Inferindo filogenias

Tabla de contenido

Árbol de máxima verosimilitud

Para inferir un árbol de máxima verosimilitud usaremos RAxML 8.2.11 - manual.

Para empezar, usaremos el archivo vcf de *Heliconius* ya filtrado y solo con sitios variables para generar una hipótesis de relación entre los 18 individuos.

Antes de empezar, vamos a crear una carpeta llamada "Filogenómica" (con el comando `mkdir`), y dentro de esta carpeta vamos crear una otra carpeta llamada "raxml". Entre en la carpeta "raxml".

Primero, necesitamos excluir los sitios que RAxML pensará que son monomórficos. Para eso, vamos nos asegurar de que los datos solo tengan sitios bialélicos y eliminaremos los sitios que tengan heterocigosidad muy alta (que no estan en HW equilibrium).

```
module load vcftools

file="<name_of_file>"

# Acá vamos calcular HW con valores de p
vcftools --gzvcf $file.vcf.gz --hardy --max-alleles 2 --out $file

# Extraer sitios para mantener
awk '{split($3,gen,"/"); \
    if(gen[1]!=0 && gen[3]!=0 && $8>1e-5) \
    print $1"\t"$2}' $file.hwe > ${file}_sites_toKeep

# Nuevo VCF con sitios para mantener
vcftools --gzvcf ../bams_subsampled_chr/$file.vcf.gz \
    --positions ${file}_sites_toKeep \
    --recode --stdout | gzip > $file.altHom.vcf.gz
```

Ahora necesitamos transformar el archivo vcf en un archivo phylip que és el formato aceptado por RAxML. Para eso clonaremos un repositorio de github donde tiene un script de Python que hace esta conversión. - Dentro de la carpeta "raxml", ejecutar este comando: `git clone https://github.com/joanam/scripts.git` Al hacer esto, debería aparecer una carpeta llamada "scripts". Dentro de "scripts" hay varios archivos, y usaremos uno llamado "vcf2phylip.py" para transformar nuestros datos. En la carpeta "raxml" execute estes comandos:

```
module load python

file="<name_of_file>"

python ./scripts/vcf2phylip.py -i $file.altHom.vcf.gz -o "${file}.phylip"
```

Ahora que tenemos nuestro archivo phylip podemos hacer un *sbatch* para ejecutar RAxML.

Los parámetros que usaremos: `:-T` especifica cuantos procesadores (SOLO VERSIÓN PTHREADS!) `:-s` especifica el nombre del archivo de entrada (phylip o fasta) `:-f a`vamos hacer un análisis de bootstrap rápida y la busqueda por la mejor arbole de ML en la misma ejecución `:-N` vas calcular 100 bootstraps `:- m` modelo de evolución. Ao usar *ASC_* usted indica que desea aplicar una corrección a el sesgo de verificación (**ascertainment bias??**) a los cálculos de verosimilitud. Para datos de SNPs vamos utilizar el modelo gamma de heterogeneidad de tasas con corrección de sesgo de verificación y optimización de las tasas de sustitución (ASC_GTRGAMMA). Con esto modelo necesitas especificar el tipo de corrección (siguiente parámetro) `:--asc-corr` permite especificar el tipo de sesgo de confirmación que desea utilizar (predeterminado: lewis) `:-o` especifique el nombre de un solo grupo externo o una lista separada por comas de grupos externos `:-n` especifica el nombre del archivo de salida `:-p` especifica un *random seed* para la inferencia inicial de parsimonia. Para todas las opciones en RAxML que requieran algún tipo de aleatorización, se debe especificar esta opción. `:-x` especifica un *random seed* para el bootstrap rápido

El código para el *sbatch* és:

```
module load raxml/8.2.11

file="<name_of_file>"

# Prueba rápida con 100 bootstraps
raxmlHPC-PTHREADS-AVX -T 2 \
    -p 12345 -x 12345 \
    -s ${file}.phylip \
    -m ASC_GTRGAMMA --asc-corr=lewis \
    -N 100 -f a \
    -o H.eth.aer.JM67,H.hec.fel.JM273,H.num.num.MJ09.4125,H.num.sil.MJ09.4184,H.par.ser.JM20
    -n RAXML_100boot.out
```

Memoria necesaria: 90mb

Tiempo de ejecución: ~25min

Output: cinco archivos `RAxML.*.<output_name>.out` (bestTree, bipartitions, bipartionsBranchLabels, bootstrap, info). Para mirar árboles és bueno tener el software FigTree en tu computadora. Descárguelo `RAxML_bipartitions.RAXML_100boot.out` en tu computadora y abra el

archivo en FigTree.

Árbol de especies

Árbol de especies se puede definir como el patrón de ramificación de linajes de especies a través del proceso de especiación. Cuando las comunidades reproductivas se dividen por especiación, las copias de genes dentro de estas comunidades también se dividen en paquetes de descendencia. Dentro de cada paquete, los árboles genéticos continúan ramificándose y descendiendo a través del tiempo. Por lo tanto, los árboles genéticos (de genes) están contenidos dentro de las ramas de la filogenia de las especies (Maddison, 1997).

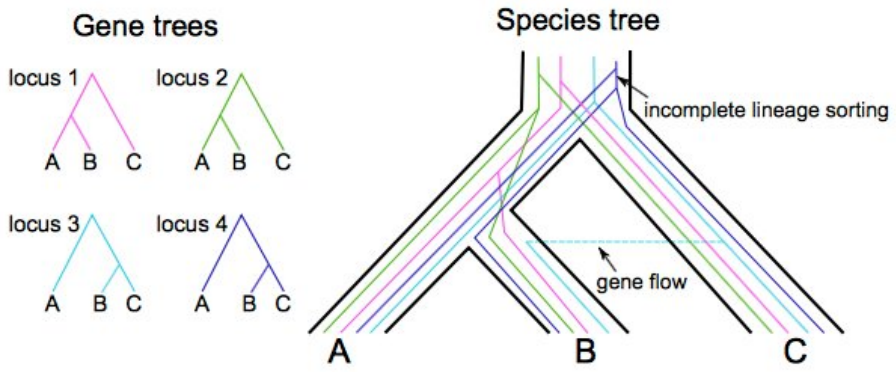

Figure 1: Conflicto entre árboles de genes y de especie - figura de Marin et al., 2020

Para inferir la árbole de especie entre las especies de *Heliconius*, vamos usar la metodología de SVDquartets implementado en PAUP que inferi relaciones entre cuartetos de taxones bajo el modelo coalescente para estimar la árbole de especies.

Antes de empezar necesitamos bajar el software: 1. Vamos generar una carpeta llamada "PAUP" dentro de la carpeta "Filogenómica". 2. dentro de la carpeta PAUP, tenemos que hacer el download del software PAUP (la versión para Linux): `wget http://phylosolutions.com/paup-test/paup4a168_centos64.gz` 3. para descomprimir PAUP: `gunzip paup4a168_centos64.gz` 4. y hacer que PAUP sea ejecutable: `chmod +x paup4a168_centos64`

Ahora estamos listos para empezar :) 1. PAUP acepta como input un archivo en el formato *nexus*. Para esto, modificaremos manualmente nuestro archivo *phylip* utilizado en RAxML a uno *nexus*. Usando `nano` cambiaremos el encabezado del archivo phylip a:

```
#NEXUS
begin data;
    dimensions ntax=18 nchar=52125;
    format datatype=nucleotide gap=- missing=N matchchar=.;
matrix
```

y agregaremos código con algunas instrucciones a PAUP al final del archivo

```
;
End;

begin sets;
    taxpartition Heliconius =
        H.eth.aer       : 1,
        H.hec.fel       : 2,
        H.melp.malleti  : 3-6,
        H.num.num       : 7,
        H.num.sil       : 8,
        H.par.ser       : 9,
        H.par.spn       : 10,
        H.tim.fln       : 11-14,
        H.tim.thx       : 15-18;
end;

begin paup;
    cd *;   [ sets the default directory to the one containing this file]

    svdq speciestree=y taxpartition=Heliconius evalQuartets=all nthreads=2 bootstrap=y nreps
    savetrees;

end;
```

- **Taxpartition** és donde diremos al software la especie a que pertenece cada muestra, con la **partition** *Heliconius*
- **speciestree** especifica realizar un análisis de árboles de especies (y)
- **evalQuartets** hará una búsqueda exhaustiva con todos los cuartetos (all)
- **nthreads** número de procesadores (2)
- **bootstrap** si hará bootstrap (y)
- **nreps** número de réplicas de bootstrap (100)
- **savetrees** salva la árbol

Ahora estamos listos para ejecutar PAUP:

```
file="heliconius.optixscaf.SNPS.FL2.nex"

# Running svdquartets
./paup4a168_centos64 $file -n
```

Memoria necesaria: ~20mb

Tiempo de ejecución: >2min (no necesita hacer sbatch)

Output: un archivo con la árbole de especie (.tre). Hacer lo mismo que con RAxML: bajar para tu computadora y mirar en FigTree.
