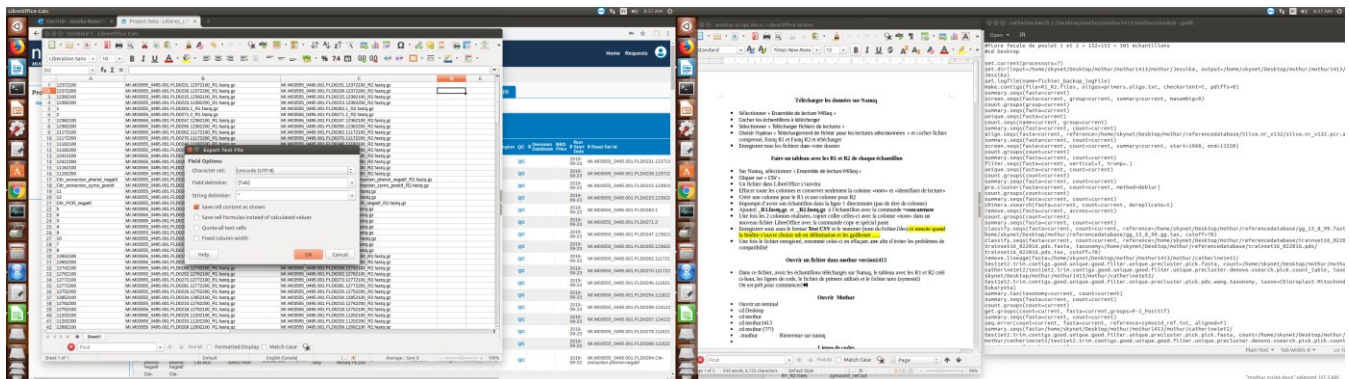


Télécharger les données sur Nanuq

- Sélectionner « Ensemble de lecture MiSeq »
- Cocher les échantillons à télécharger
- Sélectionner « Télécharger fichiers de lectures »
- Choisir l'option « Téléchargement de fichier pour les lectures sélectionnées » et cocher fichier compressé, Fastq R1 et Fastq R2 et télécharger
- Enregistrer tous les fichiers dans votre dossier

Faire un tableau avec les R1 et R2 de chaque échantillon

- Sur Nanuq, sélectionner « Ensemble de lecture MiSeq »
- Cliquer sur « CSV »
- Un fichier dans LibreOffice s'ouvrira
- Effacer toute les colonnes et conserver seulement la colonne «nom» et «identifiant de lecture»
- Créer une colonne pour le R1 et une colonne pour R2
- Important d'avoir son échantillon dans la ligne 1 directement (pas de titre de colonne)
- Ajouter **_R1.fastq.gz** et **_R2.fastq.gz** à l'échantillon avec la commande **=concatenate**
- Une fois les 2 colonnes réalisées, copier coller celles-ci avec la colonne «nom» dans un nouveau fichier LibreOffice avec la commande copy et spécial paste



- Enregistrer sous sous le format **Text CSV** et le nommer (nom du fichier.files) et ensuite quand la fenêtre s'ouvre choisir tab en délimitation et les guillemet (ex. sur la photo)
- Une fois le fichier enregistré, renommé celui-ci en effaçant **.csv** afin d'éviter les problèmes de compatibilité

Ouvrir un fichier dans mothur version1413

- Dans ce fichier, avoir les échantillons téléchargés sur Nanuq, le tableau avec les R1 et R2 créé ci-haut, les lignes de code, le fichier de primers utilisés et le fichier taxo (zymostd)
On est prêt pour commencer ☺

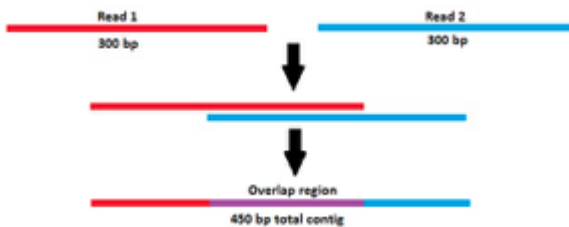
Ouvrir Mothur

- Ouvrir un terminal
- `cd Desktop`
- `cd mothur`
- `cd mothur1413`
- `cd mothur`
- `./mothur` Bienvenue sur mothur
-

Lignes de codes

- `set.current(processors=7)`
- `set.dir(input=/home/skynet/Desktop/mothur/mothur1413/mothur/test, output=/home/skynet/Desktop/mothur/mothur1413/mothur/test)`
- `set.logfile(name=test_logFile)`
- **`make.contigs`**(file=test.files, oligos=primers.oligo.txt, checkorient=t, pdiffs=0)

Combiner les 2 set de reads ensemble (R1 et R2) voir image ci-dessous



- `summary.seqs(fasta=current)`
Permet d'avoir un résumé des séquences etc.

- **`screen.seqs`**(fasta=current, group=current, summary=current, maxambig=0)

Supprime les séquences ayant des bases ambiguës (et celle plus longue que 275pb ??)

- `count.groups(group=current)`
- `summary.seqs(fasta=current)`

- **`unique.seqs`**(fasta=current)

Permet de sélectionner que les séquences uniques

L'analyse de cette séquence qui sera appliquée à chaque séquence identique de l'ensemble des données

- **`count.seqs`**(name=current, group=current)

Permet de simplifier les fichiers

Création d'une table dans laquelle les lignes sont les noms des séquences uniques et les colonnes les noms des groupes. La table est ensuite remplie avec le nombre de fois que chaque séquence unique apparaît dans chaque groupe

- `summary.seqs(fasta=current, count=current)`
- `align.seqs(fasta=current, reference=/home/skynet/Desktop/mothur/referencedatabase/Silva.nr_v132/silva.nr_v132.pcr.align, flip=t)`

Utiliser Silva pour alignement. GreenGenes contient des séquences 16S incomplètes (et non complètes) qui rendent l'alignement difficile et sujet aux erreurs.

flip = T afin que mothur vérifie à la fois le complément direct et inverse de nos séquences.

- **summary.seqs**(fasta=current, count=current)

| | Start | End | NBases | Ambigs | Polymer | NumSeqs |
|------------------|--------|---------|---------|--------|---------|---------|
| Minimum: | 13125 | 23439 | 250 | 0 | 3 | 1 |
| 2.5%-tile: | 13862 | 23444 | 252 | 0 | 3 | 3139 |
| 25%-tile: | 13862 | 23444 | 253 | 0 | 3 | 31386 |
| Median: | 13862 | 23444 | 253 | 0 | 4 | 62771 |
| 75%-tile: | 13862 | 23444 | 253 | 0 | 4 | 94156 |
| 97.5%-tile: | 13862 | 23444 | 254 | 0 | 5 | 122402 |
| Maximum: | 13875 | 26147 | 294 | 0 | 8 | 125540 |
| Mean: | 13862 | 23444.1 | 252.889 | 0 | 3.82739 | |
| # of Seqs: | 20469 | | | | | |
| total # of seqs: | 125540 | | | | | |

- screen.seqs(fasta=current, count=current, summary=current, **start=1968, end=11550**)
- count.groups(count=current)
- summary.seqs(fasta=current, count=current)
- **filter.seqs**(fasta=current, vertical=T, trump=.)

Les alignements générés par rapport aux alignements de référence (par exemple, à partir de RDP, SILVA ou de greengenes) ont souvent des colonnes où chaque caractère est un "." ou un "-". Ces colonnes ne sont pas incluses dans le calcul des distances car elles ne contiennent aucune information. En supprimant ces colonnes, le calcul d'un grand nombre de distances est accéléré.

- **unique.seqs**(fasta=current, count=current)

Creation de sequences redondantes avec etape precedente donc on refait un unique.seq

- count.groups(count=current)
- summary.seqs(fasta=current, count=current)

Les séquence devrait être revenue à 1 et contenir environ 253 pb

- **pre.cluster**(fasta=current, count=current, method=deblur)

Permet de pré regrouper les séquences qui sont très similaires

Cette commande divise les séquences par groupe, et les trie en abondance (de la plus abondante a la moins abondante)

- count.groups(count=current)
- summary.seqs(fasta=current, count=current)

- **chimera.vsearch**(fasta=current, count=current, dereplicate=t)

Permet d'identifier les chimères (erreur de polymérisation, Une polymérase haute fidélité a moins de chance de tomber et de créer des chimères et moins de cycles entraînent également moins de chimères.)

- **remove.seqs**(fasta=current, accnos=current)
Permet de supprimer les chimeres
- count.groups(count=current)
- summary.seqs(fasta=current, count=current)
- **classify.seqs**(fasta=current, count=current,
reference=/home/skynet/Desktop/mothur/referencedatabase/gg_13_8_99.fasta,
taxonomy=/home/skynet/Desktop/mothur/referencedatabase/gg_13_8_99.gg.tax, cutoff=70)
- **classify.seqs**(fasta=current, count=current,
reference=/home/skynet/Desktop/mothur/referencedatabase/trainset16_022016.pds/trainset16_022016.pds.fasta,
taxonomy=/home/skynet/Desktop/mothur/referencedatabase/trainset16_022016.pds/trainset16_022016.pds.tax, cutoff=70)

Permet de classer les sequences

- **remove.lineage**(fasta=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta
output ,
count=test.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.count_table,
taxonomy=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.taxonomy,
taxon=**Chloroplast-Mitochondria-unknown-Eukaryota**)

Permet de supprimer les sequences correspondant aux eucaryotes, archees, chloroplastes et inconnus

- summary.tax(taxonomy=current, count=current)
- summary.seqs(fasta=current, count=current)
- count.groups(count=current)
- get.groups(count=current, fasta=current, groups=**DNACommunity**)
- summary.seqs(fasta=current, count=current) (Pour communauté artificielle)
- seq.error(count=current, fasta=current, reference=zymo_std_ref.txt, aligned=F)
- summary.seqs(fasta=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.fasta,
count=test.trim.contigs.good.unique.good.filter.unique.precluster.denovo.vsearch.pick.pick.count_table)
- count.groups(count=current)
- **dist.seqs**(fasta=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.pick.fasta)

Permet de calculer la distance entre les sequences (combien il y a de difference entre chacune d elle)

- cluster(column=current, count=current, method=opti)
- **classify.otu**(list=current, count=current,
taxonomy=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.pds.wang.taxonomy,
label=0.03)

- `rename.file(input=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.pick.agc.unique_list.0.03.cons.taxonomy, new=Test_OTU_class_pds.cons.taxonomy)`
- `classify.otu(list=current, count=current, taxonomy=test.trim.contigs.good.unique.good.filter.unique.precluster.pick.gg.wang.taxonomy, label=0.03)`
- `rename.file(input=METTRE LE NOM DU FCHIER CREER PAR CLASSIFY OTU.cons.taxonomy, new=Test_OTU_class_gg.cons.taxonomy)`
- `make.shared(list=current, count=current, label=0.03)`
- `count.groups(shared=current)`
- `rarefaction.single(shared=current, calc=sobs-coverage, freq=500)`

Permet de voir si on a assez de sequence pour représenter la diversité des échantillons