80P80.hash.ndx beetle1.L80P80.hash
```

ATTENTION: Here we use the individual *beetle1* because it is the same provided in example data.

7.7 Step 7: Run novoalign

In this step we will align all the individuals using the beetle1 as reference. The parameters here presented are for illustration only. You can get more information about this in http://www.novocraft.com/main/index.php
Please type the following command to obtain the native report format:

```
$ novoalign -r E 100 -t 180 -d beetle1.hash.ndx -f beetle.cat.hash >input.novo
```

or for SAM alignment format:

```
$ novoalign -r E 100 -t 180 -d beetle1.hash.ndx -f beetle.cat.hash -o SAM >input.sam
```

7.8 Step 8: Run SPLITTER

To run SPLITTER you need to make sure that all the required conditions are satisfactory. The syntax require some options and arguments like *-f* (input file) *-t* (input file format) *-l* (sample list) *-o* (output file name).
If the import of some of these options and arguments failed, the program will show an error message, related with that option(s). Here are expressed all options available for SPLITTER:

Usage: splitter [options] [arg]	
–version	show program's version number and exit
-h, –help	show this help message and exit
-f FILENAME, –file=FILENAME	input alignment file
-t INFORMAT	input file format (novo or sam)
-l RADLIST, –list=RADLIST	input sample/barcode list
-o OUTPUT	insert the output file name

Please make sure that all the SPLITTER components are in the same directory. SPLITTER is written in *Python* and to run it is necessary import specific modules (all provided together). Ensure that splitter, *novomod.py* and *sammod.py* are in the same folder. The input files don't need to be in the same directory, but in this case you have to specify the correct directory. Also for the output, you can choose where you want to allocate it.
You can use the trial dataset provided with SPLITTER to test it. To run this type the following:

```
$ ./splitter -f input.novo -t novo -l RADlist.txt -o output.novo
```

or:

```
$ ./splitter -f input.sam -t sam -l RADlist.txt -o ouput.sam
```

ATTENTION: here we are assuming that you have all (the input file and the sample list) in the same directory as SPLITTER pipeline and you want to allocate the output file here also.

8 Troubleshooting

If you have any doubts or if you need some advise during or before running SPLITTER, the authors provide support through the discussion forum (https://groups.google.com/forum/#!forum/splitter-support) or the corresponding author.

References

Benjamin C. Hecht, Nathan R. Campbell, Dean E. Holecek, and Shawn R. Narum. Genome-wide association reveals genetic basis for the propensity to migrate in wild populations of rainbow and steelhead trout. *Molecular Ecology*, 22(11):3061–3076, 2013. ISSN 1365-294X. doi: 10.1111/mec.12082. URL `http://dx.doi.org/10.1111/mec.12082`.

Michael R. Miller, Joseph P. Brunelli, Paul A. Wheeler, Sixin Liu, Caird E. Rexroad, Yniv Palti, Chris Q. Doe, and Gary H. Thorgaard. A conserved haplotype controls parallel adaptation in geographically distant salmonid populations. *Molecular Ecology*, 21(2):237–249, 2012. ISSN 1365-294X. doi: 10.1111/j.1365-294X.2011.05305.x. URL `http://dx.doi.org/10.1111/j.1365-294X.2011.05305.x`.