

Analyse de données RADseq sous Galaxy : l'exemple de STACKS

Yvan Le Bras, Projet e-Biogenouest, CNRS UMR 6074 IRISA-INRIA, Rennes

Cyril Monjeaud, Projet GenoCloud, CNRS UMR 6074 IRISA-INRIA, Rennes

Avec l'utilisation de ressources réalisées par Julian Catchen, Institute of Ecology and Evolution, University of Oregon

Merci à Paul Hohenlohe pour m'avoir fourni les métadonnées précises associées à son article Hohenlohe *et al.* (2010).

Le but de ces exercices est de familiariser les stagiaires avec l'utilisation des données générées par les séquenceurs de nouvelles générations à partir de RRL (Reduced Representation Libraries) comme les tags associés à des sites de restriction (RAD). Ces librairies sont souvent utilisées dans le cadre du génotypage par séquençage, et peuvent fournir un jeu dense de marqueurs SNP (single nucleotide polymorphism) répartis le long du génome. Les stagiaires acquerront de l'expérience avec un pipeline d'analyse nommé STACKS, créé pour l'analyse de ce type de données. Les données permettront alors de générer une cartographie génétique d'une part et d'identifier de potentielles signatures de sélection. Il est possible d'utiliser un organisme avec ou sans génome de référence.

Les participants vont apprendre à:

1. Préparer les données brutes Illumina RAD pour leur analyse en enlevant les lectures de mauvaise qualité et pour démultiplexer un jeu d'échantillons barcodés.
2. Aligner des séquences RAD contre un génome de référence
3. Utiliser Stacks pour assembler les loci RAD, détecter des SNPs, les génotypes et haplotypes pour chaque individu de deux populations.
4. Calculer des statistiques en génétique des populations

A la fin de cette formation, vous devriez savoir:

5. Manipuler les données brutes Illumina de RAD pour les analyser en utilisant une variété de différents paramètres.
6. Aligner les tags RAD contre un génome de référence pour identifier des signatures potentielles de sélection.
7. Etendre ce qui a été appris vers des problèmes plus complexes, les vôtres.

Nous allons utiliser des jeux de données proposés par Julian dans ces formations plus des jeux de données épinoche de l'article d'Hohenlohe et al. 2010. Pour le nettoyage et l'analyse des données, nous nous reposerons principalement sur Galaxy, le pipeline STACKS et BWA.

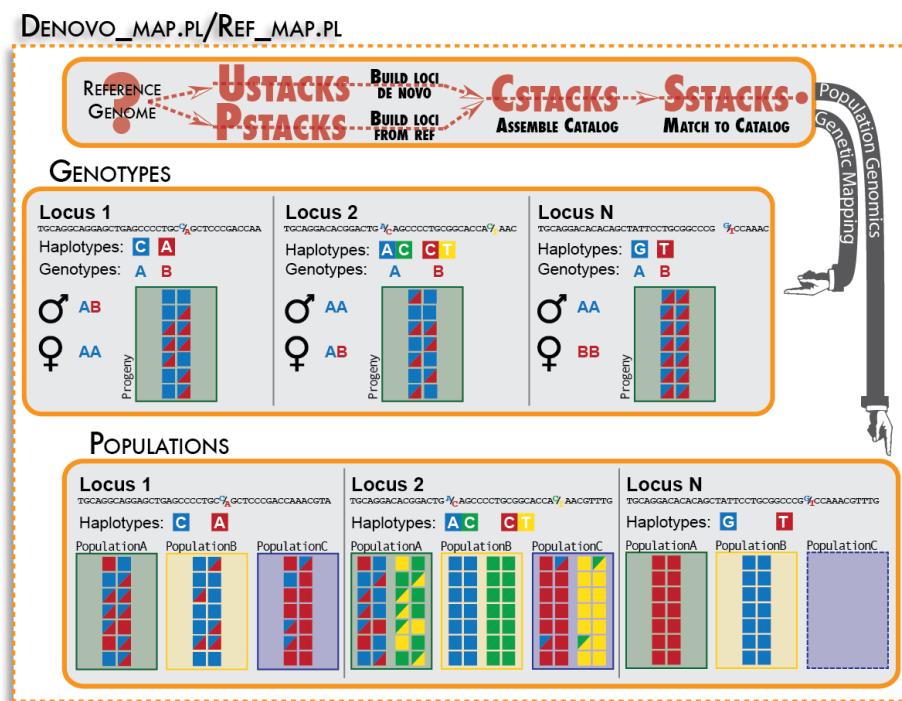
Les jeux de données seront tous produits via des séquenceurs de type Illumina GAI ou HiSeq2000.

Les logiciels sont tous open source

- **BWA** (<http://bio-bwa.sourceforge.net/>) - BWA est utilisé pour aligner des séquences contre un génome de référence. Nous l'utiliserons pour aligner les lectures RAD contre le génome de l'épinoche, puis pour analyser ces lectures via le pipeline Stacks. Si nous utilisons BWA ici, plusieurs autres algorithmes et logiciels existent pour effectuer cette tâche (comme Bowtie) et ils peuvent également être utilisés pour cette partie. Nous avons cependant développé un outil spécialement pour STACKS sous Galaxy, basé sur BWA.

- **Stacks** (<http://creskoloab.uoregon.edu/stacks/>) – il s'agit d'un ensemble de programmes open source interconnectés initialement mis en place pour l'assemblage de novo de séquences RAD en loci et cartes génétiques, aujourd'hui étendu pour être utilisé de manière plus flexible dans des études d'organismes présentant ou non un génome de référence. Le pipeline a un wrapper Perl permettant le lancement de l'ensemble des programmes. La modularité de STACKS lui permet d'être appliqué à différents types de scénarios.

La manière la plus simple d'exécuter le pipeline est d'utiliser les wrappers proposés : **denovo_map** sans génome de référence / **ref_map** avec génome de référence. A noter que le pipeline STACKS utilisant un génome de référence suit les mêmes étapes. La principale différence s'observe pendant la phase de construction des loci. **Ref_map** construit alors les loci à partir du génome de référence, en utilisant les résultats de mapping des lectures sur ce dernier (réalisé via Bowtie, BWA ou un autre logiciel de mapping). Cette étape se nomme alors **pstacks** et se déroule à la place de **uctacks**. **cstacks** et **sstacks** sont ensuite exécutés.



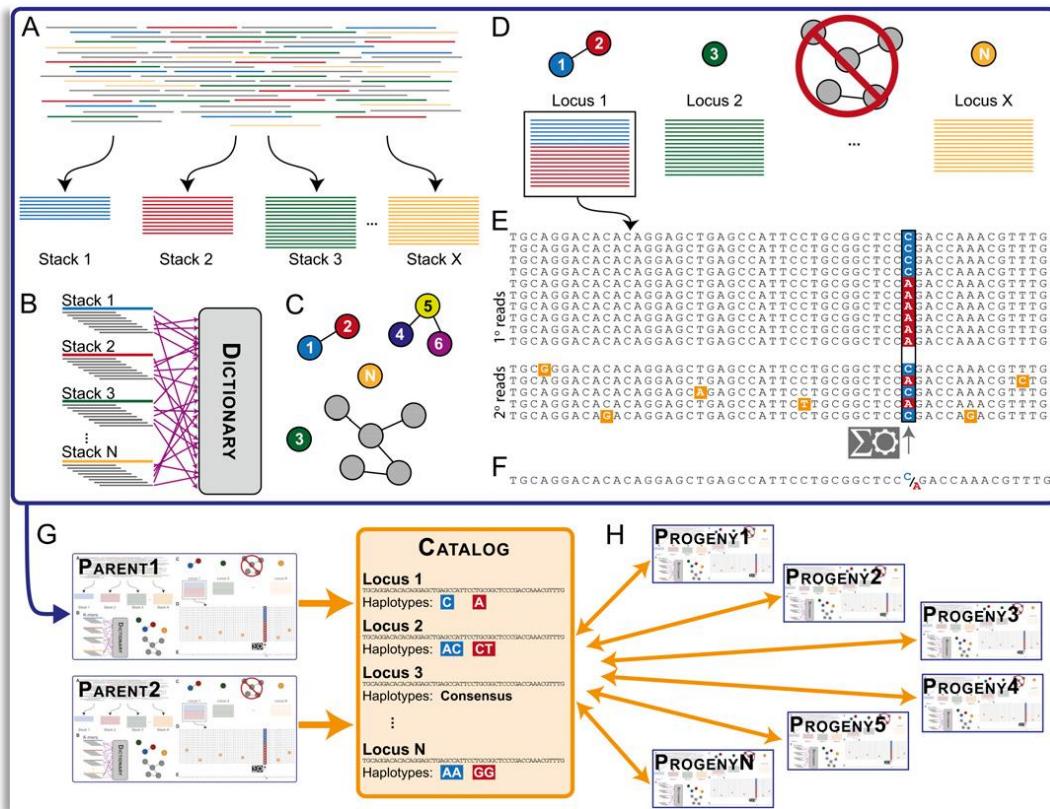
Il y a ainsi deux principaux types d'analyses pouvant être pratiquées :

-Une cartographie génétique. Il faut alors renseigner 2 groupes d'individus : Un a 2 parents d'un croisement génétique et leurs descendants. Dans ce type d'analyse, le catalogue de loci sera constitué à partir des séquences parentales uniquement. A la fin du pipeline, le programme **genotypes** sera exécuté. Cette phase analytique finale permet de mettre en avant les loci cartographiable et d'encoder

le type de chacun de ces loci (ab x aa = un locus a 2 allèles chez un parent et à un allèle chez l'autre). Le programme **genotypes** pourra être ré-exécuté pour obtenir les génotypes dans le cas d'un croisement particuliers (i.e F2 ou backcross) et pour un logiciel de cartographie particuliers (i.e. JoinMap ou OneMap).

-Une analyse de génomique des populations. Les individus sont donc regroupés en populations. Si plusieurs populations sont étudiées, un fichier décrivant le lien entre individus et populations sera utilisé en entré de l'outil. Si aucun fichier de ce type n'est fourni, tous les individus seront assimilé à une seule et même population. A la fin du pipeline, le programme **populations** sera exécuté. Ce programme permet de calculer des statistiques en génomique des populations comme l'hétérozygotie, la diversité nucléotidique Π , l'indice de fixation F_{IS} traduisant un déficit en hétérozygote dans une population lorsque positif ou encore l'indice de différenciation génétique intra ou inter-populations F_{ST} traduisant une forte différence dans les fréquences alléliques quand proche de 1.

Le pipeline sélectionné (**ref-map** ou **denovo_map**) exécutera chaque composant de Stacks.



En premier, il exécutera (A-F) **ustacks** sur chacun des échantillons, construisant les loci et détectant les SNPs à chacun d'entre eux. Les séquences présentant des correspondances exactes sont regroupées en piles ("stacks") (A). Un nombre de lectures trop faible dans une pile pouvant provenir d'erreur de séquençage, les piles uniques contenant moins de lectures que le seuil spécifié (*stack depth parameter*) sont désassemblées et les lectures mises de côté. Les lectures finalement conservées dans une pile sont nommées lectures primaires. Celles mises de côté en raison d'un nombre de lectures trop faible par pile sont nommées lectures secondaires.

A la fin de cette première étape de création de piles, ***ustacks*** calcule la moyenne de profondeur de couverture et identifie les piles présentant une profondeur supérieure à 2 écarts types au-dessus de la moyenne. Toute les piles dans ce cas de figure (nommées piles bucheronnes), ainsi que celles présentant une séquence proche au nucléotide près, sont exclues car souvent représentées par des éléments répétés (voir cas de figure des 5 piles grises reliées (C et D)).

Ensuite, des sous-ensembles de piles sont produits (C), lorsque des piles sont très proches, un nucléotide de différence. Les piles de chaque sous-ensemble peuvent alors être réunies en un seul locus, comme c'est le cas pour les piles 1 et 2 (D). Ensuite, les lectures qui étaient initialement mises de côté, car présentant une similarité trop faible avec les séquences des piles (polymorphisme sur plus d'un nucléotide) (E), sont comparées à celles constitutives des piles pour identifier si elles peuvent être raccrocher à un unique polymorphisme identifié dans les étapes précédentes. Enfin, une séquence consensus est établie (F). En second, (G) ***cstacks*** sera exécuté pour créer un catalogue de tous les loci à partir des parents du croisement. Finalement, (H) ***sstacks*** s'exécutera pour évaluer la concordance entre les loci de chaque descendant et le catalogue de loci. Le pipeline identifie ensuite les loci représentant des marqueurs cartographiables (exécution du module *genotypes*).

Enfin, dans le cadre d'une étude du type cartographie génétique, le programme ***rxstacks*** peut être utilisé afin d'effectuer des corrections sur l'affectation de génotypes et haplotypes en fonction des informations recueillies suite à l'étude d'un ensemble de données générée via l'utilisation du pipeline ***ustacks/ pstacks, cstacks, et sstacks***.

En travaillant sur des données NGS, vous noterez probablement que toutes les données générées par le séquenceur ne sont pas de bonne qualité. En général, il faudra enlever les séquences de faibles qualités de vos jeux de données avant de les analyser. En même temps, la stringence de la filtration dépendra de l'application finale. En général, une stringence plus forte est appliquée pour un assemblage de novo comparé à l'utilisation d'alignements contre un génome de référence. Par contre, des données de mauvaise qualité affecteront presque toujours les analyses futures, en produisant des faux positifs, comme par exemple la prédiction de faux SNP.

I. Analyse RAD-seq sous Galaxy : Détection de SNPs

1. L'analyse

Nous travaillerons sur les données issues de l'archive stacks_samples.tar.gz disponible sur le site de Stacks (http://catchenlab.life.illinois.edu/stacks/tut_gar.php). Sous Galaxy, commencez par créer un nouvel historique (ex: "RAD 1 : SNP calling").

Créer un historique et le nommer (« STACKS 1.40 GCC Training RAD 1 : SNP calling » par exemple).

Récupération des données brutes dans Shared data/data libraries/RADseq/genetic_map/

The screenshot shows the Galaxy Data Libraries interface. At the top, there are tabs for Analyze Data, Workflow, Shared Data, Visualization, Admin, Help, User, and a search bar. Below the tabs, it says "Using 48%". The main area is titled "DATA LIBRARIES" and shows a list of 15 items out of 97. The list includes: "female" (fasta, 3.0 MB, updated 2016-06-18 04:56 PM), "male" (fasta, 3.4 MB, updated 2016-06-18 04:56 PM), "Parents.zip" (zip, 526.0 KB, updated 2016-06-18 04:20 PM), and 18 entries for "progeny_1" through "progeny_19" (all fasta files between 884.8 KB and 1.7 MB in size, all updated 2016-06-18 04:56 PM). Each item has a "Details" button to its right.

Vous y trouverez les 2 jeux de données de séquences des parents d'un protocole de cartographie génétique auquel est associé les jeux de données de 93 descendants, que nous n'utiliserons pas ici.

Sélectionner les 2 jeux de données (**male** et **female**), et cliquer sur le bouton "**TO HISTORY**" pour sélectionner l'historique créé.

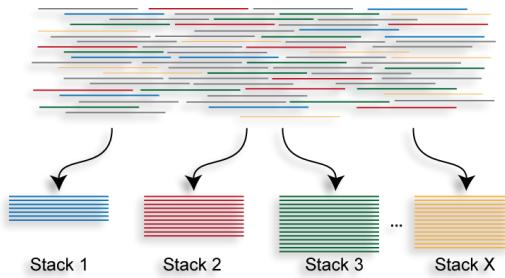
Vous vous retrouvez avec un historique contenant les deux jeux de données au format fasta.

Les données ayant déjà été nettoyées et démultiplexées, il ne sera **pas nécessaire** d'exécuter **STACKS : Process Radtags**.

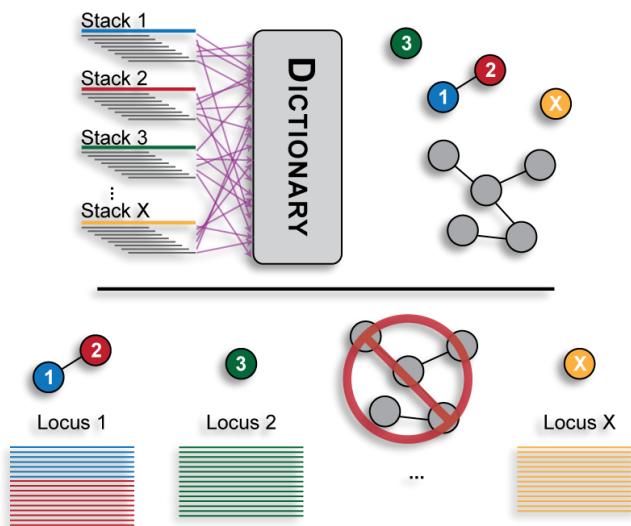
Exécuter donc directement **STACKS : ustacks** sur les individus étudiés (ici les deux parents en sélectionnant les deux fichiers téléchargés comme données d'entrée. Il faut renseigner le paramètre

« unique numeric ID », si vous n'êtes pas inspiré, entrer « test » par exemple. Nous pouvons spécifier certains paramètres :

-la profondeur minimum de la pile (-m), ici "*Minimum depth of coverage required to create a stack*". Ce paramètre, passé à **ustacks**, contrôle le nombre de lectures avec une **correspondance exacte** devant être trouvé pour créer une pile chez un individu. Nous pouvons sélectionner **3**, une pile ne sera générée chez un individu que si au moins 3 lectures correspondent exactement. Les lectures alors utilisées pour constituer ces piles sont qualifiées de **lectures primaires**. **Les autres**, qui ne correspondent donc pas exactement à des lectures primaires et qui n'étaient pas assez nombreuses pour générer une pile, sont qualifiées de **lectures secondaires**. Ce paramètre est également nommé "*stack-depth*".



-la **distance maximum permise entre des piles** pour qu'elles soient **fusionnées en un locus** potentiel **chez un individu**. Il s'agit du paramètre **-M**, ici "*Maximum distance (in nucleotides) allowed between stacks*". Nous préciserons ici une distance maximale de **2** nucléotides. En reprenant le schéma ci-dessous, les piles 1 et 2 vont être fusionnées car représentent un seul locus polymorphe (avec 2 allèles différents possédant une différence maximale correspondante au seuil fixé, ici 2), les piles 3 et X vont être associés à deux loci monomorphiques distincts, le gros paquet gris de piles représente un jeu de séquences répétées qui présentent trop d'allèles pour être biologiquement corrects. Ce paramètre est également nommé "*within-individual distance*". Valeurs préconisées : entre 2 et 4.



Les autres paramètres peuvent être laissés avec les valeurs par défaut. Temps d'exécution < 10sec.

The screenshot shows the Galaxy web interface with the Stacks ustacks tool selected. The tool configuration includes:

- Input short reads from an individual:** 2: male, 1: female
- Give a unique numeric ID to this sample:** 0
- Minimum depth of coverage required to create a stack:** 3
- Maximum distance (in nucleotides) allowed between stacks:** 2
- Retain unused reads:** Yes
- Disable calling haplotypes from secondary reads:** Yes
- SNP Model Options (ustacks options):**
 - Enable the Removal algorithm, to drop highly-repetitive stacks (and nearby errors) from the algorithm: Yes
 - Enable the Deleveraging algorithm, used for resolving over merged tags: Yes
 - Maximum number of stacks at a single de novo locus: 3
 - (-max_locus_stacks)
 - K-mer size for matching between alleles and loci (automatically calculated by default): (empty)
 - (-k_len)
- Perform gapped alignments between stacks:** Yes

The History panel shows two datasets generated by the tool:

- 2: male:** 33,021 sequences, format: fasta, database: ?
- 1: female:** 29,517 sequences, format: fasta, database: ?

Both datasets contain sequence data in FASTA format, showing DNA sequences with their corresponding IDs.

Nous obtenons ici 2 « data collection », une par jeu de données d'entrée.

Chaque « data collection » contient 4 types de fichiers, les fichiers **.tags**, **.snps**, **.alleles** et **.models**

2. Les fichiers de sortie

Soit le pipeline lancé sur deux individus, "male" et "female". Dans les fichiers générés, nous retrouvons:

a) Les fichiers tags

XXX.tags.tsv file:

Column	Name	Description
1	Sql ID	This field will always be "0", however the MySQL database will assign an ID when it is loaded.
2	Sample ID	Each sample passed through Stacks gets a unique id for that sample.
3	Stack ID	Each stack formed gets an ID.
4	Chromosome	If aligned to a reference genome using pstacks, otherwise it is blank.
5	Basepair	If aligned to ref genome using pstacks.
6	Strand	If aligned to ref genome using pstacks.
7	Sequence Type	Either 'consensus', 'primary' or 'secondary', see the Stacks paper for definitions of these terms.
8	Sequence ID	The individual sequence read that was merged into this stack.
9	Sequence	The raw sequencing read.
10	Deleveraged Flag	If "1", this stack was processed by the deleveraging algorithm & was broken down from a larger stack.
11	Blacklisted Flag	If "1", this stack was still confounded despite processing by the deleveraging algorithm.
12	Lumberja ckstack F	If "1", this stack was set aside due to having an extreme depth of coverage.

Notes: Chaque "stack" commencera avec une séquence consensus suivie par les flags de ce "stack". Puis suivent chaque lecture individuelle qui a été fusionnée dans ce "stack". Le prochain "stack" débutera avec une autre séquence consensus.

Exemple du *female.tags* :

0	1	1	primary	1	CAGTC_2_0076_1442_1576_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0079_1540_762_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0080_1139_865_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0081_458_1052_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0082_1542_1441_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0083_1732_263_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	primary	1	CAGTC_2_0087_697_1865_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	secondary	1	CAGTC_2_0088_1602_271_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	secondary	1	CAGTC_2_0093_92_1898_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	secondary	1	CAGTC_2_0098_1624_135_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	secondary	1	CAGTC_2_0091_1720_1032_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG
0	1	1	secondary	1	CTAGG_2_0038_1473_926_1[79984]	TGCAGGTACATCAATCAATCGGACTACATCTGAACCACCTGATCCAACAAAACATGTGTTTGTCTGCACGG

b) Les fichiers snps

Column	Name	Description
1	Sql ID	This field will always be "0", however the MySQL database will assign an ID when it is loaded.
2	Sample ID	
3	Stack ID	
4	SNP Column	
5	Likelihood ratio	From the SNP-calling model.
6	Rank_1	Majority nucleotide.
7	Rank_2	Alternative nucleotide.

Exemple du *female.snps* :

ustacks version 1.40; generated on 2016-06-20 14:07:28
0 0 1 0 O 73.47 T -
0 0 1 1 O 73.47 G -
0 0 1 2 O 73.47 C -
0 0 1 3 O 73.47 A -
0 0 1 4 O 73.47 G -
0 0 1 5 O 73.47 G -
0 0 1 6 O 73.47 C -
0 0 1 7 O 73.47 T -
0 0 1 8 O 73.47 A -
0 0 1 9 O 73.47 A -
0 0 1 10 O 73.47 C -
0 0 1 11 O 73.47 A -
0 0 1 12 O 73.47 T -
0 0 1 13 O 73.47 G -
0 0 1 14 O 73.47 C -
0 0 1 15 O 73.47 A -
0 0 1 16 O 73.47 G -
0 0 1 17 O 61.35 G -
0 0 1 18 O 73.47 A -
0 0 1 19 O 73.47 G -
0 0 1 20 O 73.47 G -
0 0 1 21 O 73.47 A -
0 0 1 22 O 61.35 C -
0 0 1 23 O 73.47 A -
0 0 1 24 O 73.47 G -
0 0 1 25 O 73.47 A -

Notes: Si un "stack" a présenté 2 SNPs, il y aura 2 lignes dans le fichier.

c) Les fichiers alleles

Column	Name	Description
1	Sql ID	This field will always be "0", however the MySQL database will assign an ID when it is loaded.
2	Sample ID	
3	Stack ID	
4	Haplotype	The haplotype, as constructed from the called SNPs at each locus.
5	Percent	Percentage of reads that have this haplotype
6	Count	Raw number of reads that have this haplotype

Exemple du *female.alleles* :

# ustacks version 1.40; generated on 2016-06-20 14:07:28					
0	0	2	C	48.05	37
0	0	2	T	51.95	40
0	0	7	A	52.00	13
0	0	7	G	48.00	12
0	0	8	A	40.74	22
0	0	8	G	59.26	32
0	0	13	G	56.25	36
0	0	13	T	43.75	28
0	0	14	C	48.39	75
0	0	14	T	51.61	80
0	0	18	A	46.77	29
0	0	18	T	53.23	33
0	0	19	A	64.52	60
0	0	19	G	35.48	33
0	0	24	G	62.50	35
0	0	24	T	37.50	21
0	0	25	A	66.67	12
0	0	25	G	33.33	6
0	0	26	A	47.69	31
0	0	26	C	52.31	34
0	0	27	C	66.67	16
0	0	27	T	33.33	8
0	0	29	C	53.33	40
0	0	29	T	46.67	35

d) Les fichiers models

Il s'agit de fichiers reprenant le format des fichiers **tags** en ne représentant que, par **stack** la séquence consensus et le modèle :

e)	Column	Name	Description
f)	1	Sql ID	This field will always be "0", however the MySQL database will assign an ID when it is loaded.
g)	2	Sample ID	Each sample passed through Stacks gets a unique id for that sample.
h)	3	Stack ID	Each stack formed gets an ID.
i)	4	Chromosome	If aligned to a reference genome using pstacks, otherwise it is blank.
j)	5	Basepair	If aligned to ref genome using pstacks.
k)	6	Strand	If aligned to ref genome using pstacks.
l)	7	Sequence Type	Either 'consensus', 'primary' or 'secondary', see the Stacks paper for definitions of these terms.
m)	8	Sequence ID	The individual sequence read that was merged into this stack.
n)	9	Sequence	The raw sequencing read.
o)	10	Deleveraged Flag	If "1", this stack was processed by the deleveraging algorithm & was broken down from a larger stack.
p)	11	Blacklisted Flag	If "1", this stack was still confounded despite processing by the deleveraging algorithm.
q)	12	Lumberjack stack F	If "1", this stack was set aside due to having an extreme depth of coverage.

Exemple du *female.models* :

# ustacks version 1.40; generated on 2016-06-20 14:07:28					
0	0	1	0	+	consensus
0	0	1	model	TGCAGGCTAACATGCGAGGAGACAGGCAACACATACTGTGCAATACAGTAGTGTGAGTTGATTATTCA	0 0 0 -20.6215
0	0	2	0	+	consensus
0	0	2	model	TGCAGGTACATCATCATGACTCATCTGAACCACCTGATCAAACAAACTATGTGTTTGATGTCATG	0 0 0 -12.92
0	0	3	0	+	consensus
0	0	3	model	TGCAGGGCGTCTAGACCTCCCTGTCTAGGCCAGAAGATCATGAAATAATGTCCTTGATTTGGCCAAA	0 0 0 -16.70
0	0	4	0	+	consensus
0	0	4	model	TGCAGGGACTTGGCCAGGGAACTTACCTGCAACTGTGACCTCCACTCTGCTGGGGSCGGACGTGGACGG	0 0 0 -2.08
0	0	5	0	+	consensus
0	0	5	model	TGCAGGCCAACATCGATGAAGACGTCAAAGTGGAGGCCAGGCAAGGGTGTCTGGTCCCAGTCAGT	0 0 0 -15.09
0	0	6	0	+	consensus
0	0	6	model	TGCAGGCCCTCTGAGCTGCTGACTGTTTAATGAGATCACATTAACTTTAAGCCTTGTCTTCT	0 0 0 -16.04
0	0	7	0	+	consensus
0	0	7	model	TGCAGGGAGGTGACGCCAGATGACCCCCACCTACTGACTCTCTGTCACAGTGTGAGCCCACACAGCA	0 0 0 -8.10
0	0	8	0	+	consensus
0	0	8	model	TGCAGGCCAGGAAACCGTCAACCCATGTCTCCCTGTGACAGAAGACTAACCTGAGGTGTCTGTTGCA	0 0 0 -21.25
0	0	9	0	+	consensus
0	0	9	model	TGCAGGGAGCCACAGTGGAACACGACACAGAAGGGAGAGTAAGGGGGGGTGTGATTGAAACCAAGACTGGAG	0 0 0 -21.66
0	0	10	0	+	consensus
0	0	10	model	TGCAGGGAAACACAATCCAGACAAACTGGAGGGATCAGAGAGTCTGCTGTAGAAAGTGTGAGACAGGGAAAGA	0 0 0 -18.82
0	0	11	0	+	consensus
0	0	11	model	TGCAGGCCCTGCGTCCAGCCAGCTACACCGATGTCATGAAAGAATSCGGTTAGTCTGGCTCAGCTAGAGA	0 0 0 -6.25
0	0	12	0	+	consensus
0	0	12	model	TGCAGGCCCTGGGGCTCAAATTTTTAATTGTCAGCAGACAAGGTAGCTGCTCTGCTTAGGAGAAA	0 0 0 -15.30

Nous avons détecté des SNPs chez nos 2 individus et nous pouvons déterminer lesquels sont situés sur les mêmes loci.

3. Exercices

Dans les fichiers models, quels codes sont utilisés et que signifient-ils ?

**Filtrer les likelihood ratio (colonne 6) sur fichiers .snps en ne retenant que les scores < 0.
Plus le score est négatif, plus il est significatif, plus le snp est strong.**

II. Analyse RAD-seq sous Galaxy : La cartographie génétique

Cet exercice utilise des données générées par les développeurs de STACKS. Ils ont développé une carte génétique pour le Lépisosté tacheté (*Lepisosteus oculatus*) (poisson crocodile parfois surnommé brochet crocodile) et présentent ici les données d'un seul groupe de liaison. La carte génétique provient d'un croisement de type pseudo-test cross F1 entre 2 parents et 93 de leur descendants F1. Sont conservés pour l'exercice, uniquement des marqueurs apparaissant dans un unique groupe de liaison, et les lectures brutes des "stacks" qui ont contribué à ce groupe de liaison. Les fichiers sont déjà nettoyés, il ne sera donc pas nécessaire d'utiliser l'outil "[STACKS : Process radtags Run the STACKS cleaning script](#)" ici. Il est également possible d'utiliser un génome de référence en récupérant par exemple les séquences présentes sur le serveur ftp d'ensembl : ftp://ftp.ensembl.org/pub/release-84/fasta/lepisosteus_oculatus/dna/ présentant les 29 groupes de liaison (ftp://ftp.ensembl.org/pub/release-84/fasta/lepisosteus_oculatus/dna/Lepisosteus_oculatus.LepOcu1.dna.chromosome.LG1.fa.gz à ftp://ftp.ensembl.org/pub/release-84/fasta/lepisosteus_oculatus/dna/Lepisosteus_oculatus.LepOcu1.dna.chromosome.LG29.fa.gz)

Les 95 fichiers fasta de départ sont donc déjà démultiplexés et se présentent comme suit :

```
>TTAAT_1_0046_17989_1193_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0092_18487_11460_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0094_8623_4235_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0102_18666_9095_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0114_6838_19507_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0117_2046_2348_1[67302]
TGCAGGGAGGAAGTCACAGAGATCCCTGGCCAACACTGTAGTTCGAACAGGAACCGAGCTGACAGGGCGCAGA
>TTAAT_1_0038_16287_13120_1[22594]
TGCAGGCCTTGTGAAACTGAACACACAAAAGGTTCTATCAATTAAAACCGCAGATAATTAGTTGTGTTCTCCA
>TTAAT_1_0052_11753_10140_1[22594]
TGCAGGCCTTGTGAAACTGAACACACAAAAGGTTCTATCAATTAAAACCGCAGATAATTAGTTGTGTTCTCCA
>TTAAT_1_0054_1743_17715_1[22594]
TGCAGGCCTTGTGAAACTGAACACACAAAAGGTTCTATCAATTAAAACCGCAGATAATTAGTTGTGTTCTCCA
>TTAAT_1_0074_2389_17780_1[22594]
TGCAGGCCTTGTGAAACTGAACACACAAAAGGTTCTATCAATTAAAACCGCAGATAATTAGTTGTGTTCTCCA
>TTAAT_1_0087_18378_6512_1[22594]
```

Pour chaque séquence, on retrouve le barcode (ici TTAAT), et d'autres informations.

Créer un nouvel historique et renommer le (ex : STACKS 1.40 GCC Training RAD 2: Genetic map).

Comme les données une partie des données sont les mêmes qu'utilisées dans la partie précédente, vous pouvez les rapatrier en utilisant l'option "copy datasets" de l'historique (URL d'origine : http://creskolab.uoregon.edu/stacks/tut_gar.php, les données : http://creskolab.uoregon.edu/stacks/tutorial/stacks_samples.tar.gz). Vous récupérer ainsi 2 jeux de données. Pour les autres, rendez-vous dans Shared data/data libraries/RADseq/genetic_map/ et sélectionner l'ensemble des fichiers des 93 descendants (pas d'individu progeny 69 !).

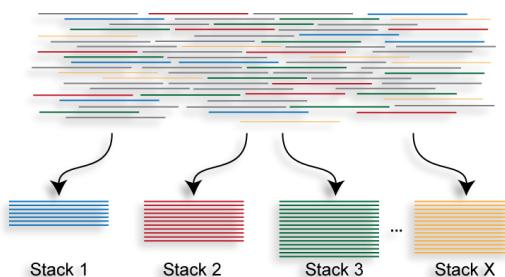
Nous sommes maintenant près à exécuter le pipeline ***STACKS denovo_map***.

Sélectionner l'outil "[Stacks: de novo map](#)" the Stacks pipeline without a reference genome (denovo_map.pl)"

Vous pouvez préciser certains paramètres en sélectionnant le mode Advanced pour le paramètre ***Stack assembly options***.

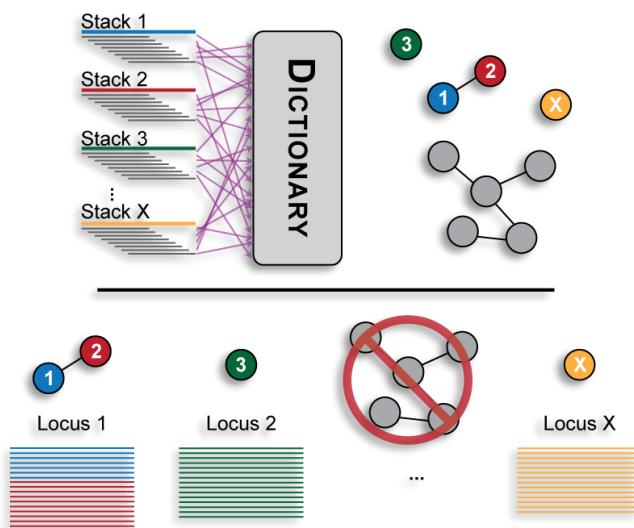
Vous pouvez préciser certains paramètres. Ainsi, en sélectionnant le mode Advanced pour le paramètre ***Stack assembly options***, vous pouvez préciser:

-la profondeur minimum de la pile (-m), ici "*Minimum number of identical raw reads required to create a stack*". Ce paramètre, passé à ***ustacks***, contrôle le nombre de lectures avec une **correspondance exacte** devant être trouvé pour créer une pile chez un individu. Nous pouvons sélectionner **3**, une pile ne sera générée chez un individu que si au moins 3 lectures correspondent exactement. Les lectures alors utilisées pour constituer ces piles sont qualifiées de **lectures primaires**. **Les autres**, qui ne correspondent donc pas exactement à des lectures primaires et qui n'étaient pas assez nombreuses pour générer une pile, sont qualifiées de **lectures secondaires**. Ce paramètre est également nommé "*stack-depth*".



-la profondeur minimum de la pile (-P), option utilisable qu'en cas de jeux de données de type descendant, est représentée ici sous l'option "*Minimum number of identical raw reads required to create a stack (progeny)*". Ce paramètre, passé à **ustacks**, contrôle le **nombre de lectures correspondant exactement devant être trouvé pour créer une pile chez un descendant**. Nous pouvons sélectionner **3**, une pile ne sera générée chez un descendant que si au moins 3 lectures correspondent exactement.

-la **distance maximum permise entre des piles** pour qu'elles soient **fusionnées en un locus** potentiel **chez un individu**. Il s'agit du paramètre **-M**, ici "*Number of mismatches allowed between loci when processing a single individual*". Nous préciserons ici une distance maximale de **2** nucléotides. En reprenant le schéma ci-dessous, les piles 1 et 2 vont être fusionnées car représentent un seul locus polymorphique (avec 2 allèles différents possédant une différence maximale correspondante au seuil fixé, ici 2), les piles 3 et X vont être associés à deux loci monomorphiques distincts, le gros paquet gris de piles représente un jeu de séquences répétées qui présentent trop d'allèles pour être biologiquement corrects. Ce paramètre est également nommé "*within-individual distance*". Valeurs préconisées : entre 2 et 4.



-le nombre maximal de différence entre loci du catalogue. Il s'agit du paramètre **-n**, "*specify the number of mismatches allowed between loci when building the catalog*" ici. Ce paramètre permet notamment de pouvoir **créer un locus de type homozygote dans le catalogue** alors qu'il est en réalité hétérozygote. Ainsi, si par exemple on fixe la valeur à 2 et que la séquence du locus ne présente pas de polymorphisme au sein de chaque individu mais présente plus de 2 nucléotides de différences entre chaque individu, alors chaque "version" du locus identifié générera un locus distinct des autres. Ce paramètre est notamment pratique pour **conserver les loci hétérozygotes entre individus mais homozygote chez chacun d'entre eux**. Si la valeur de ce paramètre est trop importante, cstacks risque de considérer des loci distincts en un seul locus. Si cette valeur est trop faible, cstacks générera potentiellement plus de loci qu'il n'y en a réellement. Nous pouvons fixer ce paramètre à **3**. Ce paramètre est également nommé "*between-individual distance*".

-supprimer ou casser les RAD-tags très répétitifs. Il s'agit de **1/supprimer les piles "bûcheronnes"** (**lumberjack**) **extrêmement surdimensionnées** (plusieurs dizaines à centaines de milliers de lectures), traduisant souvent l'existence de régions répétées dans le génome, de l'analyse, **2/briser les piles**

modérément surdimensionnées (couverture en lectures par pile > moyenne des couvertures par pile + 2 x écart-type des couvertures par pile) et **3/**ne pas considérer les loci obtenus à partir de la fusion d'un nombre de "stacks" supérieur à 3. Nous cocherons ici cette option, correspondant au paramètre **-t**. Quand une pile présente une couverture supérieure de plus de 2 écart-types de la couverture moyenne, elle, et les piles proches à 1 bp de cette pile, seront mises de côté et signalées comme "**Deleveraged**" grâce à l'ajout d'un "**1**" dans la 10^{ème} colonne des fichiers ***.tags.tsv**. L'application de ce "**Deleveraging algorithm**" permet également de ne pas considérer les loci obtenus à partir de la fusion d'un nombre de "stacks" supérieur à 3 pour construire le catalogue. Enfin, les piles bûcheronnes seront mises de côté, non considérées lors de la constitution du catalogue, et signalée grâce à l'ajout d'un "**1**" dans la dernière colonne des fichiers ***.tags.tsv**.

D'une manière générale, utiliser des options plus stringentes (comme la suppression des RAD-tags très répétitifs) va générer un nombre total de piles plus faible, augmentera le nombre de piles réunis (car manque des piles à n nucléotides près faisant le pont entre deux piles) et diminuera le nombre final de loci du catalogue. (car moins de similarité, plus de bruit de fond, entre les piles des individus).

Temps d'exécution < 2 minutes

Dans le cas présent, 5 « data collection » et 8 jeux de données sont générés dans l'historique :

The screenshot shows the Galaxy web interface. On the left, the 'Tools' sidebar lists various Stacks-related tools: stacks, STACKS test: v132, NGS: Building Loci, Stacks 1.40, Stacks: cstacks build a catalogue of loci, Stacks: ustacks align short reads into stacks, Stacks: process_radtags the Stacks demultiplexing script, Stacks: de novo map the Stacks pipeline without a reference genome (denovo_map.pl), Stacks: assemble read_pairs_by_locus run the STACKS sort_read_pairs.pl and exec_velvet.pl wrappers, Stacks: pstacks find stacks from short reads mapped to a reference genome, Stacks: populations analyze a population of individual samples ('populations' program), Stacks: sstacks match stacks to a catalog, Stacks: reference map the Stacks pipeline with a reference genome (ref_map.pl), Stacks: rxstacks make corrections to genotype and haplotype calls, and Stacks: genotypes analyse haplotypes or genotypes in a genetic cross ('genotypes' program). Below this is the 'Workflows' section with a link to 'All workflows'. The main content area displays a green banner stating 'GenOuest Galaxy server is running' and a blue info box with contact information. The title 'GenOuest Bioinformatics platform' is centered above a horizontal line. Below the line, the text 'Development, expertise and resources for bioinformatics' is followed by several logos: IRISA, Inria, Biogenoest, and Bretagne. A blue info box provides citation information for the instance. The right side of the screen shows the 'History' panel, which lists a series of dataset runs, each with a preview icon, edit icon, and delete icon. The runs include: 242: Haplotypes table (Stacks SQL format) with Stacks: de novo map on data 95, data 94, and others; 241: Markers table (Stacks SQL format) with Stacks: de novo map on data 95, data 94, and others; 240: Haplotypes table (generic format) with Stacks: de novo map on data 95, data 94, and others; 239: Haplotypes table (JoinMap format) with Stacks: de novo map on data 95, data 94, and others; 238: Catalog haplotypes (alleles) with Stacks: de novo map on data 95, data 94, and others; 237: Catalog model calls (snps) with Stacks: de novo map on data 95, data 94, and others; 236: Catalog assembled loci (tags) with Stacks: de novo map on data 95, data 94, and others; 235: denovo_map.log with Stacks: de novo map on data 95, data 94, and others; 234: Full output from denovo_map on data 95, data 94, and others (a list of datasets); 233: Matches to the catalog on data 95, data 94, and others (a list of datasets); 232: Haplotypes/alleles recorded from each locus on data 95, data 94, and others (a list of datasets); 231: Model calls from each locus on data 95, data 94, and others (a list of datasets); 230: Assembled loci from data 95, data 94, and others (a list of datasets); 95: progeny_94; and 94: progeny_93.

1. Un premier jeu de données nommé denovo_map.log, permettant de vérifier le bon déroulement du job.

```
denovo_map.pl version 1.40 started at 2016-06-20 14:53:14
/home/genouest/admin/galaxy/dependencies/stacks/1.40/iuc/package_stacks_1_40/5eac5e2d91e2/bin/denovo_map.pl
```

Nous pouvons voir le script lancé (ici denovo_map.pl), ainsi que la date et l'heure de départ du job. Ensuite, vient la ligne de commande lancée, /home/genouest/admin/galaxy/dependencies/stacks/1.40/...../bin/denovo_map.pl -p...

S'en suit le listing des étapes effectuées fichier par fichier :

```

Identifying unique stacks; file 1 of 95 [female]
/home/genouest/admin/galaxy/dependencies/stacks/1.40/iuc/package_stacks_1_40/5eac5e2d91e2/bin/ustacks -t fasta -f female.fa -o stacks_outputs
ustacks parameters selected:
  Min depth of coverage to create a stack: 3
  Max distance allowed between stacks: 2
  Max distance allowed to align secondary reads: 4
  Max number of stacks allowed per de novo locus: 3
  Deleveraging algorithm: enabled
  Removal algorithm: enabled
  Model type: SNP
  Alpha significance level for model: 0.05
  Gapped alignments: disabled
Parsing female.fa
Loaded 29517 RAD-Tags; inserted 4922 elements into the RAD-Tags hash map.
  0 reads contained uncalled nucleotides that were modified.
Initial coverage mean: 35.1762; Std Dev: 22.8259; Max: 150
Deleveraging trigger: 58; Removal trigger: 81
715 initial stacks were populated; 4207 stacks were set aside as secondary reads.
Calculating distance for removing repetitive stacks.
  Distance allowed between stacks: 1; searching with a k-mer length of 37 (39 k-mers per read); 2 k-mer hits required.
Removing repetitive stacks.
  Removed 36 stacks.
  707 stacks remain for merging.
Post-Repeat Removal, coverage depth Mean: 32.5538; Std Dev: 25.0232; Max: 81
Calculating distance between stacks...
  Distance allowed between stacks: 2; searching with a k-mer length of 25 (51 k-mers per read); 1 k-mer hits required.
Merging stacks, maximum allowed distance: 2 nucleotide(s)
  707 stacks merged into 453 stacks; deleveraged 3 stacks; removed 0 stacks.
After merging, coverage depth Mean: 52.0094; Std Dev: 30.4338; Max: 143
Merging remainder radtags
  4366 remainder sequences left to merge.
  Distance allowed between stacks: 4; searching with a k-mer length of 15 (61 k-mers per read); 1 k-mer hits required.
  Matched 3911 remainder reads; unable to match 455 remainder reads.
After remainders merged, coverage depth Mean: 61.2118; Std Dev: 32.5155; Max: 165
Calling final consensus sequences, invoking SNP-calling model...
Number of utilized reads: 29062
Writing loci, SNPs, and alleles to 'stacks_outputs/...
  Refetching sequencing IDs from female.fa... read 29517 sequence IDs.
done.

```

Nous obtenons ainsi pour le premier des 95 fichiers, le parent female :

-la ligne de commande correspondant au lancement de **ustacks**

-les paramètres renseignés lors de la soumission du job (Min depth of coverage,)

-le nombre total de lectures, qualifiées de RAD-tags, chargées (ici 29517) ainsi que le nombre d'éléments insérés dans la carte de hachage (le "dictionary" mentionné dans l'étape B du schéma présent en début de ce document) de RAD-tags (ici 4922). La couverture moyenne par RAD-tag (35.1762) avec l'écart-type (22.8259) associé et la profondeur maximum observées (150). L'activation de l' option "*supprimer ou casser les RAD-tags très répétitifs*" (-t) active l'option "*Deleveraging algorithm*". On peut voir que c'est le cas ici via la ligne "*Deleveraging algorithm: enable*". Quand une pile présente une couverture supérieure de plus de 2 écart-types de la couverture moyenne (ici > 80.828), elle, et les piles proches à 1 bp de cette pile, seront mises de côté et signalées comme "**Deleveraged**" grâce à l'ajout d'un "1" dans la 10^{ème} colonne des fichiers ***.tags.tsv**. Ces piles sont donc cassées. L'application de ce "**Deleveraging algorithm**" permet également de ne pas considérer les loci obtenus à partir de la fusion d'un nombre de "stacks" supérieur à 3 pour construire le catalogue. Pourquoi 3? Pour éviter de supprimer un locus pour lequel la majorité des individus présentaient au maximum 2 piles différentes (2 allèles) et seuls quelques-uns 3, à cause d'erreur de séquençage par exemple. Les piles "bûcheronnes" (présentant une couverture de plusieurs dizaines à centaines de milliers de lectures) seront elles supprimées (c'est ici le cas comme l'indique la ligne "*Removal algorithm: enable*"), mises de côté, non considérées lors de la constitution du catalogue, et signalées grâce à l'ajout d'un "1" dans la dernière colonne des fichiers ***.tags**.

-le nombre de lectures avec des nucléotides modifiés

-la profondeur de couverture moyenne

-le nombre de stacks répétés supprimés (ici 36)

-le nombre de stacks fusionnés car proches (ici 707 fusionnés en 453).

-le nombre de lectures restantes concordant avec les 453 stacks préalablement générés. Ici 4366.

Sur les 29517 lectures d'origines "RAD tagguées", seules 455 n'ont pas été utilisées. Il y a donc 29062 lectures finalement utilisées pour générer 453 stacks. Ustacks est ensuite exécuté sur les 94 autres individus.

Ensuite est exécuté **cstacks**.

```
/home/genouest/admin/galaxy/dependencies/stacks/1.40/iuc/package_stacks_1_40/5eac5e2d91e2/bin/cstacks -b 1 -o stacks_outputs

Number of mismatches allowed between stacks: 3
Loci matched based on sequence identity.
Constructing catalog from 2 samples.
Initializing new catalog...
  Parsing stacks_outputs/female.tags.tsv
  Parsing stacks_outputs/female.snps.tsv
  Parsing stacks_outputs/female.alleles.tsv
  425 loci were newly added to the catalog.
Processing sample stacks_outputs/female [2 of 2]
  Parsing stacks_outputs/male.tags.tsv
  Parsing stacks_outputs/male.snps.tsv
  Parsing stacks_outputs/male.alleles.tsv
Searching for sequence matches...
  Distance allowed between stacks: 3; searching with a k-mer length of 17 (59 k-mers per read); 8 k-mer hits required.
  425 loci in the catalog, 28578 kmers in the catalog hash.
Merging matches into catalog...
  390 loci were matched to a catalog locus.
  2 loci were matched to a catalog locus using gapped alignments.
  34 loci were newly added to the catalog.
  0 loci matched more than one catalog locus and were excluded.
Writing catalog to 'stacks_outputs/... done.
```

Là encore, un rappel des paramètres utilisés est fait. Le catalogue de loci est ensuite créé à partir des échantillons parentaux.

Pouvez-vous identifier à quoi correspond le nombre de loci de 425 ? En parcourant le fichier catalog.tags, identifiez les loci spécifiques à chacun des individus et ceux partagés. A partir du fichier, compter le nombre de loci du catalogue provenant de l'individu 1, de l'individu 2, ceux retrouvés chez les 2 parents.

Ici, le catalogue créé contient 459 tags (voir *catalog.tags*) dont 425 provenant de l'"individu de référence" (female) et pouvant être partagé avec le second individu (male), 424 provenant du second individu (male) et pouvant être partagé avec l'"individu de référence" (female).

Enfin, **sstacks** est exécuté sur chacun des 95 individus:

```

/home/genouest/admin/galaxy/dependencies/stacks/1.40/iuc/package_stacks_1_40/5eac5e2d91e2/bin/sstacks
Searching for matches by sequence identity...
  Parsing stacks_outputs/batch_1.catalog.tags.tsv
  Parsing stacks_outputs/batch_1.catalog.snps.tsv
  Parsing stacks_outputs/batch_1.catalog.alleles.tsv
Processing sample 'stacks_outputs/female' [1 of 95]
  Parsing stacks_outputs/female.tags.tsv
  Parsing stacks_outputs/female.snps.tsv
  Parsing stacks_outputs/female.alleles.tsv
Searching for sequence matches...
425 stacks compared against the catalog containing 459 loci.
  425 matching loci, 0 contained no verified haplotypes.
  0 loci matched more than one catalog locus and were excluded.
  0 loci contained SNPs unaccounted for in the catalog and were excluded.
  649 total haplotypes examined from matching loci, 649 verified.
Outputting to file stacks_outputs/female.matches.tsv
Processing sample 'stacks_outputs/male' [2 of 95]
  Parsing stacks_outputs/male.tags.tsv
  Parsing stacks_outputs/male.snps.tsv
  Parsing stacks_outputs/male.alleles.tsv
Searching for sequence matches...
426 stacks compared against the catalog containing 459 loci.
  426 matching loci, 0 contained no verified haplotypes.
  0 loci matched more than one catalog locus and were excluded.
  0 loci contained SNPs unaccounted for in the catalog and were excluded.
  692 total haplotypes examined from matching loci, 692 verified.
Outputting to file stacks_outputs/male.matches.tsv
Processing sample 'stacks_outputs/progeny_1' [3 of 95]
  Parsing stacks_outputs/progeny_1.tags.tsv
  Parsing stacks_outputs/progeny_1.snps.tsv
  Parsing stacks_outputs/progeny_1.alleles.tsv
Searching for sequence matches...
358 stacks compared against the catalog containing 459 loci.
  353 matching loci, 1 contained no verified haplotypes.
  1 loci matched more than one catalog locus and were excluded.
  0 loci contained SNPs unaccounted for in the catalog and were excluded.
  478 total haplotypes examined from matching loci, 476 verified.
Outputting to file stacks_outputs/progeny_1.matches.tsv

```

Pour chaque individu, **sstacks** détermine l'haplotype à chaque locus de chacun des individus du croisement. Il va alors comparer les stacks de chaque individu avec le catalogue de 459 loci. Ici, 649 haplotypes sont trouvés au total, 649 vérifiés. Quand sstacks "vérifie" un match, il vérifie en fait que : 1) le locus "query" ne match que à un seul locus du catalogue, 2) ce locus ne porte pas de SNPs non présents dans le catalogue 3) au moins un des haplotypes du catalogue correspond exactement à un haplotype du locus "query". Il est alors possible de ne pas avoir certains haplotypes vérifiés en raison d'une couverture trop faible d'un locus pour détecter un locus et/ou un seuil statistique utilisé pour le "SNP model" trop stringent.

Enfin, **genotypes** est exécuté. Il récupère les loci contenant les marqueurs identifiés chez les parents, puis y associe les haplotypes des descendants. Si le premier parent présente les haplotypes *GA* (ex : *aatggtgtGgtccctcgtaC*) et *AC* (ex : *aatggtgtAgtccctcgtaC*), et le second parent l'haplotype *GA* (ex : *aatggtgtGgtccctcgtaC*), Stacks déclare un marqueur *ab/aa* pour ce locus. Le programme **genotypes** associe alors *GA* à *a*, et *AC* à *b* chez les parents puis scanne les descendants afin d'identifier quels haplotypes sont présents pour chacun d'entre eux et enregistrer les génotypes associés (soit *ab* ou *aa* dans le cas présent).

```

/home/genouest/admin/galaxy/dependencies/stacks/1.40/iuc/package_stacks_1_40/5eac5e2d91e2/bin/genotypes -b 1 -P stacks_outputs -r 1 -c -s -t CP 2>&1
Found 95 input file(s).
  Parsing stacks_outputs/batch_1.catalog.tags.tsv
  Parsing stacks_outputs/batch_1.catalog.snps.tsv
  Parsing stacks_outputs/batch_1.catalog.alleles.tsv
  Parsing stacks_outputs/female.matches.tsv
  Parsing stacks_outputs/male.matches.tsv
  Parsing stacks_outputs/progeny_1.matches.tsv
  Parsing stacks_outputs/progeny_10.matches.tsv

```

```

    Parsing stacks_outputs/progeny_94.matches.tsv
Identified parent IDs: 1 2
Populating observed haplotypes for 95 samples, 459 loci.
Performing automated corrections...
    Parsing stacks_outputs/progeny_1.tags.tsv
    Parsing stacks_outputs/progeny_1.snps.tsv
    Parsing stacks_outputs/progeny_1.alleles.tsv
    Parsing stacks_outputs/progeny_10.tags.tsv
    Parsing stacks_outputs/progeny_10.snps.tsv
    Parsing stacks_outputs/progeny_10.alleles.tsv

Parsing stacks_outputs/progeny_94.snps.tsv
Parsing stacks_outputs/progeny_94.alleles.tsv
41571 potential genotypes in 447 markers, 38585 populated; 2121 corrected, 2032 converted to heterozygotes, 89 unsupported homozygotes removed.
Testing catalog loci for mapping parents with missing alleles...corrected 343 catalog loci.
Writing 447 loci to JoinMap file, 'stacks_outputs/batch_1.genotypes_1.loc'
Writing SQL markers file to 'stacks_outputs/batch_1.markers.tsv'
Writing SQL genotypes file to 'stacks_outputs/batch_1.genotypes_1.txt'
Writing 459 loci to observed haplotype file, 'stacks_outputs/batch_1.haplotypes_1.tsv'

denovo_map.pl completed at 2016-06-20 14:54:01

```

Au final, 447 loci, marqueurs, sont conservés pour créer le fichier génotype ***batch_1.genotypes_1.tsv***, 459 loci sont répertoriés dans le fichier haplotype observé ***batch_1.haplotypes_1.tsv***, denovo_map.pl s'est exécuté en moins d'une minute. Les 41571 génotypes potentiels sont enregistrés dans le fichier ***batch_1.genotypes_1.txt***. 447 marqueurs sont enregistrés dans le fichier ***batch_1markers.tsv***...

2. Les fichiers matches

Nous avons déjà vu dans la section précédente les fichiers tags, snps, alleles et models. Nous voyons cette fois ci apparaître la catégorie de fichier matches.

Column	Name	Description
1	Sql ID	This field will always be "0", however the MySQL database will assign an ID when it is loaded.
2	Batch ID	
3	Catalog ID	
4	Sample ID	
5	Stack ID	
6	Haplotype	
7	Stack Depth	

Sample1.snps à gauche et *sample2.snps* à droite

#SQL_ID	Batch_ID	Catalog_ID	Sample_ID	Stack_ID	Haplotype	Stack_depth	#SQL_ID	Batch_ID	Catalog_ID	Sample_ID	Stack_ID	Haplotype	Stack_depth
0	1	1	1	1	C	37	0	1	18	2	1	T	95
0	1	1	1	1	T	40	0	1	22	2	2	T	58
0	1	2	1	2	G	47	0	1	315	2	3	T	81
0	1	3	1	3	G	33	0	1	391	2	4	T	75
0	1	4	1	4	A	53	0	1	1	2	5	T	92
0	1	5	1	5	G	21	0	1	227	2	6	AT	11
0	1	6	1	6	C	73	0	1	227	2	6	GC	15
0	1	7	1	7	C	24	0	1	388	2	7	AC	47
0	1	7	1	7	G	29	0	1	388	2	7	AT	35
0	1	8	1	8	TA	50	0	1	259	2	8	T	48
0	1	9	1	9	A	22	0	1	274	2	9	G	42
0	1	9	1	9	G	32	0	1	7	2	10	G	63
0	1	10	1	10	C	67	0	1	139	2	11	C	48
0	1	11	1	11	C	39	0	1	139	2	11	T	35
0	1	12	1	12	G	49	0	1	44	2	12	AA	32
0	1	13	1	13	C	34	0	1	14	2	13	G	70
0	1	13	1	13	T	43	0	1	10	2	14	C	29
0	1	14	1	14	G	36	0	1	10	2	14	G	44
0	1	14	1	14	T	28	0	1	13	2	15	T	45
0	1	15	1	15	C	75	0	1	3	2	16	A	51
0	1	15	1	15	T	80	0	1	3	2	16	G	19

Le Catalog_ID (= Stack_ID du catalog), reprend le Stack_ID de l'individu de "référence", ici sample 1, mais la numérotation est bien différente de celle du Stack_ID du sample 2.... Ainsi, dans le fichier "*catalog.alleles.tsv*", le Stack_ID 3 correspond au Stack_ID 16 du sample 2!

<-----A modifier -----

Dans le cas présent, observons les fichiers matches (il faudra sans doute modifier le datatype "tsv" en "tabular") :

The screenshot shows the Galaxy interface with two data tables side-by-side. Both tables are titled "additional file with STACKS : De novo map". The left table is labeled "(male.matches)" and the right table is labeled "(female.matches)". Both tables have columns for Stack_ID (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), Position (e.g., 291, 61), and Alleles (e.g., CC, TG). The data shows some overlaps between the male and female datasets.

Stack_ID	Position	Alleles	Stack_ID	Position	Alleles
0	291	CC	46		
0	61	TG	43		
0	292	C	32		
0	292	T	59		

Stack_ID	Position	Alleles	Stack_ID	Position	Alleles
0	291	CG	57		
0	61	TG	41		
0	292	A	70		
0	292	T	60		

Nous pouvons alors identifier la correspondance entre le Stack_ID 2 de l'individu "female" (jeu de données de gauche), portant le SNP 61, le Stack_ID 291 de l'individu "male" (jeu de données du milieu) portant les SNPs 61 et 62 et le Stack_ID 2 du catalogue (jeu de données de droite) présentant les deux SNPs (61 et 62) :

The screenshot shows the Galaxy interface with three data tables. The first table is titled "additional file with STACKS : De novo map" and contains raw genotype data (0, 1, 2) and allele frequency data (e.g., -68.2253, -87.4622). The second table is titled "additional file with STACKS : De novo map (male.snps)" and contains SNP coordinates (e.g., 291, 61) and alleles (e.g., C, T). The third table is titled "catalog.snps with STACKS : De novo map on data 2" and contains SNP coordinates (e.g., 61, 62) and alleles (e.g., C, T). The data shows how SNPs from the male dataset align with the catalog.

Stack_ID	Position	Alleles	Stack_ID	Position	Alleles	Stack_ID	Position	Alleles
0	45	C T	0	291	61	0	61	C T
0	61	C T	0	291	62	0	62	C G
0	292	C T	0	292	29	0	30	A C
0	293	C G	0	293	54	0	4	C T
0	294	T C	0	294	24	0	5	A G
0	295	G A	0	295	50	0	6	T G
0	297	T A	0	297	42	0	7	T C
0	298	G A	0	298	30	0	8	C T
0	302	T A	0	302	65	0	9	G T
0	303	T G	0	303	30	0	10	G A
0	304	C T	0	304	26	0	11	C T
0	305	T A	0	305	31	0	12	A G
0	306	C A	0	306	54	0	13	T G
0	307	G C	0	307	19	0	14	C T
0	308	A G	0	308	9	0	15	A G

Nous avons détecter des SNPs chez nos 2 individus et nous pouvons déterminer lesquels sont situés sur les même loci.

-----A modifier -----



3. Exercices pour comprendre les différents types de fichiers

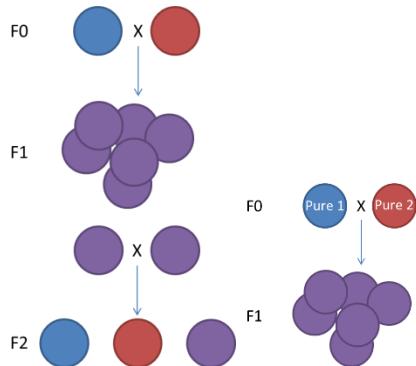
-Pouvez-vous identifier à quoi correspond le nombre de loci de 425 ? En parcourant le fichier catalog.tags, identifiez les loci spécifiques à chacun des individus et ceux partagés. A partir du fichier, compter le nombre de loci du catalogue provenant de l'individu 1, de l'individu 2, ceux retrouvés chez les 2 parents.

-Utiliser les fichiers individu.tags.tsv pour retrouver le nombre de tags par individu. Vous serez probablement amenés à utiliser l'outil "count"

4. Suite de l'analyse

Exécuter l'outil "[**STACKS : genotypes**](#)" sur la data collection « Full output from denovo_map... » sans modifier les paramètres par défaut. **ATTENTION, il faut attendre que la data collection soit bien générée (plus en jaune...).** Cette nouvelle étape permet :

1/de spécifier un type particuliers de carte (ex: F1, F2 (F1xF1), Doubles Haploid, Back Cross (F1xF0), Cross Pollination(=F1 ou F2 mais chaque parent F0 supposé pure et homozygotes))



2/d'exporter les informations génotypiques dans différents formats spécifiques de programmes tiers de cartographie (ex: JoinMap, R/qtl). Il faut savoir que le format R/QTL pour un protocole F2 peut servir en entrée d'outils comme MapMaker ou Carthagène

3/de spécifier un nombre minimum de descendants et/ou une couverture minimum dans la pile pour considérer un locus

4/d'effectuer des corrections automatiques. Concernant ce dernier point, il est effectivement possible de demander au programme **genotypes** d'effectuer des corrections automatiques pour certaines erreurs comme la vérification des tags homozygotes dans la descendance pour assurer qu'un SNP n'est pas présent. En effet, si le modèle de détection de SNP ne peut pas identifier un site comme hétérozygote ou homozygote, le site est provisoirement marqué comme homozygote pour faciliter la recherche, par **sstacks**, de concordances avec le catalogue de loci. Si un second allèle identifié dans le catalogue (i.e., chez les parents) est présent chez un des descendants à une faible fréquence (<10% des lectures de la pile considérée), le programme **genotypes** corrige le génotype. De même, il supprimera un génotype identifié comme homozygote chez un individu particulier si le génotype au locus est supporté par moins de 5 lectures. Les génotypes corrigés sont alors notés en **majuscule**. Il faut savoir que l'utilisation de l'interface web de Stacks permet de modifier manuellement les génotypes. Ceci est intéressant notamment lorsqu'un allèle est sous séquencé chez un individu. Il est alors impossible pour Stacks de le dissocier d'une erreur de séquençage, et ce génotype sera marqué comme homozygote. Si l'utilisateur sait que l'allèle alternatif existe (en observant les génotypes d'autres individus par exemple), il peut alors le corriger manuellement avant de réexécuter **genotypes**.

Reexécuter l'outil "[STACKS : genotypes Run the STACKS genotypes program](#)" sur la « data collection » « Full output from denovo_map ...» en modifiant le nombre minimum de descendants devant être génotypés pour reporter le marqueur à 93 (très drastique, tous les descendants ;)).

The screenshot shows the Galaxy web interface with the 'Stacks: genotypes' tool selected. In the 'Genotyping options' section, the 'Minimum number of reads required at a stack to call a homozygous genotype' is set to 5. The 'Heterozygote minor allele minimum frequency' is set to 0.05. The 'Marker density' is set to 93. The 'History' panel on the right lists several completed runs, including '902: Haplotypes' and '901: Markers'.

Comparer les résultats....

<-----A modifier ----->

Comparer automatic correction et non correction... :

Archive name:

- 110: total_output.zip with STACKS : De novo map on data 1 and data 2
- 27: matches_output.zip with STACKS : De novo map on data 1 and data 2
- 26: alleles_output.zip with STACKS : De novo map on data 1 and data 2
- 25: snps_output.zip with STACKS : De novo map on data 1 and data 2
- 24: rands_output.zip with STACKS : De novo map on data 1 and data 2

This is a batch mode input field. A separate job will be triggered for each dataset.

Merges all files into one: No

Tool documentation:

This tool simply decompresses an archive file (zip, gz, tar.gz, fastq.gz, fastq.bz2 or tar.bz2) and merges all files into only one. If the merge option is enabled, you can delete as many header lines as you need.

Created and integrated by:

Cyril Monjeaud
GenOuest Bio-informatics Core Facility
UMR 6074 IRISA INRIA-CNRS-UR1 Rennes (France)
support@genouest.org

If you use this tool in Galaxy, please cite :

[Y. Le Bras, A. Rout, C. Monjeaud, M. Bahin, O. Quenez, C. Heineau, A. Bretauveau, O. Sallou, O. Collin, Towards a Life Sciences Virtual Research Environment : an e-Science initiative in Western France. JOBIM 2013.](#)

On découvre alors que l'un des allèles est supporté par 6 lectures, l'autre 4. Il n'y avait donc pas une couverture suffisante pour indiquer ce SNP. La correction automatique va ici permettre de prendre en compte le deuxième allèle qui n'avait pas été considéré lors de la détection de SNP et marquer le génotype comme hétérozygote.

Sort on data 300					
0	1	41	3	147	A
0	1	41	3	147	T
0	1	42	3	340	T
0	1	43	3	145	A
0	1	43	3	145	T
0	1	44	3	181	A
0	1	45	3	366	G
0	1	47	3	43	A
0	1	47	3	43	T
0	1	49	3	364	G
0	1	49	3	364	T
0	1	50	3	140	A
0	1	50	3	140	C
0	1	51	3	240	C
0	1	51	3	240	T
0	1	52	3	270	G
0	1	52	3	270	T

decompress_an_archive.log (progeny_1.alleles.tsv)					
0	3	225	C	33.3333	3
0	3	225	T	66.6667	6
0	3	226	C	45	9
0	3	226	T	55	11
0	3	228	A	53.8462	7
0	3	228	C	46.1538	6
0	3	232	A	40	2
0	3	232	G	60	3
0	3	236	A	52.381	11
0	3	236	T	47.619	10
0	3	239	CG	66.6667	6
0	3	239	TA	33.3333	3
0	3	240	C	40	4
0	3	240	T	60	6
0	3	241	C	60	6
0	3	241	G	40	4

Un autre type de correction aurait pu être apporter. Par exemple, si un génotype avec une couverture trop faible avait été indiqué homozygote, il aurait été supprimé du jeu de données car on ne peut pas être sûr qu'un autre allèle n'existe pas, et qu'il n'a juste pas été séquencé. De même, si les couvertures sont trop faible pour chacun des deux allèles, genotypes peut supprimer ce génotype. C'est ici le cas pour le locus 458 du catalogue, correspondant au locus 64 de l'individu progeny_1. Les allèles ont une couverture de 3 et 2. Le seuil par défaut étant 5, les deux allèles sont possiblement douteux.

The screenshot shows the Galaxy / GenOuest interface with two data tables. The left table is titled "additional file with STACKS : genotypes" and contains the following data:

450	--/ab	33	1	--	
451	--/ab	33	1	-	
452	--/ab	44	1	--	
453	--/ab	90	1	aa	
454	--/ab	93	1	bb	
455	--/ab	36	1	--	
456	--/ab	90	1	-	
457	--/ab	32	1	--	
458	--/ab	39	1	aa	
459	--/ab	80	1	aa	

The right table is titled "Sort on data 306" and contains the following data:

0	1	439	3	331	GT
0	1	441	3	130	A
0	1	441	3	130	G
0	1	442	3	228	AG
0	1	443	3	25	TA
0	1	444	3	39	A
0	1	444	3	39	T
0	1	446	3	381	C
0	1	446	3	381	T
0	1	447	3	10	A
0	1	448	3	121	C
0	1	449	3	27	A
0	1	449	3	27	C
0	1	451	3	189	C
0	1	453	3	355	A
0	1	453	3	355	G
0	1	454	3	52	C
0	1	455	3	368	G
0	1	456	3	363	GT
0	1	458	3	64	AT
0	1	458	3	64	TC
0	1	459	3	377	A

A modifier



a) Les fichiers genotypes.tsv

Une ligne par locus, une colonne par individu (aa, ab, AB si correction automatique, bb, bc, ...) avec le génotype observé à chacun des loci.

# Catalog ID	Marker	Cnt	Seg Dist	progeny_1	progeny_10	progeny_11	progeny_12	progeny_13	progeny_14
1	ab/aa	95		1 ab	ab	aa	ab	ab	ab
2	ab/aa	95		1 ab	ab	ab	ab	aa	aa
3	aa/ab	94		1 -	aa	ab	aa	ab	aa
4	ab/ac	94		1 aa	ac	-	ac	ab	bc
5	aa/ab	94		1 ab	aa	ab	aa	ab	aa
6	aa/ab	93		1 -	aa	ab	aa	AB	aa
7	ab/aa	92		1 -	aa	ab	aa	aa	ab
8	ab/aa	95		1 aa	ab	aa	aa	aa	aa
9	aa/ab	90		1 -	aa	AB	aa	-	aa
10	aa/ab	95		1 AB	aa	ab	aa	aa	AB
11	ab/ac	95		1 bc	ab	ac	ab	ac	aa
12	ab/ac	94		1 ac	aa	ac	aa	ac	ab
13	ab/aa	95		1 aa	aa	aa	aa	aa	ab
14	aa/ab	94		1 -	aa	ab	aa	ab	aa
15	aa/ab	95		1 ab	aa	ab	aa	ab	aa
16	ab/aa	84		1 -	ab	ab	ab	ab	aa
17	aa/ab	95		1 ab	ab	aa	ab	aa	aa
18	ab/aa	95		1 ab	ab	ab	ab	ab	aa
19	aa/ab	95		1 ab	aa	ab	aa	ab	aa
20	ab/aa	93		1 AB	aa	ab	ab	aa	aa
21	ab/ac	94		1 ab	bc	aa	bc	bc	ab
22	aa/ab	95		1 aa	ab	aa	ab	aa	AB
23	ab/aa	95		1 aa	aa	ab	aa	ab	ab

b) Les fichiers genotypes.txt

Une ligne par individu et pour chaque individu, à chaque locus du catalogue, le génotype.

# SQL ID	Batch ID	Catalog Locus ID	Sample ID	Genotype
0	1	1	3	ab
0	1	1	4	ab
0	1	1	5	aa
0	1	1	6	ab
0	1	1	7	ab
0	1	1	8	ab
0	1	1	9	ab
0	1	1	10	aa
0	1	1	11	aa
0	1	1	12	aa
0	1	1	13	ab
0	1	1	14	ab
0	1	1	15	ab
0	1	1	16	aa
0	1	1	17	AB
0	1	1	18	aa
0	1	1	19	aa
0	1	1	20	ab
0	1	1	21	ab
0	1	1	22	ab
0	1	1	23	ab
0	1	1	24	ab
0	1	1	25	aa

c) Les fichiers haplotypes.tsv

Cnt	Seg Dist	female	male	progeny_1	progeny_10	progeny_11	progeny_12	progeny_13	progeny_14
95	0.5	A/G	A	A/G	A/G	A	A/G	A/G	A/G
95	0.5	A/T	T	A/T	A/T	A/T	A/T	T	T
94	0.5	T	C/T	-	T	C/T	T	C/T	T
95	0.1	AA/GA	GA/GG	GA	GA/GG	GA	GA/GG	AA/GA	AA/GG
94	0.5	C	A/C	A/C	C	A/C	C	A/C	C
93	0.5	A	A/T	-	A	A/T	A	T	A
92	0.05	A/G	A	-	A	A/G	A	A	A/G
95	0.5	A/T	T	T	A/T	T	T	T	T
90	0.0005	A	A/G	-	A	A	A	G	A
95	0.5	G	C/G	C	G	C/G	G	G	C
95	0.5	AC/GC	AC/GG	GC/GG	AC/GC	AC/GG	AC/GC	AC/GG	AC
94	0.5	AT/CA	AA/CA	AA/CA	CA	AA/CA	CA	AA/CA	AT/CA
95	0.5	A/G	A	A	A	A	A	A	A/G

A la fin de cette partie, nous obtenons des génotypages d'individus pouvant être utilisés dans certains logiciels permettant de créer des cartes génétiques comme Carthagène ou Mapmaker

5. Exercices

L'application du deleverage algorithm permet de ne pas considérer les loci obtenus à partir de la fusion d'un nombre de stacks > 3. Pourquoi 3 alors que biologiquement on s'attend à 2 pour individu à génome diploide?

Réponse : Pour éviter de virer des loci pour lesquels on a 99,9% des individus qui ont 2 allèles et 0,01 % qui ont 3 allèles à cause d'erreurs de séquençage. Il est donc important de vérifier cette proportion de loci à 3 allèles.

Dans le catalogue de loci, combien sont spécifiques à l'individu « female », puis « male » et combien sont commun ?

Spé female 35/Spé male 34/Commun 390 soit 459 en tout.

Relancer denovo_map avec une valeur de modèle de SNP plus stringente (ie 0,001 vs 0,05 par défaut)

Oui, voir par exemple l'individu progeny_93 où on passe de 2 à 3 haplotypes non vérifiés....

III. Analyse RAD-seq sous Galaxy : Construction de mini contigs à partir de séquences pairées

Il est possible d'assembler des mini-contigs à partir de lectures pairées issues de RAD-seq. Disposant ainsi de plusieurs centaines de nucléotides génomiques situés à proximité de chaque marqueur, l'ancrage de ces marqueurs à des librairies d'EST, et donc leur connexion à des gènes codant pour des protéines chez d'autres organismes, est facilité.

L'outil "[**STACKS : assemble read pairs by locus Run the STACKS sort_read_pairs.pl and exec_velvet.pl wrappers**](#)" permet de rassembler les lectures pairées associées à chaque stack au sein d'un fichier fasta. Une fois un fichier fasta généré par locus du catalogue, Velvet est utilisé pour assembler les lectures de chaque fichier.

Comme le fichier fasta doit porter dans l'identifiant de la séquence l'identifiant du locus du catalogue, il est possible d'ajouter manuellement des séquences à un locus. Ainsi, en plus des mini-contigs associés automatiquement aux loci par l'outil "[**STACKS : assemble read pairs by locus Run the STACKS sort_read_pairs.pl and exec_velvet.pl wrappers**](#)", nous pouvons ajouter manuellement des séquences d'EST disponibles, ou construites de novo en utilisant des données de RNA-seq. Cette étape peut être réalisée après avoir associé des séquences au catalogue de loci en utilisant Blast ou Bowtie.

Créer un nouvel historique et renommer le (ex : STACKS 1.40 GCC Training RAD 3: Create mini contig from PE sequences).

Nous travaillerons sur les données issues de l'archive pe_sample.tar.gz disponible sur le site de Stacks (http://catchenlab.life.illinois.edu/stacks/pe_tutorial/pe_samples.tar.gz). Récupération des données brutes dans Shared data/data libraries/1 Galaxy teaching folder/EnginesOn/mini-contig (URL d'origine : http://catchenlab.life.illinois.edu/stacks/pe_tut.php) . Vous y trouverez

- 2 jeux de données de lectures pairées pour un individu nommée « female » : **f0_female.1.fq**, **f0_female.2.fq**.

Sélectionner les deux jeux de données, et cliquer sur le bouton "to History".

name	description	data type	size	time updated (UTC)
f0_female.fq_1		fastq	256.7 MB	2016-06-20 02:53 PM
f0_female.fq_2		fastq	246.7 MB	2016-06-20 02:56 PM

Vous débutez donc avec les deux jeux de données dans votre historique comme suit

GenOuest Bioinformatics platform

Development, expertise and resources for bioinformatics

Afin d'étudier la qualité et le format fastQ des, nous pouvons exécuter **FastQC**, révélant un format d'encodage de qualité Illumina 1.5, puis **FastQ groomer** pour formater les données au format *fastqsanger*.

ATTENTION !!!!!

- Pour utiliser denovo_map, il ne faut pas des noms de dataset d'entrée contenant des espaces !!!! Il faut donc renommer (en utilisant le petit crayon d'édition des attributs) les datasets de fastQGroomer en remplaçant les espaces par des underscores par exemple.

Sélectionner l'outil "[**STACKS : De novo map Run the STACKS denovo_map.pl wrapper**](#)".

Indiquez que vous souhaitez utiliser la fonction "Genetic map".

STACKS ne gère pas les données pairées. En effet, si d'un côté du fragment le séquençage se fera toujours au niveau du même site (correspondant au site de coupure de l'enzyme), de l'autre, ce n'est pas le cas. Du coup, une belle pile présentant une forte couverture sera déterminée dans le premier sens de séquençage et pas dans l'autre. L'utilisation des lectures pairées entraînant de nombreux biais, en tout cas, présentant un intérêt très limité lors de l'exécution de "[**STACKS : De novo map Run the STACKS denovo_map.pl wrapper**](#)", il est conseillé de n'utiliser que les lectures provenant d'un seul sens de séquençage. Toutefois, si vous disposez, comme c'est le cas ici, de lectures pairées (avec toute les paires au complet), vous pouvez utiliser l'option "*Paired-end fastq files*". En cochant cette option lors de l'exécution de "[**STACKS : De novo map Run the STACKS denovo_map.pl wrapper**](#)", vous indiquez à l'outil de trier les fichiers fastq présents dans l'archive, et de ne s'exécuter que sur les premiers jeux de données pairées, donc les données séquencées en forward. Ainsi, la même archive pourra être utiliser en entrée de l'outil "[**STACKS : assemble read pairs by locus Run the STACKS sort_read_pairs.pl and exec_velvet.pl wrappers**](#)".

Vous pouvez préciser certains paramètres. Ainsi, en sélectionnant le mode Advanced pour le paramètre **Stack assembly options**, vous pouvez préciser:

-la profondeur minimum de la pile (-m), ici "Minimum number of identical". Ce paramètre, passé à ustacks, contrôle le nombre de lectures correspondant exactement devant être trouvé pour créer une pile chez un individu. Nous pouvons sélectionner 3, une pile ne sera générée chez un individu que si au moins 3 lectures correspondent exactement.

-la distance maximum permise entre des piles pour qu'elles soient fusionnées en un locus potentiel chez un individu. Il s'agit du paramètre -M, ici "Number of mismatches allowed between loci when processing a single individual". Nous préciserons ici une distance maximale de 3 nucléotides.

-supprimer ou casser les RAD-tags très répétitifs. Il s'agit de supprimer les piles "bûcheronnes" de l'analyse et briser les piles modérément surdimensionnées. Nous cocherons ici cette option, correspondant au paramètre -t.

Temps d'exécution < 10min.

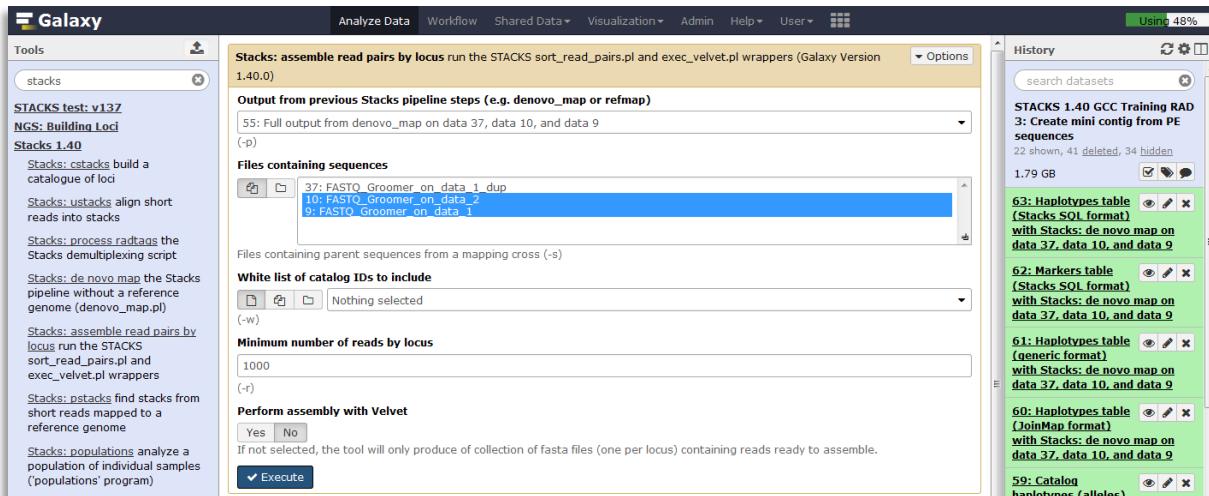
Dans la « data collection » *Full output...* générée, nous trouvons plusieurs jeux de données:

- f0_male.2.alleles.tsv : répertorie les haplotypes identifiés à chaque locus.
- f0_male.2.matches.tsv : répertorie les haplotypes vérifiés et se référant aux loci du catalogue. Ici 292 499 haplotypes sont répertoriés. Il s'agit du dernier fichier généré.

Trois fichiers répertorient le même type d'information (loci, SNPs et allèles) mais à l'échelle de tous les échantillons.

- batch_1.catalog.tags.tsv : qui présente les 382 005 loci enregistrés dans le catalogue.
- batch_1.catalog.snps.tsv : qui présente les 72 466 SNPs enregistrés dans le catalogue.
- batch_1.catalog.alleles.tsv : qui présente les 146 753 allèles enregistrés dans le catalogue.
- batch_1.haplotypes_1.tsv : qui présente les 381 735 haplotypes enregistrés dans le catalogue
- batch_1.genotypes_1.tsv : vide car pas de descendants indiqués.
- batch_1.genotypes_1.txt : vide car pas de descendants indiqués.
- batch_1.markers.tsv : vide car pas de descendants indiqués.

Sélectionner ensuite l'outil "[STACKS : assemble read pairs by locus](#) Run the STACKS sort_read_pairs.pl and exec_velvet.pl wrappers". Temps d'exécution ~ 1 heures 30



Indiquez :

- la data collection « Full output... » générée par l'outil "[STACKS : De novo map](#) Run the STACKS denovo_map.pl wrapper".

- les fichiers de lectures.

- Si besoin, une liste des loci à assembler. Cette liste peut correspondre par exemple à l'ensemble des loci du catalogue de loci présentant des SNPs....

Nous spécifierons un nombre minimum de lectures pour assembler un locus de 200.

Nous activerons enfin l'option « Perform assembly with velvet »

9543 loci ont été assemblé sur les 65877 créé précédemment pendant l'exécution de denovo_map.

Observez bien le fichier fasta généré. Pouvez-vous me dire combien de contig ont une taille supérieure à 300 pb puis 400 pb ? Les outils FASTA-TO-TABULAR, Convert delimiters to tab et Filter pourront vous être utiles.

IV. Analyse RAD-seq sous Galaxy : La génomique des populations

Nous allons travailler à partir des données de la publication d'*Hohenlohe et al. 2010*.

OPEN  ACCESS Freely available online

PLOS GENETICS

Population Genomics of Parallel Adaptation in Threespine Stickleback using Sequenced RAD Tags

Paul A. Hohenlohe^{1*}, Susan Bassham^{1*}, Paul D. Etter², Nicholas Stiffler³, Eric A. Johnson², William A. Cresko^{1*}

¹ Center for Ecology and Evolutionary Biology, University of Oregon, Eugene, Oregon, United States of America, ² Institute of Molecular Biology, University of Oregon, Eugene, Oregon, United States of America, ³ Genomics Core Facility, University of Oregon, Eugene, Oregon, United States of America

Abstract

Next-generation sequencing technology provides novel opportunities for gathering genome-scale sequence data in natural populations, laying the empirical foundation for the evolving field of population genomics. Here we conducted a genome scan of nucleotide diversity and differentiation in natural populations of threespine stickleback (*Gasterosteus aculeatus*). We used Illumina-sequenced RAD tags to identify and type over 45,000 single nucleotide polymorphisms (SNPs) in each of 100 individuals from two oceanic and three freshwater populations. Overall estimates of genetic diversity and differentiation among populations confirm the biogeographic hypothesis that large panmictic oceanic populations have repeatedly given rise to phenotypically divergent freshwater populations. Genomic regions exhibiting signatures of both balancing and divergent selection were remarkably consistent across multiple independently derived populations, indicating that replicate parallel phenotypic evolution in stickleback may be occurring through extensive parallel genetic evolution at a genome-wide scale. Some of these genomic regions co-localize with previously identified QTL for stickleback phenotypic variation identified using laboratory mapping crosses. In addition, we have identified several novel regions showing parallel differentiation across independent populations. Annotation of these regions revealed numerous genes that are candidates for stickleback phenotypic evolution and will form the basis of future genetic analyses in this and other organisms. This study represents the first high-density SNP-based genome scan of genetic diversity and differentiation for populations of threespine stickleback in the wild. These data illustrate the complementary nature of laboratory crosses and population genomic scans by confirming the adaptive significance of previously identified genomic regions, elucidating the particular evolutionary and demographic history of such regions in natural populations, and identifying new genomic regions and candidate genes of evolutionary significance.

Citation: Hohenlohe PA, Bassham S, Etter PD, Stiffler N, Johnson EA, et al. (2010) Population Genomics of Parallel Adaptation in Threespine Stickleback using Sequenced RAD Tags. PLoS Genet 6(2): e1000862. doi:10.1371/journal.pgen.1000862

Editor: David J. Begun, University of California Davis, United States of America

Received October 20, 2009; Accepted January 28, 2010; Published February 26, 2010

Copyright: © 2010 Hohenlohe et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: This work was funded by grants from the National Science Foundation (IOS-0642264) and from the National Institutes of Health (R24GM079486-01A1 and Ruth L. Kirschstein National Research Service Award F32 GM079494). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: wcresko@uoregon.edu

© These authors contributed equally to this work.

Introduction

Population genetics provides a rich and mathematically rigorous framework for understanding evolutionary processes in natural populations. This theory was built over the last hundred years by modeling the processes of selection, genetic drift, mutation and migration in spatially distributed populations [1–6]. The field has concentrated primarily on the dynamics of one or a small number of genetic loci, largely because of methodological limitations. However, genes are not islands, but rather form part of a genomic community, integrated both by physical proximity on chromosomes and by various evolutionary processes [7–10]. With technological advances, such as Next Generation Sequencing (NGS) [11–13], the emerging field of population genomics now allows us to address evolutionary processes at a genomic scale in natural populations [14–20]. Population genetic measures like Wright's F statistics [2,21,22], traditionally viewed as point estimates, can now be examined as continuous distributions across a genome [23–29]. As a result, in addition to estimating genome-wide averages for such statistics, we can identify specific genomic regions that exhibit significantly increased or decreased differentiation among populations, indicating regions that have likely been under strong diversifying or stabilizing natural selection [9,30–41]. These signatures of selection can then be used to identify candidate pathways, genes and alleles for targeted functional analyses [42–47].

An excellent opportunity for this type of population genomics approach exists in the threespine stickleback, *Gasterosteus aculeatus* [48–50]. This small fish is distributed holocentrally and inhabits a large number of marine, estuarine and freshwater habitats in Asia, Europe and North America. In many regions replicate extant freshwater stickleback populations have been independently derived from oceanic ancestors when stickleback became isolated postglacially in newly created freshwater habitats [49,51]. Population genetic data support this inference, and also indicate that present day oceanic populations can be used as surrogates for stock that gave rise to nearby derived freshwater populations [52–64]. Because of the varied selection regimes in novel habitats,

La génétique des populations est une très vieille discipline riche en théories mathématiques, et utilisant différentes approches statistiques permettant l'inférence de paramètres à partir de données génétiques. Ces statistiques se retrouvent à travers la détermination de la diversité nucléotidique (π), ou de coefficients de différenciation (i.e. F_{ST}), ainsi que les mesures de covariances génétique comme le déséquilibre de liaison (D and D'). Cependant, à cause de limitations méthodologiques notamment, la majorité des travaux théoriques, statistiques et empiriques en génétique des populations s'est concentré sur un nombre restreint de loci. Avec l'avènement des NGS, des dizaines ou centaines de milliers de marqueurs génétiques peuvent désormais être examinés sur de nombreux individus, permettant à la discipline nommée génomique des populations de devenir une réalité. Une nouvelle activité très excitante en génomique des populations réside dans le fait d'identifier des signatures de sélection dans les populations sauvages. Aujourd'hui, on travaillera sur des données de RAD à partir d'échantillonnage océanique et d'eau douce de populations d'épinoches.

1. Récupération des données

Sous Galaxy, commencez par créer un nouvel historique (ex: "STACKS 1.40 GCC Training RAD 4: population genomics").

Les données peuvent être récupérées selon 2 manières:

-via l'outil **EBI SRA ENA SRA** de la section **Get Data**. Les numéros d'accésion des 6 jeux de données vont de SRR034310 à SRR034316 (attention, le moteur de recherche est sensible à la casse ;(). Est représenté ici la récupération du jeu de données **SRR034310** en cliquant sur (**Fasta files (galaxy)**)

The screenshot shows the Galaxy web interface with the EBI SRA ENA SRA tool selected. The main content area displays the ENA (European Nucleotide Archive) search results for run SRR034310. The results table includes columns for Study accession, Secondary study accession, Sample accession, Secondary sample accession, Experiment accession, Run accession, Scientific name, Instrument model, Library layout, Fastq files (ftp), Fastq files (galaxy), Submitted files (ftp), Submitted files (galaxy), CoL tax ID, Col. scientific name, and Reference alignment. The data for SRR034310 is as follows:

Study accession	Secondary study accession	Sample accession	Secondary sample accession	Experiment accession	Run accession	Scientific name	Instrument model	Library layout	Fastq files (ftp)	Fastq files (galaxy)	Submitted files (ftp)	Submitted files (galaxy)	CoL tax ID	Col. scientific name	Reference alignment
SRP001747	SRP001747	SAMN00010788	SRS009839	SRX015871	SRR034310	Gasterosteus aculeatus	Illumina Genome Analyzer	SINGLE	File 1	File 1			13700198	Gasterosteus aculeatus	N

-via la section Shared data/data libraries/RADseq/Stickleback population Genomics. Vous y trouverez les données brutes Illumina ainsi que les informations concernant les barcodes et les populations liées aux différents numéros d'accèsion. Ne sélectionner que les fichiers correspondants au run de séquençage SRR034310 et le fichier « population_map ».

Detailed description of the screenshot: This is a screenshot of the Galaxy web interface. At the top, there's a navigation bar with links for Analyze Data, Workflow, Shared Data, Visualization, Admin, Help, and User. A progress bar on the right indicates 'Using 48%'. Below the navigation is a search bar with the text 'Barcode_SR034310.txt'. Underneath is a table listing 16 items from the library:

name	description	data type	size	time updated (UTC)
Barcode_SR034310.txt		tabular	320 bytes	2016-05-25 01:53 PM
Barcode_SR034311.txt		tabular	320 bytes	2016-05-25 01:54 PM
Barcode_SR034312.txt		tabular	352 bytes	2016-05-25 01:58 PM
Barcode_SR034313.txt		tabular	352 bytes	2016-05-25 01:58 PM
Barcode_SR034314.txt		tabular	352 bytes	2016-05-25 01:59 PM
Barcode_SR034315.txt		tabular	88 bytes	2016-05-25 01:59 PM
Barcode_SR034316.txt		tabular	352 bytes	2016-05-25 01:59 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034310/SRR034310.fastq.gz		fastq.gz	289.5 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034311/SRR034311.fastq.gz		fastq.gz	479.6 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034312/SRR034312.fastq.gz		fastq.gz	786.8 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034313/SRR034313.fastq.gz		fastq.gz	794.1 MB	2016-05-25 02:50 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034314/SRR034314.fastq.gz		fastq.gz	748.4 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034315/SRR034315.fastq.gz		fastq.gz	664.9 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034316/SRR034316.fastq.gz		fastq.gz	855.2 MB	2016-05-25 02:00 PM
population.map		txt	272 bytes	2016-06-23 02:11 PM
Reference_genome_11_chromosomes.fa		fasta	223.9 MB	2016-05-25 01:52 PM

Sélectionner les données d'entrée, l'archive de reads, le fichier de barcode. Spécifier l'enzyme (ici SbfI) et préciser le format d'output préféré, ici FastQ. (Temps d'exécution < 2 minutes) :

Detailed description of the screenshot: This is a screenshot of the Galaxy interface showing the configuration of the 'Stacks: process_radtags' tool. The tool is set to process single-end reads from a fastq.gz file ('SRR034310.fastq.gz'). It uses 'Barcode_SR034310.txt' as the barcode file. The enzyme specified is 'sbfI'. The output format is set to 'fastq'. The 'Capture discarded reads to a file' option is set to 'No'. The 'Output format' dropdown is set to 'fastq'. The 'Execute' button is visible at the bottom. To the right, the 'History' panel shows the results: 'population.map' (16 lines, format: txt, database: ?) and 'Barcode_SR034310.txt' (16 lines, format: tabular, database: ?). The 'population.map' file contains entries like 'SR034310_CCCC' and 'SR034310_GCGG'.

L'outil **STACKS : Process Radtags** produit une data collection contenant les données démultipliquées et un fichier de log. Examinez le et répondez aux questions suivantes:

Combien de lectures brutes étaient présentes ?

Combien ont été retenues?

Sur celles non retenues, quelles étaient les raisons?

Que vous apprend ce résultat au sujet de l'analyse de données et du design de barcodes en général?

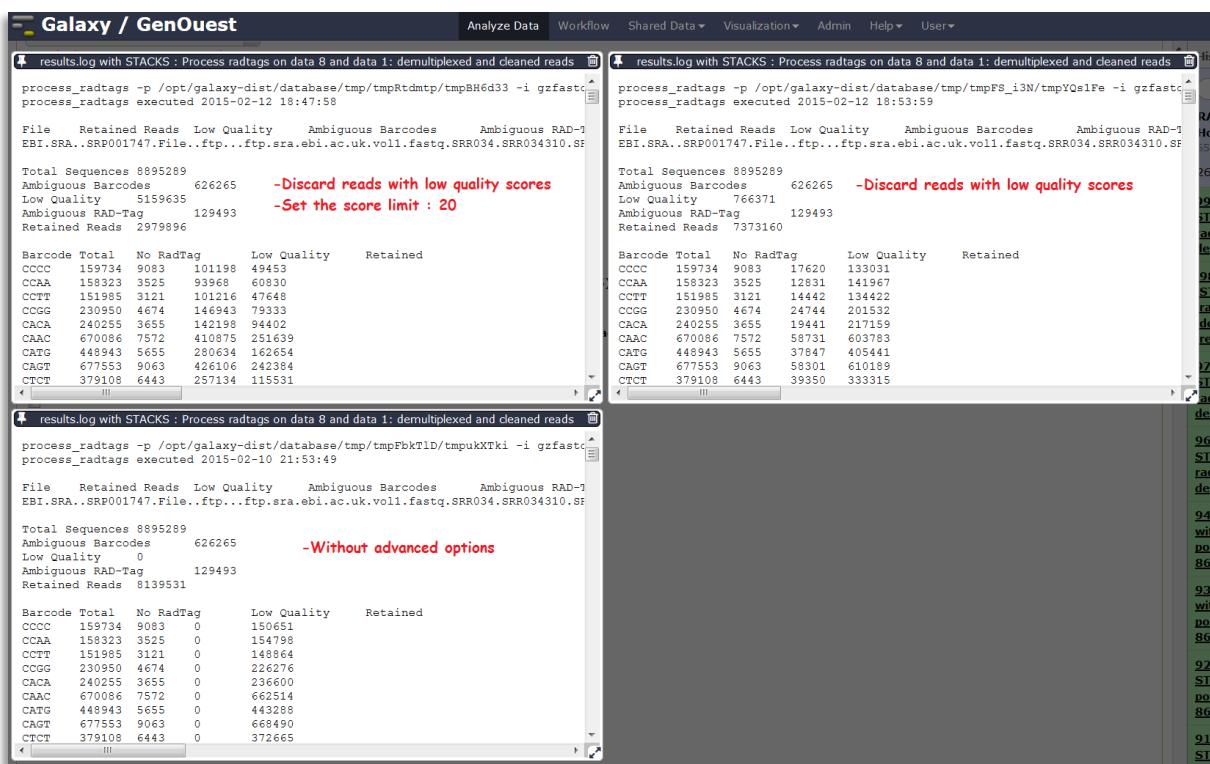
Vous pourrez effectuer plusieurs exécutions de **STACKS : Process Radtags** en jouant sur le score seuil affecté à une fenêtre glissante et en ajoutant les barcodes mentionnés dans le log précédent.

Quel est l'effet de l'augmentation du seuil du score?

Vous pourrez également jouer sur l'ensemble des paramètres de **STACKS : Process Radtags** en spécifiant notamment une mauvaise enzyme de restriction, en faisant varier le score seuil de la fenêtre glissante.

Ci-dessous sont représentés les résultats obtenus en fonction de certaines options choisies :

-Si le filtre sur les faibles qualités n'enlève pas beaucoup de données (ici ~10% de séquences enlevées), la sélection d'un score limite de 20 peut faire fortement chuter le nombre de lectures conservées (ici ~30% de séquences conservées)



-L'utilisation du filtre de type **clean data, remove any read with an uncalled base**, n'a ici que peu d'incidence

Galaxy / GenOuest

Analyze Data Workflow Shared Data Visualization Admin Help User Using 48%

results.log with STACKS : Process radtags on data 1 and data 15: demultiplexed and cleaned reads

```
process_radtags -p /opt/galaxy-dist/database/tmp/tmpGMV9_8/tmpnieQ13 -i gzipfastq -b /opt/galaxy-dist/database/files/049/dataset_49060.dat -o /opt/galaxy-dist/database/tmp/tmpWkg6QS/tmp9Cae6R
process_radtags executed 2015-02-12 19:56:15

File Retained Reads Low Quality Ambiguous Barcodes Ambiguous RAD-Tag Total
EBI.SRA..SRP001747.File..ftp...ftp.sra.ebi.ac.uk.vol1.fastq.SRR034310.SRR034310.fastq.gz 7373160 766371 626265 129493 8895289

Total Sequences 8895289
Ambiguous Barcodes 626265
Low Quality 766371
Ambiguous RAD-Tag 129493
Retained Reads 7373160

Barcode Total No RadTag Low Quality Retained
!!!!
```

results.log with STACKS : Process radtags on data 1 and data 15: demultiplexed and cleaned reads

```
process_radtags -p /opt/galaxy-dist/database/tmp/tmpWkg6QS/tmp9Cae6R -i gzipfastq -b /opt/galaxy-dist/database/files/049/dataset_49060.dat -o /opt/galaxy-dist/database/tmp/tmpWkg6QS/tmp9Cae6R
process_radtags executed 2015-02-13 08:34:22

File Retained Reads Low Quality Ambiguous Barcodes Ambiguous RAD-Tag Total
EBI.SRA..SRP001747.File..ftp...ftp.sra.ebi.ac.uk.vol1.fastq.SRR034310.SRR034310.fastq.gz 7369780 769751 626265 129493 8895289

Total Sequences 8895289
Ambiguous Barcodes 626265
Low Quality 769751
Ambiguous RAD-Tag 129493
Retained Reads 7369780

Barcode Total No RadTag Low Quality Retained
!!!!
```

NGS analysis on non assembled data (default 10): 10

Clean data, remove any read with an uncalled base:

Discard reads with low quality scores:

Rescue barcodes and RAD-Tags:

518 lines
format: txt, database: 2

Using Phred+33 encoding for quality scores.
Found 1 input file(s).
Searching for single-end, inlined barcodes.
Loaded 16 barcodes (4 bp).
Processing file 1 of 1
[EBI.SRA..SRP001747.File..ftp...ftp.sra.ebi.ac.uk.vol1.fastq.SRR034310.SRR034310.fastq.gz]

-Dans le cas présent, nous remarquons plusieurs barcodes associés à un nombre significatif de lectures rangées dans la catégorie **Ambiguous RAD-Tag**. Cela peut s'expliquer par des erreurs de séquençage mais dans le cas présent également par le fait que d'autres échantillons biologiques ont été séquencés sur le même run. Ceci est souvent utilisé, notamment pour éviter les problèmes de clustering intervenant lors de séquençage Illumina de séquences très semblables.

Galaxy / GenOuest

Analyze Data Workflow Shared Data Visualization Admin Help User

process_radtags -p /opt/galaxy-dist/database/tmp/tmpOkEg6y/tmpFcyvy5 -i gzipfastq -b /opt/galaxy-dist/database/files/064/dataset_64711.dat
process_radtags executed 2015-02-12 20:01:38

```
File Retained Reads Low Quality Ambiguous Barcodes Ambiguous RAD-Tag Total
EBI.SRA..SRP001747.File..ftp...ftp.sra.ebi.ac.uk.vol1.fastq.SRR034314.SRR034314.fastq.gz 5468301 636056 8973372 149114 15226843

Total Sequences 15226843
Ambiguous Barcodes 8973372
Low Quality 636056
Ambiguous RAD-Tag 149114
Retained Reads 5468301

Barcode Total No RadTag Low Quality Retained
CAACT 363962 8548 31781 323633
CACTC 456186 8545 40268 407373
CCAC 323673 6239 28670 288764
CCGG 297232 6614 25334 265284
GGAG 334752 11499 39148 284105
GGGG 393285 11866 45914 335505
GTACA 489931 11911 57937 420083
GTGTG 49700 2206 5310 42184
CGATA 667123 14915 64425 587783
CGGG 433323 9082 44966 379375
CTAGG 361429 10343 26467 324619
CTGAA 216192 6074 16073 194045
GAAGC 435685 12719 52882 370084
GAGAT 567900 11400 59633 496867
GCATT 524666 10451 57390 456825
GCGCC 338432 6702 39958 291772

Sequences not recorded
Barcode Total
TAGCA 803854
TGACC 759728
ACACG 644399
TTGGC 598416
AGGAC 541409
AAGGG 505397
ATGCT 486979
ATATC 481049
TCAGG 480914
TGGTT 479964
TAATG 452243
AGAGT 426923
ACGTA 370614
TCGAG 364919
AAAAA 341844
TTAAT 132402
```

2. Sans génome de référence

Nous allons exécuter **STACKS : de novo map** sur les individus étudiés. Il est possible de charger des fichiers FastQ ou directement l'archive "all_files.zip" générée par l'outil **STACKS : Process Radtags** comme c'est le cas ici. Il faudra alors préciser un fichier de type STACKS "*population map*" si l'analyse se fait sur plusieurs populations

Vous pouvez préciser certains paramètres. Ainsi, en sélectionnant le mode Advanced pour le paramètre **Stack assembly options**, vous pouvez préciser:

-la profondeur minimum de la pile (-m), ici "Minimum number of identical raw reads required to create a stack". Ce paramètre, passé à **ustacks**, contrôle le nombre de lectures correspondant exactement devant être trouvé pour créer une pile chez un individu. Nous pouvons sélectionner 3, une pile ne sera générée chez un individu que si au moins 3 lectures correspondent exactement.

-la distance maximum permise entre des piles pour qu'elles soient fusionnées en un locus potentiel chez un individu. Il s'agit du paramètre -M, ici "Number of mismatches allowed between loci when processing a single individual". Nous préciserons ici une distance maximale de 2 nucléotides.

-le nombre maximal de différence entre loci du catalogue. Ce paramètre permet notamment de pouvoir créer un locus de type homozygote dans le catalogue alors qu'il est en réalité hétérozygote. Ceci est pratique pour conserver les loci hétérozygotes entre parents mais homozygote chez chacun d'entre eux. Il s'agit du paramètre -n, "specify the number of mismatches allowed between loci when building the catalog" ici. Nous pouvons fixer ce paramètre à 3.

-supprimer ou casser les RAD-tags très répétitifs. Il s'agit de supprimer les piles "bûcheronnes" de l'analyse et briser les piles modérément surdimensionnées. Nous cocherons ici cette option, correspondant au paramètre -t.

Temps d'exécution ~ 3 heures

Une fois le job terminé, nous pouvons consulter les fichiers "result.log" et "catalog.*" (* couvrant trois types de fichiers déjà mentionnés dans la première partie consacrée à la détection de SNP, à savoir *snps*, *alleles* et *tags*).

```

denovo_map.log with Stacks: de novo map on data 52, data 101, and others
denovo_map.pl version 1.40 started at 2016-06-21 16:58:00
/home/genouest/admin/galaxydependencies/stacks/1.40/luc/package_stacks_1_40/5eac5e2d91e2/bin
Identifying unique stacks: file 1 of 16 [SRRO34310_CAC]
/home/genouest/admin/galaxydependencies/stacks/1.40/luc/package_stacks_1_40/5eac5e2d91e2/bin
stacks_params_set.py: loaded parameters from SRRO34310.CAC
stacks_params_set.py: setting max coverage allowed to create a stack: 3
Max distance allowed between stacks: 2
Max distance allowed to align secondary reads: 4
Max number of stacks allowed per de novo locus: 3
Levenshtein algorithm: enabled
Removal algorithm: enabled
Model type: SNF
Alpha significance level for model: 0.05
Gapfill algorithm: disabled
Parameters SRRO34310.CAC.fq
Loaded 462514 RAD-Tags; inserted 197953 elements into the RAD-Tags hash map.
831 reads contained uncalled nucleotides that were modified.
Initial coverage mean: 12.68027; Std Dev: 54.97661; Max: 6732
Deleveraging: ignoring 100% of tags.
39320 initial stacks were populated; 158633 stacks were set aside as secondary reads.
Calculating distance for removing repetitive stacks.
Distance allowed between stacks: 1; searching with a k-mer length of 15 (18 k-mers per read)
Removing repetitive stacks.
38499 stacks remain for merging.
Post-Reopen Removal, coverage depth Mean: 11.4237; Std Dev: 86.3679; Max: 123
Calculating distance between stacks...
Using Levenshtein algorithm for searching with a k-mer length of 9 (12 k-mers per read)
Merging stacks, maximum allowed distance: 2 (nucleotide(s))
38499 stacks merged into 36321 stacks; leveraged 80 stacks; removed 65 stacks.
After merging, coverage depth Mean: 11.9633; Std Dev: 89.0933; Max: 171
Mean k-mer length: 15.0000
169328 remainder sequences left to merge.
Distance allowed between stacks: 4; searching with a k-mer length of 5 (28 k-mers per read)
Matched 104218 remainder reads; unable to match 59710 remainder reads.
After merging, coverage depth Mean: 14.8423; Std Dev: 89.2292; Max: 203
Calling final consensus sequence, invoking SNP-calling model...
Number of utilized reads: 602094
Writing loci, SNPs, and alleles to 'stacks_outputs/...'
Fetching sequencing IDs from SRRO34310.CAC.fq... read 662514 sequence IDs.
done.
Identifying unique stacks: file 2 of 16 [SRRO34310_CAC]
/home/genouest/admin/galaxydependencies/stacks/1.40/luc/package_stacks_1_40/5eac5e2d91e2/bin
stacks_params_set.py: loaded parameters from SRRO34310.CAC
stacks_params_set.py: setting max coverage allowed to create a stack: 3
Max distance allowed between stacks: 2
Max distance allowed to align secondary reads: 4
Max number of stacks allowed per de novo locus: 3
Levenshtein algorithm: enabled
Removal algorithm: enabled
Model type: SNF
Alpha significance level for model: 0.05
Gapfill algorithm: disabled
Parameters SRRO34310.CAC.fq
Loaded 236400 RAD-Tags; inserted 88362 elements into the RAD-Tags hash map.
300 reads contained uncalled nucleotides that were modified.
Initial coverage mean: 7.15242; Std Dev: 23.9071; Max: 1591
Deleveraging: ignoring 100% of tags.
23028 initial stacks were populated; 65334 stacks were set aside as secondary reads.
Calculating distance for removing repetitive stacks.
Distance allowed between stacks: 1; searching with a k-mer length of 15 (18 k-mers per read)
Removing repetitive stacks.
Removed 501 stacks.

```

Afin de spécifier plus d'options et de pouvoir filtrer les résultats, il est possible de ré exécuter le dernier module de **Stacks : de novo map**, à savoir **populations** sur la data collection "Full output...." générée dans l'étape précédente en spécifiant un fichier de type population map.

The screenshot shows the Galaxy web interface with the 'Stacks: populations' tool selected. The main panel contains several configuration sections:

- Input:** Set to 'Output from previous Stacks pipeline steps (e.g. denovo_map or refmap)' with entry '121: Full output from denovo_map on data 52, data 101, and others'.
- Specify a population map:** Set to '52: population_map'.
- Data filtering options:**
 - Minimum percentage of individuals in a population required to process a locus for that population: 0.5
 - Minimum number of populations a locus must be present in to process a locus: 2
 - Specify a minimum stack depth required for individuals at a locus: 1
 - Specify a minimum minor allele frequency required before calculating Fst at a locus (between 0 and 0.5): 0.25
 - Maximum observed heterozygosity required to process a nucleotide site at a locus: 0.5
- Correction type:** Set to 'No correction'.
- Filter loci with log likelihood values below this threshold:** Set to '(-ln_llim)'.
- Restrict data analysis to only the first SNP per locus:** Set to 'Yes'.
- Restrict data analysis to one random SNP per locus:** Set to 'Yes'.
- Output options:**
 - Enable SNP and haplotype-based F statistics: Yes

The right sidebar displays the history of previous runs, including:

- 131: Stacks SQL format with Stacks: de novo map on data 52, data 101, and others
- 130: Summary statistics for each population with Stacks: de novo map on data 52, data 101, and others
- 129: Summary of summary statistics for each population with Stacks: de novo map on data 52, data 101, and others
- 128: Populations log with Stacks: de novo map on data 52, data 101, and others
- 127: Haplotype-based summary statistics for each locus in each population with Stacks: de novo map on data 52, data 101, and others
- 126: Observed haplotypes with Stacks: de novo map on data 52, data 101, and others
- 125: Catalog haplotypes (alleles) with Stacks: de novo map on data 52, data 101, and others
- 124: Catalog model calls (snps) with Stacks: de novo map on data 52, data 101, and others
- 123: Catalog assembled loci (tags) with Stacks: de novo map on data 52, data 101, and others
- 122: denovo_map.log with Stacks: de novo map on data 52, data 101, and others
- 121: Stacks SQL format with Stacks: de novo map on data 52, data 101, and others

Details for run 122 are shown in a box:

Identifying unique stacks; file 1 of 16 [SRR034310_CAAC]
Identifying unique stacks; file 2 of 16 [SRR034310_CACA]

Il existe différentes options avancées. Il est notamment possible de préciser :

File output options :

- VCF
- Genepop
- Structure
- FASTA
- PHASE/fastPHASE
- Beagle
- PLINK
- Phylip

Kernel-smoothing algorithm options :

-Il s'agit ici de lisser des valeurs de Fst obtenues en moyennant les valeurs dans une fenêtre de taille en paires de base définie. Cette option ne peut être utilisée que si un génome de référence est utilisé.

Genomic output options :

-l'export de chaque position nucléotidique (polymorphe ou non) de tous les individus d'une population dans un fichier. Il faudra alors préciser l'enzyme de restriction utilisée

Population advanced options :

-l'utilisation d'une whitelist. Dans ce cas, un fichier texte constitué d'une colonne reprenant les Stack_ID à considérer pour l'analyse sera donné en entrée.

-l'utilisation d'une blacklist. Dans ce cas, un fichier texte constitué d'une colonne reprenant les Stack_ID à ne pas considérer pour l'analyse sera donné en entrée.

-le pourcentage minimum d'individus par population pour pouvoir considérer le locus dans l'analyse

-le nombre minimum de populations dans lesquels le locus doit être présent pour le considérer dans l'analyse

-la profondeur minimum d'une pile par individu à un locus donné

-une fréquence d'allèle alternatif / minoritaire minimum pour calculer un Fst au locus considéré

-un type de correction à appliquer aux valeurs de Fst

-un seuil de p-valeur pour conserver les valeurs de Fst

-d'effectuer un ré-échantillonnage par bootstrap en précisant la précision et le nombre. Attention, cela n'est applicable qu'à des données initialement mappées contre un génome de référence (via l'utilisation de l'outil **Reference_map**).

The screenshot shows the Galaxy interface with several data tables and a phylogenetic tree:

- structure file with STACKS : populations on data 46 and data 52**: A table with columns 1 through 7. Row 1: # Stacks v1.18; Structure v2.3; July 27, 2015. Rows 2-6: SRR034310.CAAC, SRR034310.CAAC, SRR034310.CACA, SRR034310.CACA.
- vcf file with STACKS : populations on data 46 and data 52**: A VCF header table with columns Chrom, Pos, ID, Ref, Alt, Qual, Filter, Info, Format.
- sumstats.tsv with STACKS : populations on data 46 and data 52**: A table with columns 1 through 11. Rows 1-2: # 1 SRR034310.CAAC, SRR034310.CAAC, SRR034310.CAGT, SRR034310.CATG, SRR034310.CCA/. Rows 3-15: # 2 SRR034310.CTAG, SRR034310.CCTG, SRR034310.CTGA, SRR034310.CTTC, SRR034310.GGA/. Rows 3-15: # Batch ID, Locus ID, Chr, BP, Col, Pop ID, P Nuc, Q Nuc, N, P, OI.
- sumstats_summary.tsv with STACKS : populations on data 46 and data 52**: A table with columns 1 through 6. Rows 1-2: # Variant positions, # Pop ID Private Num Indv Var StdErr P Var StdErr Obs Het Var StdErr Obs Hom Var StdErr Exp. Rows 3-4: 0 0 845289 2302 2302 0.272333 5.7, 1 0 845720 2730 2730 0.322802 7.32.
- result.log with STACKS : populations on data 46 and data 52**: A table with columns 1 through 2. Rows 1-15: within_population incompatible_locus 182 un 5821 28 1, within_population incompatible_locus 1104 un 35118 21 1, within_population incompatible_locus 1332 un 42622 29 1, within_population incompatible_locus 2781 un 88988 27 1, within_population incompatible_locus 8189 un 262043 26 1, within_population incompatible_locus 14380 un 460158 29 1, within_population incompatible_locus 16575 un 530385 16 1, within_population incompatible_locus 20698 un 662325 20 1, within_population incompatible_locus 24560 un 785900 11 1, within_population incompatible_locus 30182 un 949021 28 1, within_population incompatible_locus 182 un 52027 2 2, within_population incompatible_locus 458 un 14631 6 2, within_population incompatible_locus 896 un 28666 25 2, within_population incompatible_locus 896 un 28672 31 2.
- phylogenetic tree**: A phylogenetic tree visualization.
- write only the first SNP per locus in Genepop and Structure outputs**: A note at the bottom left.
- with STACKS : populations on data 46 and data 52**, **sumstats_summary.tsv with STACKS : populations on data 46 and data 52**, **result.log with STACKS : populations on data 46 and data 52**, **49 lines**, **format: txt, database: ?**, **Fst kernel smoothing: off**: Notes at the bottom right.

Plusieurs types de fichiers sont générés :

Le fichier result.log

Il reprend les informations concernant les loci incompatible au sein de chacune des populations puis entre les populations.

NB : Pour consulter la sortie standard dans laquelle l'outil a écrit lors du déroulement du job, il faut cliquer sur le  d'un des jeux de données généré par l'outil puis sur "stdout".

```
Tool: STACKS : populations
Name: result.log with STACKS : populations on data 14 and data 19
Created: Fri Aug 8 13:56:58 2014 (UTC)
Filesize: 6.3 KB
Dbkey: ?
Format: txt
Galaxy
Tool      1.0.0
Version:
Tool
Version:
Tool
Standard stdout
Output:
Tool
Standard stderr
Error:
Tool Exit 0
Code:
API ID: 2f4933c7d721d5e8
Full Path: /omaha-beach/galaxy/58/database/files/000/dataset_224.dat
python /opt/shed_tools/dev-galaxy/genouest.org/repos/cmonjeau/stacks_toolsuite
Job    beach/galaxy/58/database/files/000/dataset_180.dat --vcf true --genepop false --
Command-beach/galaxy/58/database/files/000/dataset_225.dat --s /omaha-beach/galaxy/58/
Line:   beach/galaxy/58/database/files/000/dataset_228.dat --os=/omaha-beach/galaxy/
beach/galaxy/58/database/tmp
```

Le fichier batch_X.sumstats.tsv

Batch ID	The batch identifier for this data set.
Locus ID	Catalog locus identifier.
Chromosome	If aligned to a reference genome.
Basepair	If aligned to a reference genome. This is the alignment of the whole catalog locus. The exact basepair reported is aligned to the location of the RAD site (depending on whether alignment is to the positive or negative strand).
Column	The nucleotide site within the catalog locus.
Population ID	The ID supplied to the populations program, as written in the population map file.
P Nucleotide	The most frequent allele at this position in this population.
Q Nucleotide	The alternative allele.
Number of Individuals	Number of individuals sampled in this population at this site.
P	Frequency of most frequent allele.
Observed Heterozygosity	The proportion of individuals that are heterozygotes in this population.
Observed Homozygosity	The proportion of individuals that are homozygotes in this population.
Expected Heterozygosity	Heterozygosity expected under Hardy-Weinberg equilibrium.
Expected Homozygosity	Homozygosity expected under Hardy-Weinberg equilibrium.
pi	An estimate of nucleotide diversity.
Smoothed pi	A weighted average of p depending on the surrounding 3s of sequence in both directions (Seulement si "Kernel options" active).
Smoothed pi P-value	If bootstrap resampling is enabled, a p-value ranking the significance of p within this population (Seulement si "Kernel options" active).

FIS

The inbreeding coefficient of an individual (I)

relative to the subpopulation (S).

Smoothed FIS

A weighted average of FIS depending on the

surrounding 3s of sequence in both directions (**Seulement si "Kernel options" active**).

Smoothed FIS P-value

If bootstrap resampling is enabled, a p-value ranking the significance of FIS within this population (**Seulement si "Kernel options" active**).

Private allele

True (1) or false (0), depending on if this allele is only occurs in this population.

#	# SRR034310.CAAC_SRR034310.CACA_SRR034310.CAGT_SRR034310.CATG_SRR034310.CCAA_SRR034310.CCCC_SRR034310.CCGG_SRR034310.CCTT																									
#	Batch ID	Locus ID	Chr	BP	Col	Pop ID	P	Nuc	Q	Nuc	N	P	Obs Het	Obs Hom	Exp Het	Exp Hom	Pi	Smoothed Pi	Smoothed Pi	Pi	Fis	Smoothed Fis	Smoothed Fis	Pi	P-value	Private
1	1	un	20	19	1	G	T	7	0.928571	0.142857	0.857143	0.132653	0.867347	0.142857	0.00000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	1	un	20	19	2	G		8	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	4	un	104	7	1	T	G	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	4	un	104	7	2	T		1	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	4	un	108	11	1	G	A	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	4	un	108	11	2	A		1	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	4	un	111	14	1	G	A	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	4	un	111	14	2	A		1	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	4	un	115	18	1	T	C	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	4	un	115	18	2	C		1	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	4	un	117	20	1	C	A	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	4	un	117	20	2	A		1	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	15	un	474	25	1	A		6	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	15	un	474	25	2	A	G	7	0.928571	0.142857	0.857143	0.132653	0.867347	0.142857	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	17	un	525	12	1	G		6	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	17	un	525	12	2	G	A	5	0.8	0.4	0.6	0.32	0.68	0.355556	0.000000000	0	0.12555555555	0.00000000000	0	1	0	0.00000000000	0	1		
1	19	un	608	31	1	T	G	3	0.5	1	0	0.5	0.5	0.6	0.000000000	0	0.65666666667	0.00000000000	0	0	0	0.00000000000	0	0		
1	19	un	608	31	2	G	T	3	0.666667	0.666667	0.333333	0.444444	0.555556	0.533333	0.000000000	0	0.25333333333	0.00000000000	0	0	0	0.00000000000	0	0		
1	20	un	625	16	1	C	A	1	0.5	1	0	0.5	0.5	1	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	20	un	625	16	2	A		5	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	20	un	634	25	1	G	T	2	0.75	0.5	0.5	0.375	0.625	0.5	0.000000000	0	0.00000000000	0.00000000000	0	1	0	0.00000000000	0	1		
1	20	un	634	25	2	G		4	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	20	un	636	27	1	G	A	2	0.75	0.5	0.5	0.375	0.625	0.5	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	20	un	636	27	2	G	A	5	0.6	0.4	0.6	0.48	0.52	0.533333	0.000000000	0	0.25333333333	0.00000000000	0	0	0	0.00000000000	0	0		
1	24	un	755	18	1	T	C	7	0.571429	0.571429	0.428571	0.489796	0.510204	0.527473	0.000000000	0	0.08333333333	0.00000000000	0	0	0	0.00000000000	0	0		
1	24	un	755	18	2	T	C	8	0.625	0.5	0.5	0.46875	0.53125	0.5	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		
1	28	un	881	16	1	G		5	1	0	1	0	1	0	0.000000000	0	0.00000000000	0.00000000000	0	0	0	0.00000000000	0	0		

Accompagné du fichier sumstats_summary.tsv (disposé en 2 parties pour faciliter la visualisation) :

Variant positions
Pop ID Private Num Indv Var StdErr P Var StdErr Obs Het Var StdErr Obs Hom Var StdErr Exp Het Var StdErr Exp Hom Var StdErr Pi Var StdErr Fis Var StdErr
0 2665 4.15392 4.32523 0.0184255 0.874743 0.0285654 0.00149739 0.217513 0.0959329 0.00274409 0.782487 0.0959329 0.00274409 0.162009 0.0369834
1 6654 5.40317 5.47518 0.0197589 0.82038 0.0275747 0.00140223 0.31146 0.0922011 0.00256408 0.68854 0.0922011 0.00256408 0.239568 0.0303993 0.0
All positions (variant and fixed)
Pop ID Private Sites Variant Sites Polymorphic Sites % Polymorphic Loci Num Indv Var StdErr P Var StdErr Fis Var StdErr
0 2665 1395856 12740 6086 0.436005 4.24817 4.80505 0.00185536 0.998857 0.000402586
1 6654 1551311 14024 11359 0.732219 5.52033 6.39777 0.00203079 0.998376 0.000538288

Le fichier batch_X.fst_Y-Z.tsv

Batch ID The batch identifier for this data set.

Locus ID Catalog locus identifier.

Population ID 1 The ID supplied to the populations program, as written in the population map file.

Population ID 2 The ID supplied to the populations program, as written in the population map file.

Chromosome If aligned to a reference genome.

Basepair If aligned to a reference genome. This is the alignment of the whole catalog locus. The exact basepair reported is aligned to the location of the RAD site (depending on whether alignment is to the positive or negative strand).

Column The nucleotide site within the catalog locus.

Overall pi An estimate of nucleotide diversity across the two populations.

différentiation de populations, de déséquilibres génotypiques par paires de loci et d'isolation par la distance;

- PHASE / fastPHASE, pour servir de fichier d'entrée des logiciels PHASE et fastPHASE (<http://stephenslab.uchicago.edu/software.html>), dédié à la reconstruction d'haplotypes et à l'estimation des génotypes manquants à partir de données populationnelles ;
- beagle, pour servir de fichiers d'entrée du logiciel Beagle (<http://faculty.washington.edu/browning/beagle/beagle.html>), dédié à la détection de segment d'identity-by-descent, à l'imputation de marqueurs non génotypés ainsi qu'au phasage de génotype. Les fichiers générés sont répartis dans 2 archives. La première archive, nommée "markers.zip", contient un fichier par chromosome et dans chacun des fichiers, par colonne, sont mentionnés l'identifiant du SNP, la position correspondante puis les allèles trouvés. La seconde archive, nommée "unphase.zip" présente également un fichier par chromosome dont la seconde ligne, commençant par un "I" indique les individus (un par colonne), la troisième ligne, commençant par un "S", indique les populations et chaque ligne débutant par "M" reprend l'identifiant du SNP et les génotypes de chaque individus ;
- plink, pour servir de fichier d'entrée du logiciel Plink (<http://pngu.mgh.harvard.edu/~purcell/plink/>), dédié à l'analyse d'association génomique.

3. Utilisation d'un génome de référence

Après avoir récupéré la data collection « Demultiplexed reads... » générée précédemment par **STACKS : Process Radtags**, le fichier à 2 colonnes contenant les informations de population, il faut télécharger un génome de référence au choix via :

- la section Shared data/data libraries/RADseq/Stickleback population Genomics

name	description	data type	size	time updated (UTC)
Barcode_SRR034310.txt		tabular	320 bytes	2016-05-25 01:53 PM
Barcode_SRR034311.txt		tabular	320 bytes	2016-05-25 01:54 PM
Barcode_SRR034312.txt		tabular	352 bytes	2016-05-25 01:58 PM
Barcode_SRR034313.txt		tabular	352 bytes	2016-05-25 01:58 PM
Barcode_SRR034314.txt		tabular	352 bytes	2016-05-25 01:59 PM
Barcode_SRR034315.txt		tabular	88 bytes	2016-05-25 01:59 PM
Barcode_SRR034316.txt		tabular	352 bytes	2016-05-25 01:59 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034310/SRR034310.fastq.gz		fastq.gz	289.5 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034311/SRR034311.fastq.gz		fastq.gz	479.6 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034312/SRR034312.fastq.gz		fastq.gz	786.8 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034313/SRR034313.fastq.gz		fastq.gz	794.1 MB	2016-05-25 02:50 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034314/SRR034314.fastq.gz		fastq.gz	748.4 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034315/SRR034315.fastq.gz		fastq.gz	664.9 MB	2016-05-25 02:00 PM
ftp://ftp.sra.ebi.ac.uk/vol1/fastq/SRR034/SRR034316/SRR034316.fastq.gz		fastq.gz	855.2 MB	2016-05-25 02:00 PM
population_map		txt	272 bytes	2016-06-23 02:11 PM
Reference_genome_11_chromosomes.fa		fasta	223.9 MB	2016-05-25 01:52 PM

- l'UCSC (Stickelback gold, sequence)

- ou le dernier assemblage publié dans Genes – Genomes - Genetics (Glazer et al., 2015) (<http://datadryad.org/bitstream/handle/10255/dryad.89110/FileS6%20revisedAssemblyMasked.fa.zip?sequence=1>)

L'étape suivante consiste à aligner les lectures démultipléxées sur le génome. Pour cela, nous utiliserons l'outil **Map with BWA for Illumina** en prenant en fichier d'entrée la data collection « demultiplexed reads... Temps d'exécution ~ 2h (pour le jeu de données SRR034310 démultiplié soit 8 895 289 séquences).

-Générer un fichier de 2 colonnes exactement contenant sur chaque ligne le barcode et le nom de la population associé.

Nous pouvons ensuite exécuter **STACKS : Reference map**.

The screenshot shows the Galaxy web interface with the Stacks pipeline. The main panel displays the 'Stacks: reference map' tool configuration. It includes fields for 'Population', 'Specify a population map' (containing a step for '25: Regex Replace on data 16'), 'Minimum depth of coverage' (set to 3), and 'SNP Model Options'. The right panel shows the 'History' with a log entry for '79: Map with BWA for Illumina on collection 60: mapped reads'.

Une fois le job terminé, nous pouvons consulter les fichiers "result.log" et "catalog.*" (* couvrant trois types de fichiers déjà mentionnés dans la première partie consacrée à la détection de SNP, à savoir *snps*, *alleles* et *tags*).

The screenshot shows the Galaxy / GenOuest interface with four windows open:

- result.log with STACKS : Reference map on data 12 and data 9**: Displays the log output of the ref_map.pl script, showing the analysis of sequence reads from SRR034310.CCAA.sam and SRR034310.GGCC.sam, resulting in 45490 unique stacks and 460986 unique stacks respectively.
- catalog.tags with STACKS : Reference map on data 12 and data 9**: Displays a table of tag counts across nine positions (1-9) for each stack. The table shows consensus sequences and their frequencies.
- catalog.alleles with STACKS : Reference map on data 12 and data 9**: Displays a table of allele counts across nine positions (1-9) for each stack. The table shows the presence of different alleles (A, T, C, G) at each position.
- catalog.snps with STACKS : Reference map on data 12 and data 9**: Displays a table of SNP counts across nine positions (1-9) for each stack. The table shows the number of SNPs at each position.

Afin de spécifier plus d'options et de pouvoir filtrer les résultats, il est possible de ré exécuter le dernier module de **Stacks : Reference map**, à savoir **populations** sur la data collection "Full output ..." générée dans l'étape précédente.

Il existe différentes options avancées. Il est notamment possible de préciser :

File output options :

- VCF for SNPs (nous cocherons cette option ici)
- VCF for haplotypes (nous cocherons cette option ici)
- Genepop
- Structure
- FASTA
- PHASE/fastPHASE
- Beagle
- PLINK
- Phylip

Kernel-smoothing algorithm options :

-Il s'agit ici de lisser des valeurs de Fst obtenues en moyennant les valeurs dans une fenêtre de taille en paires de base définie. Cette option ne peut être utilisée que si un génome de référence est utilisé.

Genomic output options :

-l'export de chaque position nucléotidique (polymorphe ou non) de tous les individus d'une population dans un fichier. Il faudra alors préciser l'enzyme de restriction utilisée

Population advanced options :

-l'utilisation d'une whitelist. Dans ce cas, un fichier texte constitué d'une colonne reprenant les Stack_ID à considérer pour l'analyse sera donné en entrée.

-l'utilisation d'une blacklist. Dans ce cas, un fichier texte constitué d'une colonne reprenant les Stack_ID à ne pas considérer pour l'analyse sera donné en entrée.

-le pourcentage minimum d'individus par population pour pouvoir considérer le locus dans l'analyse

-le nombre minimum de populations dans lesquels le locus doit être présent pour le considérer dans l'analyse

-la profondeur minimum d'une pile par individu à un locus donné

-une fréquence d'allèle alternatif / minoritaire minimum pour calculer un Fst au locus considéré

-un type de correction à appliquer aux valeurs de Fst

-un seuil de p-valeur pour conserver les valeurs de Fst

-d'effectuer un ré-échantillonnage par bootstrap en précisant la précision et le nombre. Attention, cela n'est applicable qu'à des données initialement mappées contre un génome de référence (via l'utilisation de l'outil **Reference_map**).

Galaxy / GenOuest

Analyze Data Workflow Shared Data Visualization Admin Help User

result.log with STACKS : populations on data 12 and data 40

within_population	incompatible_locus	3013	chrII
within_population	incompatible_locus	5264	chrIV
within_population	incompatible_locus	7171	chrI
within_population	incompatible_locus	12355	chrVII
within_population	incompatible_locus	15858	chrVI
within_population	incompatible_locus	15963	chrVI
within_population	incompatible_locus	17107	chrV
within_population	incompatible_locus	20796	chrXIV
within_population	incompatible_locus	24423	chrXVI
within_population	incompatible_locus	24913	chrXVI
within_population	incompatible_locus	30327	chrXXI
within_population	incompatible_locus	30623	chrXX
within_population	incompatible_locus	37644	chrVII

sumstats_summary.tsv with STACKS : populations on data 12 and data 40

1	2	3	4	5	6	7	8
# Variant positions							
# Pop ID	Private	Indv	Num Var	StdErr P	Var StdErr	Obs Het	Var StdErr Exp Het Va
0 2831	4.78125	4.06315	0.0165524	0.89835	0.0231006	0.00124807	0.130377 0.0506276 0.001
1 8009	6.36743	3.2034	0.0145407	0.85009	0.0212006	0.00118292	0.199199 0.0459898 0.001
# All positions (variant and fixed)							
# Pop ID	Private	Sites	Variant Sites	Polymorphic Sites	% Polymorphic Loci	Num In	
0 2831	1424645		14830		6217	0.436389	5.029%
1 8009	1483785		15151		11715	0.789535	6.574%

sumstats.tsv with STACKS5 : populations on data 12 and data 40

1	2	3	4	5	6	7	8
# 1 SRR034310.CAAC,SRR034310.CACA,SRR034310.CAGT,SRR034310.CATG,							
# 2 SRR034310.CTAG,SRR034310.CTCT,SRR034310.CTGA,SRR034310.CTTC,							
# Batch ID	Locus ID	Chr	BP	Col	Pop ID	P Nuc	Q Nuc
1	7472	chrI	21269	6	1	C	
1	7472	chrI	21269	6	2	C	A
1	8431	chrI	56476	28	1	G	T
1	8431	chrI	56476	28	2	G	
1	8462	chrI	60099	13	1	T	G
1	8462	chrI	60099	13	2	T	

structure file with STACKS : populations on data 12 and data 40

1	2	3	4
# Stacks v1.18; Structure v2.3; August 11, 2015			
	7472_6	8431_28	
SRR034310.CAAC	1	2	3
SRR034310.CAAC	1	2	3
SRR034310.CACA	1	0	0
SRR034310.CAAC	1	2	3
SRR034310.CAAC	1	2	3
SRR034310.CACA	1	0	0
SRR034310.CACA	1	0	0

vcf file with STACKS : populations on data 12 and data 40

chrI	29413703	35955	G	A	.	PASS	NS=8;AF=0.688,0.312	GT:1
chrI	29479319	8175	G	C	.	PASS	NS=8;AF=0.938,0.062	GT:1
chrI	29479322	8175	T	C	.	PASS	NS=7;AF=0.643,0.357	GT:1
chrI	29545252	35959	G	A	.	PASS	NS=9;AF=0.778,0.222	GT:1
chrI	29545256	35959	C	T	.	PASS	NS=9;AF=0.944,0.056	GT:1
chrI	29571357	8181	T	C	.	PASS	NS=16;AF=0.969,0.031	GT:1
chrII	91853	34045	T	C	.	PASS	NS=8;AF=0.688,0.312	GT:1
chrII	91857	34045	C	A	.	PASS	NS=9;AF=0.556,0.444	GT:1
chrII	175924	1984	T	C	.	PASS	NS=10;AF=0.950,0.050	GT:1
chrII	189958	2070	G	T	.	PASS	NS=10;AF=0.850,0.150	GT:1
chrII	205708	33703	A	G	.	PASS	NS=4;AF=0.750,0.250	GT:1
chrII	222547	2350	C	T	.	PASS	NS=15;AF=0.967,0.033	GT:1
chrII	296369	2557	G	T	.	PASS	NS=13;AF=0.962,0.038	GT:1
chrII	315390	2569	A	T	.	PASS	NS=10;AF=0.950,0.050	GT:1
chrII	315424	33856	T	C	.	PASS	NS=10;AF=0.500,0.500	GT:1
chrII	323698	2575	G	A	.	PASS	NS=15;AF=0.967,0.033	GT:1
chrII	333224	2579	G	T	.	PASS	NS=12;AF=0.667,0.333	GT:1
chrII	471348	33902	T	C	.	PASS	NS=8;AF=0.625,0.375	GT:1
chrII	487238	33907	C	T	.	PASS	NS=7;AF=0.786,0.214	GT:1
chrII	512587	2704	T	C	.	PASS	NS=13;AF=0.577,0.423	GT:1
chrII	512639	33916	G	A	.	PASS	NS=12;AF=0.792,0.208	GT:1
chrII	526882	2717	C	A	.	PASS	NS=11;AF=0.955,0.045	GT:1
chrII	534613	2719	C	T	.	PASS	NS=14;AF=0.957,0.043	GT:1

Comme nous travaillons avec un génome de référence, il est possible de visualiser les fichiers VCF obtenus (SNPs et haplotypes) via un visualisateur de génome comme Trackster. Pour ce faire, il faut ajouter le génome de référence utilisé dans l'historique en tant que « custom build » puis affecter chaque vcf au « custom build » créé.

Pour créer un « custom build », rendez-vous dans User/Custom Builds

The screenshot shows the Galaxy web interface. On the left, a sidebar lists various tools and workflows, including 'stacks' and 'NGS: Building Loci' sections. The main area displays a table of existing builds:

Name	Key	Number of chroms/contigs	
pouletao2016	poulet_2016	1	Delete
Pf0-1	Pf0-1	1	Delete
pseudofluo	pseudo_pf1	1	Delete
Poulet AO	pouletAO	1	Delete
Stacks_pop_denovo	Stacks_pop_denovo	1	Delete
GL349685	GL349685	1	Delete
Poulet_AO	poulet_ao	1	Delete

A link 'Show loaded, system-installed builds' is at the bottom.

Add a Custom Build

New Build

Name (eg: Hamster): Stickelback2016

Key (eg: hamster_v1): Stickelback_2016

Definition:

- FASTA
- Len File
- Len Entry

3: Reference_genome_11_chromosome

FASTA format

This is a multi-fasta file from your current history that provides the genome sequences for each chromosome/contig in your build.

Here is a snippet from an example multi-fasta file:

```
>chr1
ATTATATATAAGACCAAGAGAGAATATTTGCCGG..
>chr2
GGCGGGCCGCGGCATATAGAACTACTCATTATATA..
...
```

Puis sélectionner le fichier de l'historique à utiliser et renseigner un nom (Stickelback2016 par exemple) et une clé (Stickelback_2016 par exemple) avant de valider en cliquant sur submit.

Vous pouvez à présent cliquer sur le ? du champ « database » de la prévisualisation de chacun des datasets au format VCF pour préciser le génome de référence Stickelback2016.

The screenshot shows the Galaxy web interface. The left sidebar lists various tools and workflows, including 'stacks' and 'NGS: Building Loci' sections. The main area shows the 'Edit Attributes' dialog for a dataset named 'SNPs in VCF format with Stacks: pf':

Attributes

Name: SNPs in VCF format with Stacks: pf

Info:

- Fst kernel smoothing: off
- Bootstrap resampling: off
- Percent samples limit per

Annotation / Notes:

Add an annotation or notes to a dataset; annotations are available when a history is viewed.

Database/Build:

Stickelback2016 (Stickelback_2016) [Custom]

The right side of the screen shows a history panel with a dataset named 'Stickelback 1.40 GCC Training RAD 5: population genomics with reference genome'. The dataset details are shown in a green box:

603: SNPs in VCF format with Stacks: populations on data 25, data 520, and others

1,481 lines, 10 comments
format: tabular, database: 2

Fst kernel smoothing: off
Bootstrap resampling: off
Percent samples limit per population: 0.5
Locus Population limit: 2
Minimum stack depth: 1
Log likelihood filtering: off;
threshold: 0
Minor allele frequency cutoff: 0.25
Maximum observed heterozygosity

Il sera peut être nécessaire de modifier le « datatype » en VCF s'il est en tabular. Si vous avez un pb avec BEDtools pour visualiser les données vcf, utiliser l'outil VCF To GFF3 pour convertir les VCF et ainsi les visualiser.

Références :

Ressources du groupe RADseq sur le HUB eBGO : <https://www.e-biogenouest.org/groups/radseq/resources>

Hohenlohe, P. A. et al. 2010. Population genomics of parallel adaptation in threespine stickleback using sequenced RAD tags. *PLoS Genetics* 6. 1-23.

<http://evomics.org/learning/genomics/stacks/> : Cours de Julian Catchen sur Evomics. Voir en particuliers les références mentionnées. Je les reprends ici pour information :

Core readings for the lecture and workshop

Amores, A., et al. 2011. Genome evolution and meiotic maps by massively parallel DNA sequencing. *Genetics* 188:799-808.

Broman, K. W. 2010. Genetic map construction with R/qtl. Univ. Wisc. Technical Report #214.

Catchen, J. M. et al. 2011. Stacks: building and genotyping loci de novo from short-read sequences. *G3: Genes, Genomes and Genetics* 1; 171-182.

Davey, J. W., et al. 2011. Genome-wide genetic marker discovery and genotyping using next-generation sequencing. *Nature Reviews Genetics* 12:499-510.

Etter, P. D., et al. 2011. SNP Discovery and Genotyping for Evolutionary Genetics using RAD sequencing. in *Molecular Methods in Evolutionary Genetics*, Rockman, M., and Orgonogozo, V., eds. (in press).

Ekblom, R., and J. Galindo. 2010. Applications of next generation sequencing in molecular ecology of non-model organisms. *Heredity* 107:1-15.

Hohenlohe, P. A. et al. 2010. Population genomics of parallel adaptation in threespine stickleback using sequenced RAD tags. *PLoS Genetics* 6. 1-23.

NGS population genomics background, concepts and statistical considerations

Broman, K. W., et al. 2003. R/qtl: QTL mapping in experimental crosses. *Bioinformatics* 19:889-890.

Broman, K. W., and S. Sen. 2009. *A Guide to QTL Mapping with R/qtl*. Springer.

Gompert, Z., and C. A. Buerkle. 2011a. A hierarchical Bayesian model for next-generation population genomics. *Genetics* 187:903-917.

Gompert, Z., and C. A. Buerkle. 2011b. Bayesian estimation of genomic clines. *Molecular Ecology* 20:2111-2127.

Lynch, M. 2009. Estimation of allele frequencies from high-coverage genome-sequencing projects. *Genetics* 182:295-301.

Nielsen, R., et al. 2005. Genomic scans for selective sweeps using SNP data. *Genome Research* 15:1566-1575.

Hohenlohe, P. A., et al. 2010. Using population genomics to detect selection in natural populations: Key concepts and methodological considerations. *International Journal of Plant Sciences* 171:1059-1071.

Stapley, J., et al. 2010. Adaptation genomics: the next generation. *Trends in Ecology and Evolution* 25:705-712.

Luikart, G., et al. 2003. The power and promise of population genomics: from genotyping to genome typing. *Nature Reviews Genetics* 4:981-994.

Nielsen, R., et al. 2011. Genotype and SNP calling from next-generation sequencing data. *Nature Reviews Genetics* 12:443-451.

Genetic mapping using RRL and RAD sequencing

Altshuler, D., et al. 2000. An SNP map of the human genome generated by reduced representation shotgun sequencing. *Nature* 407:513-516.

Baxter, S. W., et al. 2011. Linkage mapping and comparative genomics using next-generation RAD sequencing of a non-model organism. *PLoS ONE* 6:e19315.

Chutimanitsakun, Y., et al. 2011. Construction and application for QTL analysis of a Restriction Site Associated DNA (RAD) linkage map in barley. *BMC Genomics* 12: 1-13.

Gore, M. A., et al. 2009. A first-generation haplotype map of maize. *Science* 326:1115-1117.

RAD-seq genotyping methodology

Baird, N. A., et al. 2008. Rapid SNP discovery and genetic mapping using sequenced RAD markers. *PLoS ONE* 3:e3376.

Emerson, K. J., et al. 2010. Resolving postglacial phylogeography using high-throughput sequencing. *Proceedings of the National Academy of Sciences* 107:16196-16200.

Etter, P. D., et al. 2011. Local De Novo Assembly of RAD Paired-End Contigs Using Short Sequencing Reads. *PLoS ONE* 6:e18561

Hohenlohe, P. A., et al. 2011. Next-generation RAD sequencing identifies thousands of SNPs for assessing hybridization between rainbow and westslope cutthroat trout. *Molecular Ecology Resources* 11 Suppl 1:117-122.

Miller, M. R., et al. 2007. Rapid and cost-effective polymorphism identification and genotyping using restriction site associated DNA (RAD) markers. *Genome Research* 17:240-248.

Willing, E. M., et al. 2011. Paired-end RAD-seq for de novo assembly and marker design without available reference. *Bioinformatics* 27:2187-2193.

Other reduced representation library (RRL) methodologies

Andolfatto, P., et al. 2011. Multiplexed shotgun genotyping for rapid and efficient genetic mapping. *Genome Research* 21:610-617.

Elshire, R. J., et al. 2011. A Robust, Simple Genotyping-by-Sequencing (GBS) Approach for High Diversity Species. *PLoS ONE* 6:e19379.

Rigola, D., et al. 2009. High-Throughput Detection of Induced Mutations and Natural Variation Using KeyPoint™ Technology. *PLoS ONE* 4:e4761.

van Orsouw, N. J., et al. 2007. Complexity reduction of polymorphic sequences (CRoPS): a novel approach for large-scale polymorphism discovery in complex genomes. *PLoS ONE* 2:e1172.

van Tassell, C. P., et al. 2008. SNP discovery and allele frequency estimation by deep sequencing of reduced representation libraries. *Nature Methods* 5:247-252.

Useful links

1. Quality scores

1. http://en.wikipedia.org/wiki/FASTQ_format
2. http://en.wikipedia.org/wiki/Phred_quality_score
3. <http://www.phrap.com/phred/>
4. http://www.illumina.com/truseq/quality_101/quality_scores.ilmn

2. Basic Unix, R and PERL commands

1. <http://mally.stanford.edu/~sr/computing/basic-unix.html>
2. http://korflab.ucdavis.edu/Unix_and_Perl/
3. <http://www.r-project.org/>
4. <http://cran.r-project.org/doc/manuals/R-intro.html>
5. <http://manuals.bioinformatics.ucr.edu/home/programming-in-r>

3. Stacks download and tutorials

1. <http://creskolab.uoregon.edu/stacks/>
4. Great site for information on next gen sequencing

1. <http://seqanswers.com/>