# Projet Bioperl – M1 janvier 2015

Ce projet traite les données de 33 génomes mitochondriaux complets de plantes. Il comporte 6 parties qui s'enchaînent. Le dossier Correction contient tous les fichiers que vous devez utiliser et/ou produire. Donc vous pouvez développer les parties dans l'ordre que vous préférez, en utilisant les données intermédiaires du dossier Correction si besoin. Utilisez aussi les fichiers du dossier Correction pour vérifier vos résultats.

Envoyez-moi vos programmes par mail sophie.gallina@univ-lille1.fr

Critères d'évaluation :

1. Les programmes ne doivent pas planter
2. Ils doivent générer des données similaires à celles du dossier Correction
3. Ils doivent être « propres », ie utiliser des fonctions, déclarer les variables etc...
4. Ils doivent avoir des commentaires
5. Ils peuvent ne pas implémenter toutes les fonctionnalités (par exemple, dans la partie 4, si vous ne voyez pas comment trier ou ajouter la colonne supplémentaire, faites le tableau sans ces fonctionnalités et signalez-le dans le mail).

## *Partie 1*

Récupérer des fichiers au format genbank pour une liste de numéros d'accession.

- Lire le fichier 'accessions_list.txt' pour avoir la liste des numéros d'accession
  - Ignorer les lignes vides et celles commençant par # (utilisées pour les commentaires)
- Récupérer chaque accession séparément
- Écrire les données dans un fichier nommé <ACCESSION>.gbk, dans le dossier 'genbank'
  - Exemple : genbank/NC_008285.gbk
  - Si le fichier existe déjà, ne pas le télécharger une nouvelle fois
  - Si le dossier 'genbank' n'existe pas, le créer

Programme : part1.pl
Input : accessions_list.txt
Output : genbank/<ACCESSION>.gbk

## *Partie 2*

Pour chaque accession, extraire les séquences codantes des gènes dans un fichier fasta.

- Lire le fichier 'accessions_list.txt' pour avoir la liste des numéros d'accession
- Pour chaque accession, lire le fichier au format genbank dans le dossier 'genbank'
- Traiter uniquement les CDS ayant un tag 'gene'
  - Ignorer les ORF et les gènes blacklistés dans le fichier skip_genes.txt.
  - Pour les autres, extraire la séquence nucléique de la partie codante
- Écrire ces séquences dans un fichier <SPECIE>.fasta placé dans le dossier fasta
  - L'identifiant de la séquence sera le nom du gène, et la description sera vide.
- Créer le fichier accessions_table.txt avec le numéro d'accession, le nombre de gènes et le nom de l'espèce

Conseils

- Utiliser la méthode spliced_seq() pour extraire la séquence nucléique de la partie codante
- Utiliser les méthodes id() et description() pour gérer les informations qui seront placées sur le ligne '> XXX' du fichier fasta

- Le nom de l'espèce est utilisée pour créer le fichier <SPECIE>.fasta, donc attention aux caractères interdits dans les noms de fichiers (remplacez-les par le caractère _).

Remarques

- Pour certaines séquences, vous pouvez avoir des messages d'avertissement, par exemple :

```
 -------------------- WARNING ----------------------
MSG: feature strand is different from location strand!
-------------------------------------------------
```

Programme : part2.pl
Input : accessions_list.txt, genbank/<ACCESSION>.gbk
Output : accessions_table.txt, fasta/<SPECIE>.fasta

## *Partie 3*

Collecter toutes les séquences des fichiers fasta/<SPECIE>.fasta afin de générer un fichier unique nommé fasta/bank1.fasta. Ce fichier pourra être utilisé pour générer une banque et « blaster » d'autres séquences sur cette banque. Le nom de chaque séquence doit donc être adapté pour contenir le nom de l'espèce en plus du nom du gène (pensez à utiliser la méthode id() pour modifier les identifiants des séquences). Le fichier bank1.fasta devrait contenir 882 séquences.

*Exemple : identifiants des séquences du gène nad9 dans les fichiers par espèce :*
grep nad9 fasta/*
fasta/Arabidopsis_thaliana.fasta:>nad9
fasta/Brassica_carinata.fasta:>nad9
….
*et identifiants des séquences du gène nad9 dans le fichier global*
grep nad9 fasta/bank1.fasta
fasta/bank1.fasta:>nad9_Arabidopsis_thaliana
fasta/bank1.fasta:>nad9_Brassica_carinata
...

Programme : part3.pl
Input : accessions_table.txt, fasta/<SPECIE>.fasta
Output : fasta/bank1.fasta

## *Partie 4*

Générer un état résumé par gènes et par espèce.

- Lire toutes les données à partir des fichiers fasta par espèce fasta/<SPECIE>.fasta
- Générer un fichier csv contenant un tableau ( les gènes en lignes et les espèces en colonnes).
- Chaque cellule du tableau indiquera la taille du gène en bp ou 0 si le gène est manquant.
- La dernière colonne indiquera le pourcentage d'espèces pour lesquelles ce gène est présent.
- Les lignes et les colonnes seront triées par ordre alphabétique.

Conseils

- Utiliser un tableau associatif pour collecter les données de chaque paire (espèce, gène)
- Utiliser un second tableau associatif pour collecter la liste des noms de gènes.

Programme part4.pl
Input : accessions_table.txt, fasta/<SPECIE>.fasta,
Output : summary.csv

Exemple d'état (seules les premières lignes et quelques colonnes sont montrées) :

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | | Arabidopsis_thaliana | Beta_macrocarpa | … | Zea_perennis | % of species where gene is present |
| 2 | CcmFn | 0 | 0 | … | 0 | 3.03 |
| 3 | ND2 | 0 | 0 | … | 0 | 3.03 |
| 4 | RNA_pol | 0 | 0 | … | 0 | 3.03 |
| 5 | atp1 | 1524 | 1521 | … | 1527 | 90.91 |
| 6 | atp1-1 | 0 | 0 | … | 0 | 3.03 |
| 7 | atp1-a | 0 | 0 | … | 0 | 3.03 |
| 8 | atp1-a1 | 0 | 0 | … | 0 | 3.03 |
| 9 | atp1-a2 | 0 | 0 | … | 0 | 3.03 |
| 10 | atp1-b | 0 | 0 | … | 0 | 3.03 |
| 11 | atp4 | 0 | 597 | … | 666 | 66.67 |
| 12 | atp6 | 0 | 774 | … | 0 | 87.88 |
| 13 | atp6-1 | 1158 | 0 | … | 1233 | 12.12 |
| 14 | atp6-2 | 1050 | 0 | … | 0 | 6.06 |
| 15 | atp8 | 0 | 489 | … | 462 | 72.73 |
| 16 | atp8-1 | 0 | 0 | … | 0 | 3.03 |
| 17 | atp8-2 | 0 | 0 | … | 0 | 3.03 |
| 18 | atp9 | 258 | 225 | … | 225 | 96.97 |

## *Partie 5*

Le fichier XXX.fasta contient un assemblage de-novo pour une nouvelle espèce. Utiliser la banque blast générée dans la partie 3 pour annoter ce nouvel assemblage. Avant d'écrire le programme part5.pl, vous devez réaliser les étapes suivantes :

- Si les outils blast sont installés sur votre système, créez les index blast pour la banque de gènes, puis blastez les contigs du nouveau génome
  - cd fasta ; formatdb -p F -i bank1.fasta -n bank1 ; cd ..
  - blastn -db fasta/bank1 -evalue 0.01 -query XXX.fasta -out XXX_blast_out.txt
- Sinon, copiez le fichier résultat à partir du dossier Correction
  - cp Correction/XXX_blast_out.txt .

Vous pouvez maintenant filtrer les résultats de blast pour ne conserver que les hits ayant au moins 95 % d'identité sur au moins 300bp. Vous devriez obtenir 399 hits.

Programme : part5.pl
Input :XXX_blast_out.txt
Output : XXX_blast.csv

## *Partie 6*

1. Combien de gènes sont communs à toutes les espèces (ie présents dans 100 % des 33 espèces utilisées) ? (Cf résultats de la partie 4)
2. Parmi les gènes communs aux 33 espèces combien sont retrouvés dans l'assemblage de l'espèce XXX ? (Cf résultats de la partie 5)
3. Parmi les gènes trouvés dans l'assemblage XXX, lesquels semblent dupliqués, tripliqués ou plus par rapport à Arabidopsis thaliana ?

Dans le cas de ce projet, le nombre de données est limité, vous pourriez donc réponde à ces questions simplement en regardant les fichiers de résultats. Essayez cependant de développer un script perl pour produire les réponses.