Comparación De Algoritmos Para La Generación De Árboles Filogenéticos

Adriana Michel Ávila García
Fernando Márquez Pérez

Facultad de Ciencias UNAM
Genómica Computacional - 2023-1

Fecha de Entrega: 14 de Diciembre de 2022

Introducción

Pandas, ¿Osos o mapaches?

Caracterizado por su apariencia y su pelaje blanco y negro, el **panda gigante** (*Ailuropoda melanoleuca*), también conocido simplemente como panda o oso panda, es una de las especies más famosas de China que actualmente se considera una especia vulnerable debido a la intervención humana es sus hábitats naturales.

La taxonomía, ciencia que se centra en nombrar, caracterizar y clasificar organismos y que nos permite entender la relación entre estos organismos y la variedad de vida existente, resulta ser una **ciencia inexacta** que tiene problemas para clasificar organismos que no coinciden con ninguno de las categorías ya determinadas o que comparten muchas características con más de una. Tal es el caso del panda gigante, que comparte características con los pandas rojos (una especie que también ha tenido problemas para clasificarse por su semejanza con los mapaches al punto de haber estado en la misma familia, aunque actualmente tiene la suya, *Ailuridae*) como su dieta de bambú y su *"falso pulgar"*, y con la familia de los osos, *Ursidae*, en especial con los osos polares. En la actualidad se considera parte de esta última familia.

Árboles filogenéticos

Un **árbol filogenético**, o filogenia, es un diagrama en forma de árbol que muestra, en términos evolutivos, la relación entre diversos organismos basado en sus semejanzas y diferencias físicas o genéticas. Las hojas, llamadas unidades taxonómicas, representan especies y las líneas describen el desarrollo evolutivo inferido con respecto a las demás especies del diagrama para converger en lo que se conoce como un ancestro común, aunque también los hay sin raíz, que sólo describen semejanza entre un grupo de organismo. Estos árboles son un complemento muy útil para la visualización de la clasificación y diversidad biológica.

Existen varios métodos para para poder generar un árbol filogenético pero en general podemos verlos en dos grandes grupos: Aquellos que lo generan a través de la información de distancias entre las secuencias, como UPGMA y Neighbor-Joining, y aquellos que lo hacen a través de carácteres discretos (usan la propia secuencia durante la inferencia) como Maximum likelihood y los métodos bayesianos. Nos centraremos en comparar un algoritmo de cada grupo entre sí.

Neighbor-Joining

Es un algoritmo de clustering aglomerativo (*bottom-up*) que supone tener la distancia entre cualquier par de secuencias para hacer el árbol, es por ello que espera tener secuencias previamente alineadas, pues de esta manera puede usar, por ejemplo, su distancia como la fracción de posiciones que no coinciden (ignorando *gaps*). El algoritmo es el siguiente:

1. Iniciar con un árbol en forma de estrella donde cada punta son las secuencias.
2. Repetir hasta que el árbol esté completado (ya no haya más pares para unir):
   2.1. Calcular una matriz especial de distancias Q basada en las distancias de las secuencias recibidas.
   2.2. Encontrar el par (f, g) distintos de la matriz con la distancia más corta y unirlas a través de un nuevo nodo u.
   2.3. Conectar el nuevo nodo al nodo central.
   2.4. Calcular las distancias de f y g a u y actualizar a Q.
   2.5. Calcular las distancias de los demás nodos a u y actualizar a Q.

Se puede apreciar que es un algoritmo *greedy* pues solo se fija en su "estado actual" para actualizar la matriz. Con respecto a las especificaciones de Q y de cómo actualizar las distancias, se utilizan fórmulas que han probado funcionar apropiadamente, pero en general su elección no afecta el funcionamiento general del algoritmo, aunque claro, sí su eficiencia.

La definición de la matriz Q es la siguiente, $d(i, j)$ representa la distancia entre i y j:

$$Q(i,j) = (n-2)d(i,j) - \sum_{k=1}^{n} d(i,k) - \sum_{k=1}^{n} d(j,k)$$

Y el cálculo de las nuevas distancias de f y g a u (que se denotan δ) son

$$\delta(f,u) = \frac{1}{2}d(f,g) + \frac{1}{2(n-2)}\left[\sum_{k=1}^{n} d(f,k) - \sum_{k=1}^{n} d(g,k)\right]$$

y

$$\delta(g,u) = d(f,g) - \delta(f,u)$$

Por último, las distancias de los demás nodos a u es:

$$d(u,k) = \frac{1}{2}[d(f,k) + d(g,k) - d(f,g)]$$

La mayor ventaja de Neighbor-Joining es que es rápido en comparación a los demás métodos pero, a diferencia de ellos, tienden a dar resultados más imprecisos, es por ello que es muy comúnmente usado para analizar conjuntos de datos de muy grandes dimensiones.

Maximum likelihood

Es un método para inferir el mejor árbol filogenético a partir de las probabilidades de que un modelo dado haya generado las secuencias deseadas, con lo que genera nuevos modelos y busca aquellos con la mayor probabilidad (o likelihood) de llegar al estado deseado.

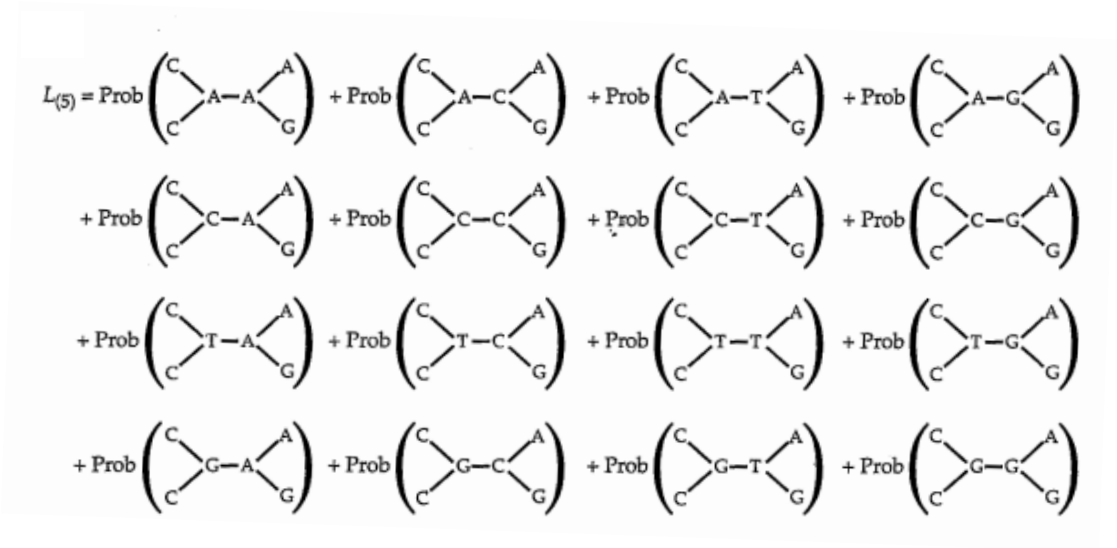
La idea el algoritmo es la siguiente:

1. Tomamos valores iniciales para los parámetros de nuestro modelo, que incluyen pero no se limitan a: La topología del árbol, la longitud de las ramas, frecuencia de los nucleótidos, etc.
2. Obtenemos del modelo la matriz de probabilidades de que cualquier par de bases haya mutado (converga en el tiempo en el árbol).
3. Con estas calculamos el likelihood del modelo como multiplicación de los likelihoods individuales de cada columna en el alineamiento, aunque usualmente se prefiere calcular la suma del logaritmo de estos valores para evitar valores tan pequeños.
    3.1. Los likelihoods individuales se calculan como la suma de las probabilidades de las “historias” (que son los nucleótidos raíz que desconocemos para cada)
4. Al final alteramos ligeramente los valores de los parámetros y calculamos el likelihood hasta maximizarlo tanto como sea posible.

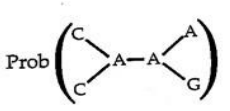
Si tenemos entonce nuestras secuencias alineada algo de esta forma:

	1	2	3	4	5	6	7	8	9	$\ldots n$
OTU1	A	A	G	A	C	T	T	C	A	...N
OTU2	A	G	C	C	C	T	T	C	T	...N
OTU3	A	G	A	T	A	T	C	C	A	...N
OTU4	A	G	A	G	G	T	C	C	T	...N

Calculamos la suma de logaritmos de cada L_i con i de 1 hasta n. En donde, por ejemplo L_5, cuya columna tiene C, C, A y G, quedaría:



Donde cada probabilidad se obtiene multiplicando la probabilidad de los pares unidos por aristas según su correspondiente probabilidad en la tabla generada de los parámetros. También multiplicamos la probabilidad de frecuencia (de que aparezca) de la raíz, que en modelos reversibles es un nodo arbitrario. Si por ejemplo tomamos como raíz A en el siguiente ejemplo tenemos:



$$= P_{Aeq} * P_{AC} * P_{AC} * P_{AA} * P_{AA} * P_{AG}$$

Maximum Likelihood es una heurística que depende de la elección de los parámetros iniciales para encontrar un valor óptimo y de ejecutar varias iteraciones en distintos valores iniciales para no atascarse en *máximos locales* (como pasa con ascenso de colina).

Por todo esto es un algoritmo muy eficiente y el preferible cuando se busca exactitud pero que depende del modelo usado para generar las probabilidades, de los valores iniciales y de la cantidad de iteraciones. Lo que también hace que sea muy caro computacionalmente e ineficiente en grandes cantidades de datos.

Pregunta de Investigación

Entre **Neighbor-Joining** y **Maximum Likelihood** ¿Qué modelo para el diseño de árboles filogenéticos es mejor para lidiar con las secuencias del gen de los pandas gigantes, pandas rojos y algunos osos?

Hipótesis

Dada la pequeña cantidad de secuencias y su reducido tamaño (no es el genoma completo), esperamos que Maximum Likelihood tenga mejor desempeño.

Objetivos

Objetivo Principal

Generar el árbol filogenético de panda gigante, panda rojo y algunas especies de osos.
Usando los algoritmos de Neighbor-Joining y Maximum Likelihood y comparar sus resultados.

Objetivos secundarios

- Tratar de deducir las semejanzas evolutivas del panda con las demás especies antes mencionadas.
- Obtención y alineamiento de secuencias. Generación del árbol.

Métodos

Obtención de datos

Decidimos comparar el gen cytochrome b de las siguientes especies:

- Panda gigante (Ailuropoda melanoleucapanda)
- Panda rojo (Ailurus fulgens)
- Oso pardo (Ursus arctos)
- Oso tibetano (Ursus thibetanus thibetanus)
- Oso polar (Ursus maritimus)

Cytochrome b es una proteína que se encuentra en la mitocondria de células eucariotas, y forma parte de la cadena de transporte de electrones.

Secuencias utilizadas

A través de la base de datos del NCBI obtuvimos las secuencias de ADN para el gen del citocromo b de cada una de las especies mencionadas. Estas están adjuntadas en el archivo osos-adn-seqs.fasta.

Además, obtuvimos (también del NCBI) las secuencias de aminoácidos de la proteína citocromo b de cada una de las especies de las que hablamos. Estas se encuentran en el archivo osos.seqs.fasta.

Los enlaces a las secuencias en NCBI son las siguientes:

- Panda gigante
- Panda rojo
- Oso pardo
- Oso tibetano
- Oso polar

Alineamiento

Utilizamos el programa MEGA para realizar los alineamientos, por lo que usamos el algoritmo MUSCLE.

Generación de árboles filogenéticos

Una vez alineadas las secuencias, utilizamos la opción de MEGA para generar árboles filogenéticos a partir de estas.

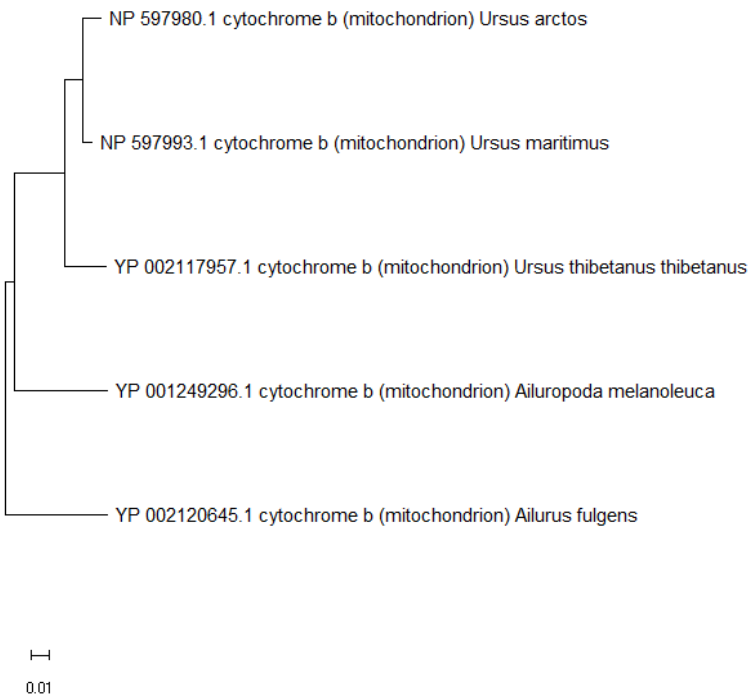
Todos los árboles de aminoácidos se generaron con el modelo de Poisson y los de AND con el modelo evolutivo de Jukes-Cantor, que supone que la sustitución de una base por cualquier otra tiene la misma probabilidad.

Resultados

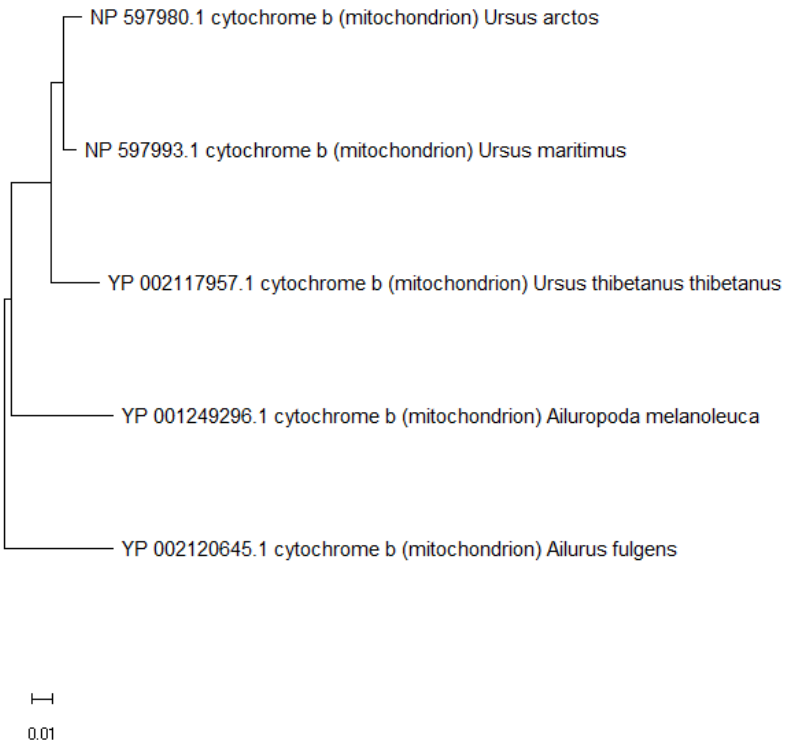
Anexamos los árboles filogenéticos generados con los diversos métodos tanto de las secuencias de aminoácidos como de las de nucleótidos.

Con secuencias de aminoácidos

Árbol filogenético generado con Maximum Likelihood Tree:



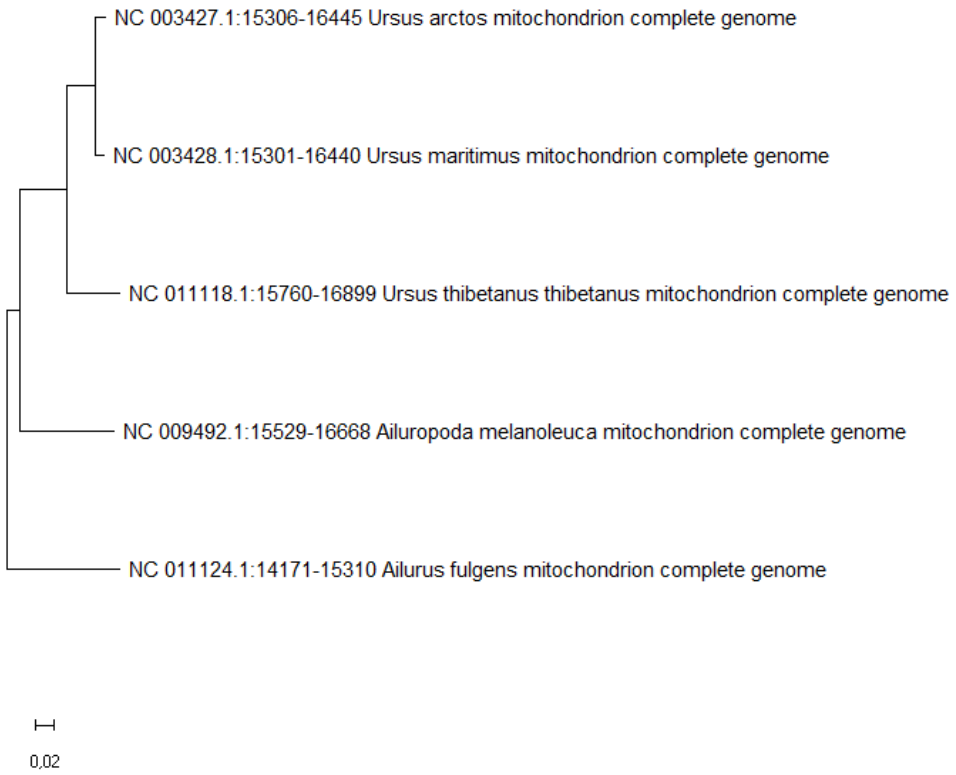
Árbol filogenético generado con Neighbor-joining-tree:



Con secuencias de nucleótidos

Árboles generados a partir de secuencias de nucleótidos alineados:

- Usando Maximum Likelihood Tree:



- Usando Neighbor-joining-tree:

Conclusiones

En general notamos que todos los árboles generados presentan la misma estructura independiente del algoritmo utilizado para generarlos y de si son aminoácidos o nucleótidos. Era algo que podíamos esperar debido a la pequeña magnitud de secuencias y de tamaños que utilizamos. Todos los modelos cumplen con asociar a los pandas gigantes con la familia de los osos antes que con el panda rojo.

Si bien los resultados no lo reflejan, mencionamos que ambos algoritmos se utilizan en diferentes situaciones: Cuando hay muchos datos es preferible Neighbor-Joining y cuando se busca precisión es más común usar Maximum Likelihood.

Bibliografía

1. Archer J, Robertson DL. CTree: comparison of clusters between phylogenetic trees made easy. Bioinformatics. 2007
2. Maddison DRaWPM. MacClade version 4: Analysis of phylogeny and character evolution. Sinauer Associates, Sunderland Massachusetts; 2000
3. arr CS, Lee B, Campbell D, Bederson BB. Visualizations for taxonomic and phylogenetic trees. Bioinformatics. 2004
    4. Saitou NM. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol Biol Evol. 1987
5. Schmidt HA, Strimmer K, Vingron M, Haeseler aAv. TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. Bioinformatics. 2002
6. Paradis E, Claude J, Strimmer K. APE: Analyses of Phylogenetics and Evolution in R language. Bioinformatics. 2004
7. https://academic.oup.com/mbe/article/18/4/465/979709