

TREECLUSTER: CLUSTERING BIOLOGICAL SEQUENCES USING PHYLOGENETIC TREES

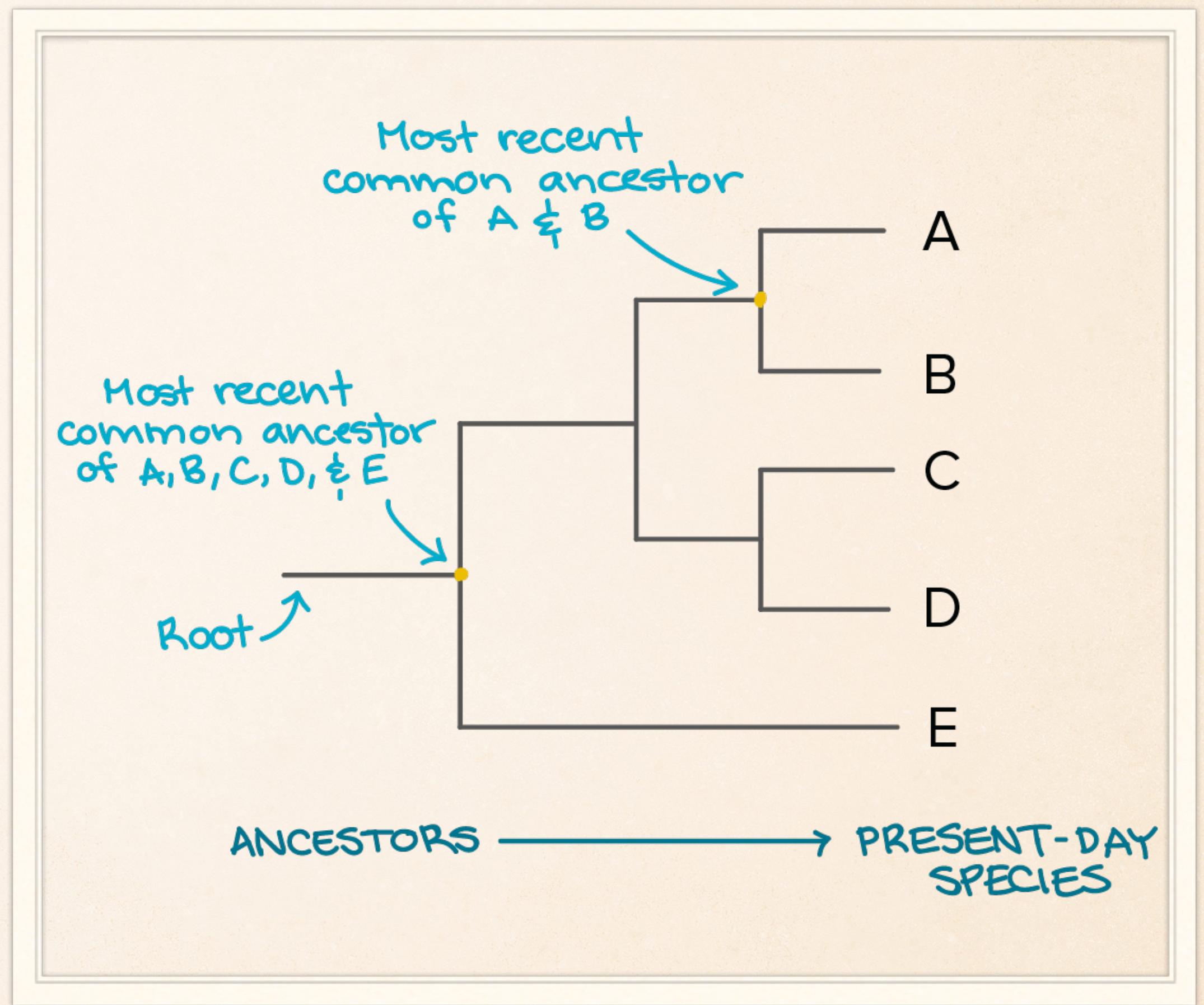
STEFANIE MUROYA LEI

“Herramienta de agrupamiento de secuencias basada en árboles filogenéticos”

MARCO TEÓRICO

ÁRBOLES FILOGENÉTICOS

- ❖ Se usa para representar las relaciones evolutivas entre organismos.
- ❖ Refleja cómo un grupo de organismos ha evolucionado a partir de ancestros comunes.
- ❖ Se construyen a partir de información como características físicas o DNA.
- ❖ Son tan buenos como los datos con los que se construyeron.
- ❖ Son hipótesis.
- ❖ Es ultramétrico si las distancias de todas las hojas al nodo son iguales.



UNIDAD TAXONÓMICA OPERATIVA (OTU)

- ❖ Unidad de clasificación que se determina por el investigador para individualizar a objetos de estudio para poder construir un árbol filogenético.
- ❖ *E.g. usar un umbral.*

INTRODUCCIÓN

¿POR QUÉ ÁRBOLES FILOGENÉTICOS?

- ❖ Reflejan distancias evolutivas, relaciones y divergencia entre organismos.
- ❖ Se pueden estimar en una complejidad sub-cuadrática y elimina la tarea de calcular todas distancias por pares.
 - ❖ Mejora la escalabilidad y velocidad
 - ❖ Usualmente ya se encuentran listos para usar puesto que tienen más área de aplicación.

ENFOQUE DEL TRABAJO

- ❖ Si el árbol es ultramétrico entonces el problema es trivial
- ❖ Rara vez nos topamos con árboles ultramétricos.
- ❖ Por tanto, puede ser tratado como un problema de optimización:

“Encontrar el mínimo número de clústers tal que un criterio se cumpla”

Familia de problemas de particionamiento de árboles con soluciones que
tienen complejidad de tiempo lineal.

DEFINICIONES DEL PROBLEMA

EL ÁRBOL FILOGENÉTICO

$T(E, V)$: árbol binario acíclico y no dirigido con un conjunto E y V de aristas y vértices respectivamente.

AGRUPAMIENTO VÁLIDO

- ❖ Remover un subconjunto de aristas tal que:
 - ❖ Tengamos componentes conexos.
 - ❖ Todas las hojas se pueden asignar a algún componente conexo

PROBLEMA: PARTICIONAMIENTO MIN-CUT

Se busca que la máxima carnalidad sea la menor posible.

$$\forall L_i \in \{L_1, \dots, L_n\}, f_T \leq \alpha$$

- ❖ T: árbol binario, acíclico y no dirigido.
- ❖ V: vértices del árbol.
- ❖ L_i : partición de un conjunto de hojas
 $L \subseteq V$
- ❖ $f_T : L \rightarrow \mathbb{R}$
- ❖ α : número real

PROBLEMA: MIN-CUT CON MÁXIMO DIÁMETRO

$$f_t = \max_{u,v \in L_i} (d(u, v))$$

Nota: no siempre resulta ser una representación precisa.

- ❖ L : conjunto de hojas, $L \subseteq V$.
- ❖ V : vértices del árbol.
- ❖ L_i : partición de un conjunto de hojas
 $L \subseteq V$
- ❖ $f_T : L \rightarrow \mathbb{R}$
- ❖ $d(u, v)$: distancia entre los vértices u y v.

PROBLEMA: MIN-CUT CON MÁXIMA SUMA

$$f_t = \sum_{(u,v) \in E(T|L_i)} w(u, v)$$

- ❖ T: árbol binario, acíclico y no dirigido.
- ❖ V: vértices del árbol.
- ❖ L : conjunto de hojas, $L \subseteq V$.
- ❖ L_i : partición de un conjunto de hojas $L \subseteq V$
- ❖ $f_T : L \rightarrow \mathbb{R}$
- ❖ $w(u, v)$: peso asignado a la arista que conecta u con v.
- ❖ $E(T|L_i)$: conjunto de aristas que se dan en un árbol restringido por un conjunto de aristas L_i .

PROBLEMA: MIN-CUT CON ENLACE ÚNICO

Se busca maximizar las distancias entre clústers

$$f_t = \max_{S \subset L_i} \{ \min_{u \in S, v \in (L_i - S)} d(u, v) \}$$

- ❖ L_i : partición de un conjunto de hojas
 $L \subseteq V$
- ❖ $f_T : L \rightarrow \mathbb{R}$
- ❖ $d(u, v)$: distancia entre los vértices u y v.

ALGORITMOS

Algorithm 1: Linear-time solution for Max-diameter min-cut partitioning

Input: A tree $T^o = (V, E)$ and a threshold α

```
1  $B(v) \leftarrow 0$  for  $v \in V$ 
2 for  $u \in$  post order traversal of internal nodes of  $T^o$  do
3   if  $B(u_l) + w_l + B(u_r) + w_r > \alpha$  then
4     if  $B(u_l) + w_l \leq B(u_r) + w_r$  then
5        $E \leftarrow E - \{ (u, u_r) \}$ 
6        $B(u) \leftarrow B(u_l) + w_l$ 
7     else
8        $E \leftarrow E - \{ (u, u_l) \}$ 
9        $B(u) \leftarrow B(u_r) + w_r$ 
10   else
11      $B(u) \leftarrow \max(B(u_l) + w_l, B(u_r) + w_r)$ 
12 return Leafsets of every connected component in  $T^o$ 
```

♦ T^o : árbol enraizado en un nodo arbitrario o.

♦ $B(u)$:Retorna la máxima distancia de un nodo u a cualquier otro nodo en el clúster donde se encuentra.

♦ w_i : peso de la arista que conecta nuestro actual nodo raíz con el nodo i.

MIN-CUT CON MÁXIMA SUMA

- ❖ Inspirado en Algoritmo I.
- ❖ $B(u)$ ahora es la suma de los pesos de todos los descendientes de u .
- ❖ Rompemos la arista que mayor peso provoca si se pasa un umbral.

PROBLEMA: MIN-CUT CON ENLACE ÚNICO

- ❖ Cada par de nodos es puesto en un mismo cluster si (pero no solo si) su distancia no excede un umbral.
- 1. En post-orden haya la hoja más cercana de los sub-árboles derechos e izquierdos de un árbol enraizado en u.
- 2. En pre-orden haya el nodo más cercano fuera del sub-árbol enraizado en u.
- 3. Se corta una arista cuando las distancias entre los nodos más cercanos de los sub-arboles derechos e izquierdos exceden un umbral y se excede un umbral de la distancia de cualquiera de estos a el nodo más cercano fuera del sub-arbol de u.

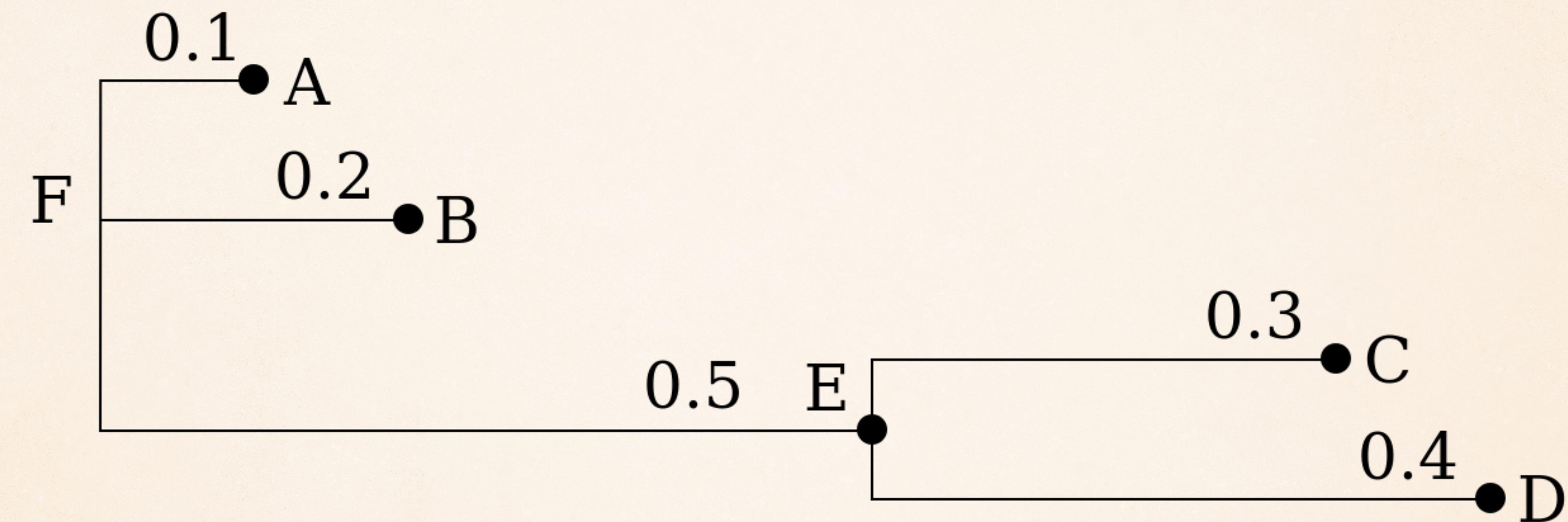
```
if minBelow[ul] + wl + minBelow[ur] + wr > α and  
minBelow[ul] + wl + minAbove[u] > α then  
E ← E \ (u, ul)  
if minBelow[ul] + wl + minBelow[ur] + wr > α and  
minBelow[ur] + wr + minAbove[u] > α then  
E ← E \ (u, ur)  
if minBelow[ul] + wl + minAbove[u] > α and  
minBelow[ur] + wr + minAbove[u] > α then
```

NOTAS ADICIONALES

- ❖ Los algoritmos pueden ser modificados para que cada sub-árbol contenga a todos sus descendientes (monophyletic clade).
- ❖ Para representar al centroide:
 - ❖ Hoja que esta más cercana al punto medio.
 - ❖ Por consenso: escoger por cada posición la letra más común en las secuencias.
 - ❖ Primero encontramos el punto de balance, se efectúa *maximum likelihood ancestral state reconstruction (ASR)* y se utiliza esto como centroide.

APLICACIÓN DE TREE CLUSTER

NEWICK TREE



(:o.1,:o.2,(:o.3,:o.4):o.5):o.o;

CENTROIDE DE UN CLÚSTER

- ❖ Varianza: Encontrar el nodo que minimice la varianza de las hojas.
- ❖ Consenso: Elegir la letra que más se repite por posición.
- ❖ Ancestral State Reconstruction (ASR): Reconstrucción del ancestro, y usar este ancestro como representante.

GREENGENES V13.5

203 452 SECUENCIAS

PRUEBA CON MAX-DIAMETER

❖ Respecto a U-CLUST:

- ❖ U-CLUST tiene mas singletons
- ❖ El cluster más grande de U-CLUST es 3 veces más grande que el de TreeCluster.
- ❖ U-CLUST tiene a irse por extremos en cuanto tamaño de clústers.

threshold	singletons	total #clusters	max. cluster size
0	0.015	86387.0	123456.0
1	0.030	42510.0	77263.0
2	0.045	24795.0	54068.0
3	0.060	15257.0	39809.0
4	0.090	6396.0	23631.0
5	0.120	3003.0	15052.0
6	0.150	1525.0	10112.0

Resultados de TreeCluster

DIVERSIDAD DEL CLÚSTER

- ❖ L_i : hojas que forman un clúster
- ❖ $d(i, j)$: función de distancia
- ❖ Se mide utilizando el algoritmo I.

$$\mu(\{L_1, \dots, L_N\}) = \frac{\sum_{k=1}^N |L_k| \frac{\sum_{i,j \in L_k} d(i,j)}{|L_k|^2}}{\sum_{k=1}^N |L_k|} = \frac{1}{n} \sum_{k=1}^N \sum_{i,j \in L_k} \frac{d(i,j)}{|L_k|}$$

EXTENSIÓN FASTA

>III0860

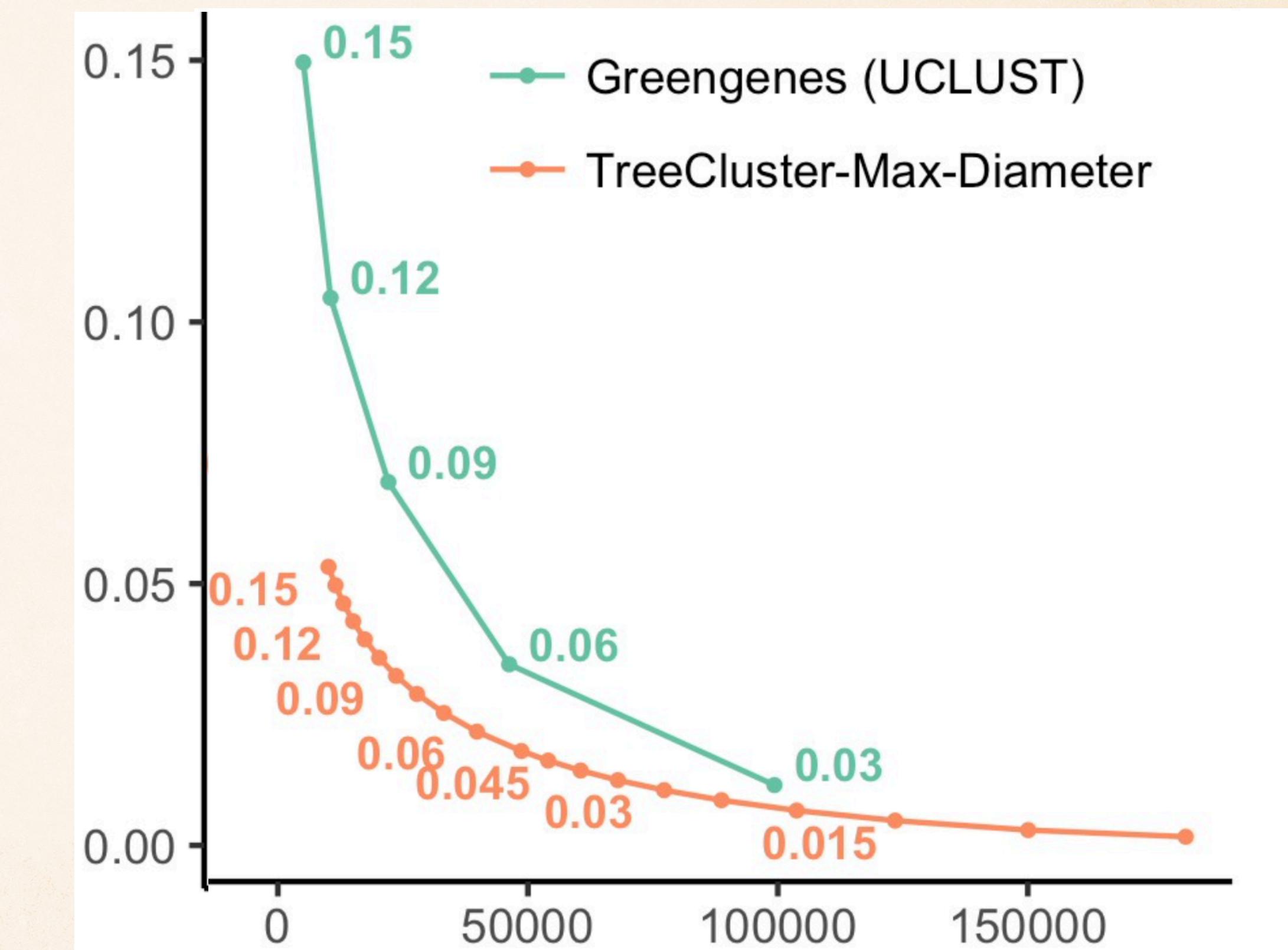
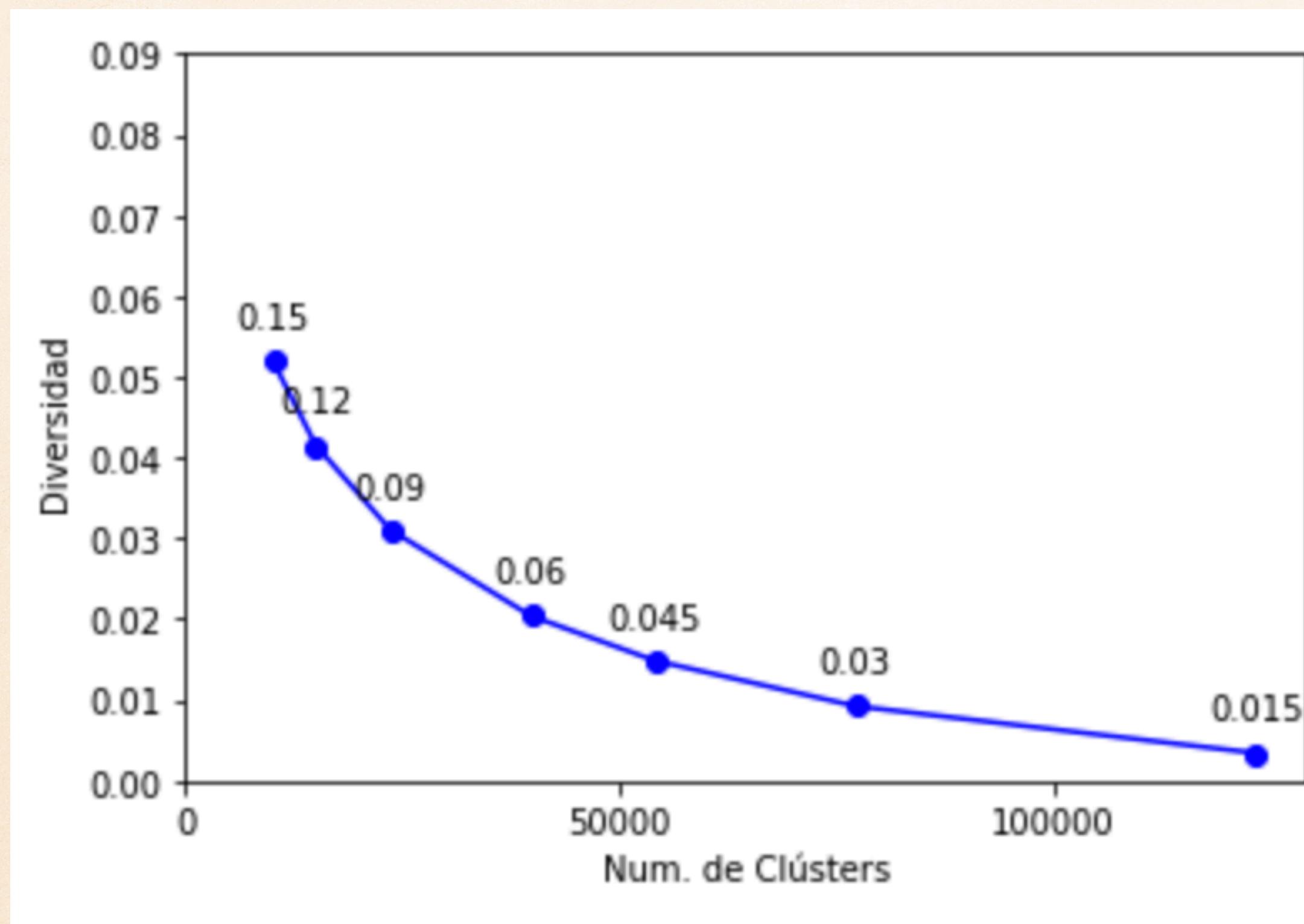
A-A—AG--U--

>III0861

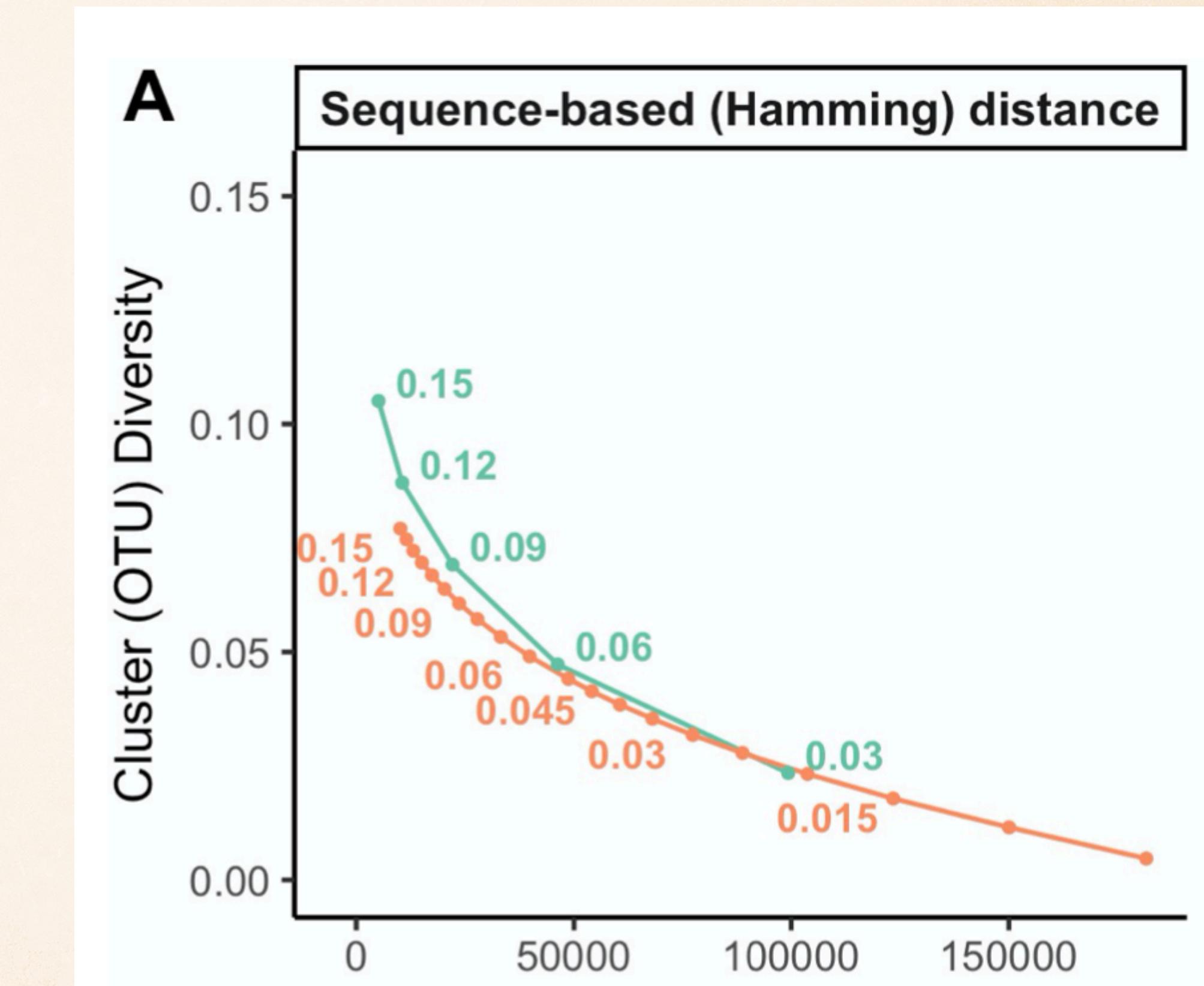
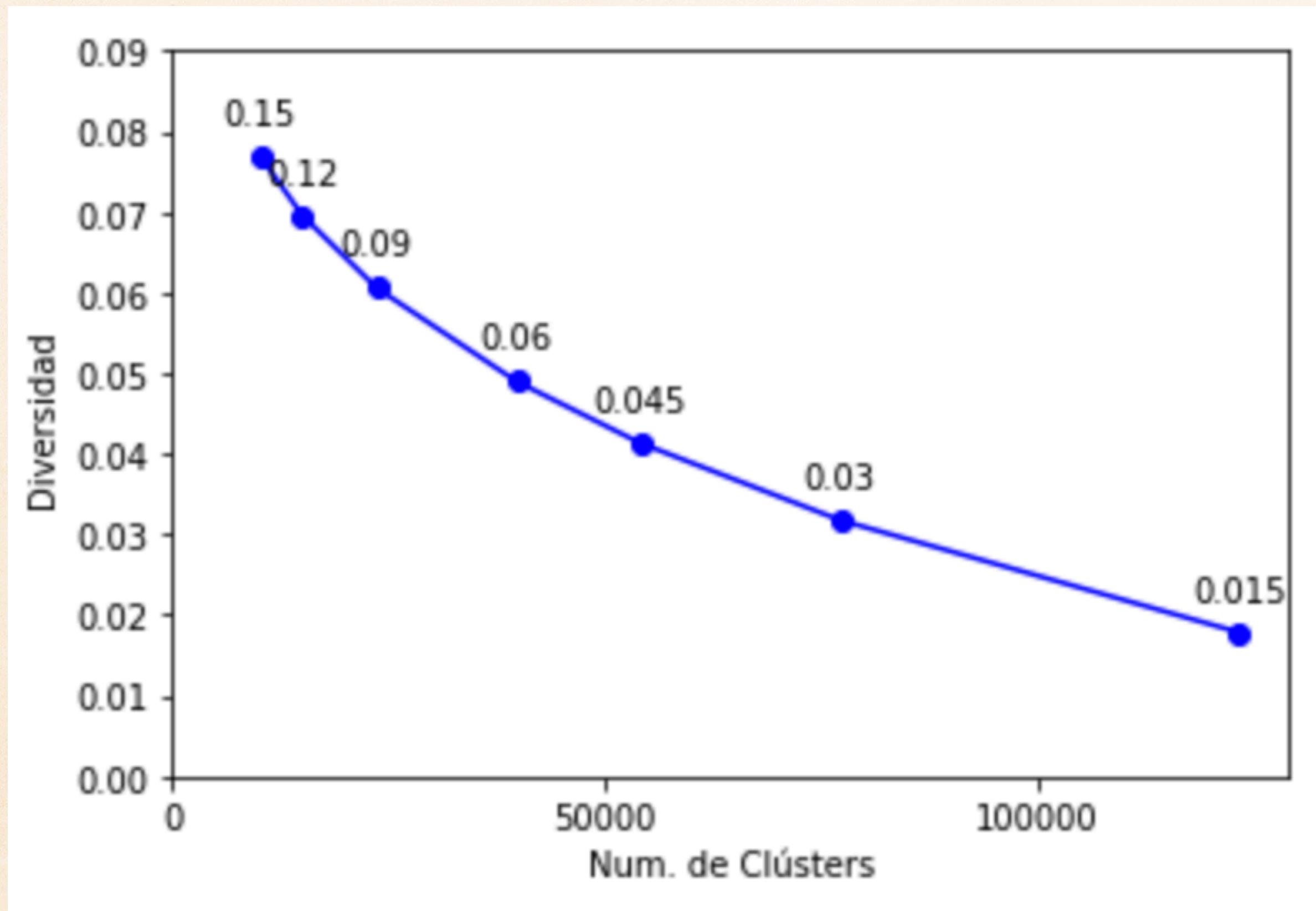
A—A—AG—U--

....

POR LONGITUD DE CAMINO



POR DISTANCIA DE HAMMING



Longitud de secuencias: 7683

BIBLIOGRAFÍA

- ❖ <https://www.khanacademy.org/science/high-school-biology/hs-evolution/hs-phylogeny/a/phylogenetic-trees>
- ❖ <https://niemasd.github.io/TreeSwift/>
- ❖ <https://evolution.genetics.washington.edu/phylip/newicktree.html>
- ❖ <https://zhanglab.ccmb.med.umich.edu/FASTA/>
- ❖ <https://github.com/niemasd/NiemaDS>
- ❖ <https://doi.org/10.1371/journal.pone.0221068>