

INFERENCIA FILOGENÉTICA

1. DISTANCIA

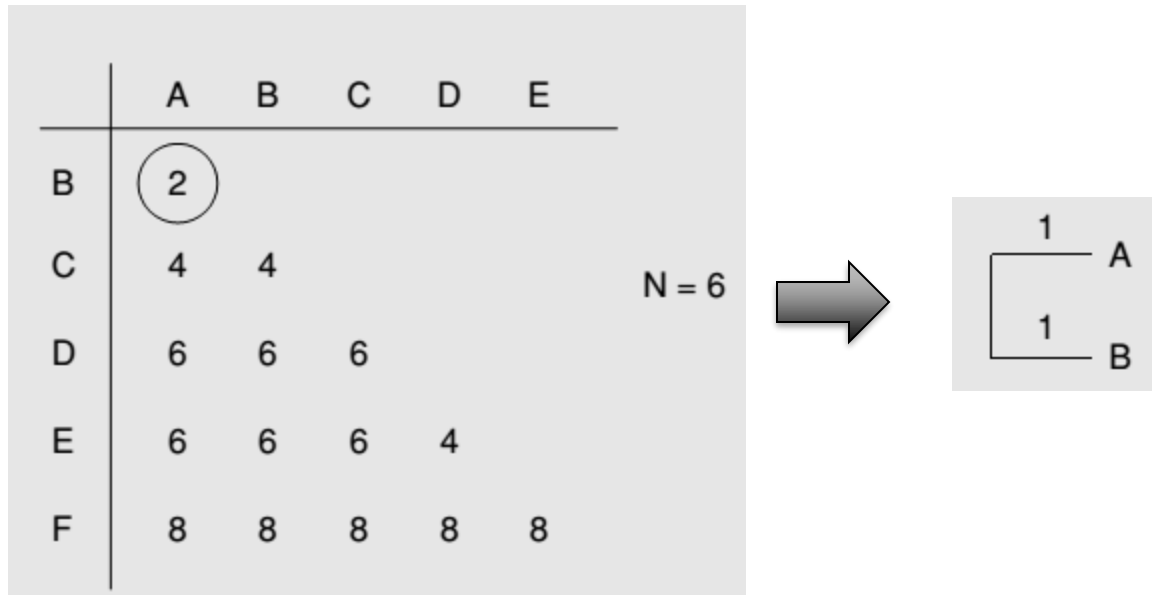
2. ARGUMENTACIÓN HENNIGIANA

MÉTODOS DE DISTANCIA

Forma simple de medir **divergencia** entre dos terminales y construir árboles filogenéticos

Método 1: UPGMA (unweighted-pair group method with arithmetic means)

1. Medir distancias pareadas (p-distance) y agrupar a aquellos que tienen menor distancia



MÉTODOS DE DISTANCIA

Método 1: UPGMA (unweighted-pair group method with arithmetic means)

2. Medir distancias pareadas (p-distance) desde el cluster a los otros terminales y repetir primer paso hasta terminar

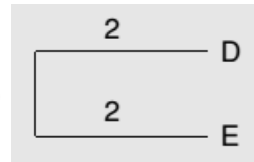
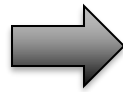
$$d_{(AB)C} = (d_{AC} + d_{BC})/2 = 4$$

$$d_{(AB)D} = (d_{AD} + d_{BD})/2 = 6$$

$$d_{(AB)E} = (d_{AE} + d_{BE})/2 = 6$$

$$d_{(AB)F} = (d_{AF} + d_{BF})/2 = 8$$

	(AB)	C	D	E
C	4			
D	6	6		
E	6	6	4	
F	8	8	8	8



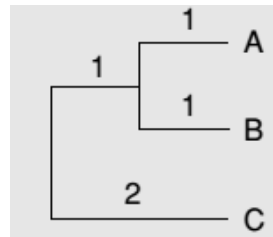
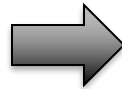
$$d_{(DE)(AB)} = (d_{D(AB)} + d_{E(AB)})/2 = 6$$

$$d_{(DE)C} = (d_{DC} + d_{EC})/2 = 6$$

$$d_{(DE)F} = (d_{DF} + d_{EF})/2 = 8$$

	(AB)	C	(DE)
C	4		
(DE)	6	6	
F	8	8	8

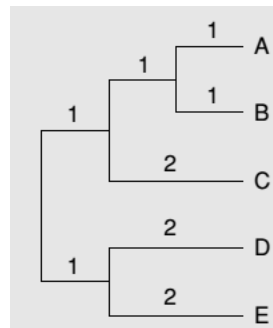
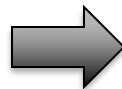
$N = N - 1 = 4$



$$d_{(ABC)(DE)} = (d_{AB(DE)} + d_{C(DE)})/2 = 6$$

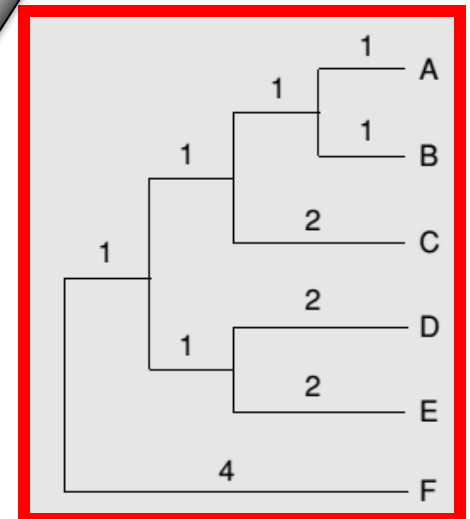
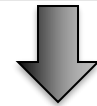
$$d_{(ABC)F} = (d_{ABF} + d_{CF})/2 = 8$$

	(ABC)	(DE)
(DE)	6	
F	8	8



$N = N - 1 = 3$

	(ABC), (DE)
F	8



MÉTODOS DE DISTANCIA

Método 1: UPGMA (unweighted-pair group method with arithmetic means)

PROBLEMA: no estima correctamente la distancia genética entre dos terminales

SOLUCIÓN: convertir distancias pareadas a distancias evolutivas usando MODELOS DE EVOLUCIÓN

P.e. corrección con **JC69**

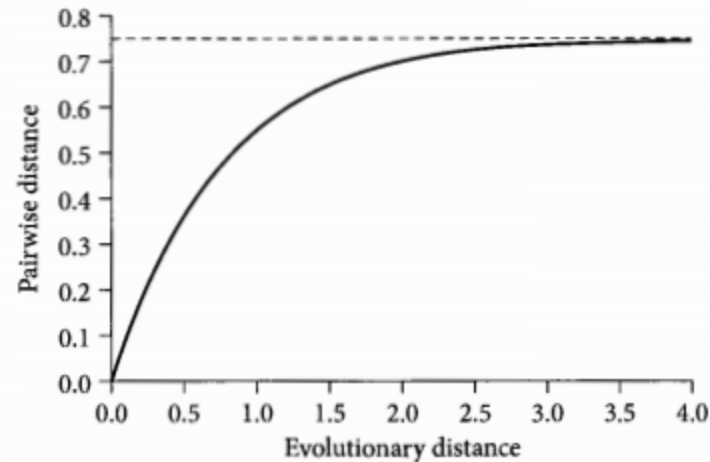
$$P_{ii}(t) = 1/4 + 3/4 \exp(-\mu t)$$

$$P_{ij}(t) = 1/4 - 1/4 \exp(-\mu t)$$

$$\mu t = -1/2 \log(1 - 4/3 p)$$

$$p = 3/4[1 - \exp(-2\mu t)]$$

$$d = -3/4 \ln(1 - 4/3 p)$$



MÉTODOS DE DISTANCIA

Método 2: Neighbor-Joining: minimiza la longitud total de las ramas

1. Matriz de distancias evolutivas

	a	b	c	d	e
a	0	5	9	9	8
b	5	0	10	10	9
c	9	10	0	8	7
d	9	10	8	0	3
e	8	9	7	3	0

2. Primer ensamblaje de vecinos: convertir valores a matriz Q_1 y escoger el par con valores más bajos

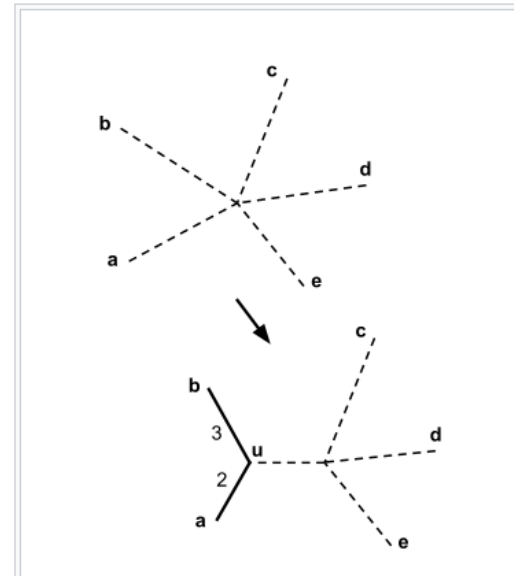
	a	b	c	d	e
a		-50	-38	-34	-34
b	-50		-38	-34	-34
c	-38	-38		-40	-40
d	-34	-34	-40		-48
e	-34	-34	-40	-48	

$$Q_1(a, b) = (n - 2)d(a, b) - \sum_{k=1}^5 d(a, k) - \sum_{k=1}^5 d(b, k)$$

3. Primera estimación de long. de ramas

$$\delta(a, u) = \frac{1}{2}d(a, b) + \frac{1}{2(5-2)} \left[\sum_{k=1}^5 d(a, k) - \sum_{k=1}^5 d(b, k) \right] = \frac{5}{2} + \frac{31-34}{6} = 2$$

$$\delta(b, u) = d(a, b) - \delta(a, u) = 5 - 2 = 3$$



MÉTODOS DE DISTANCIA

Método 2: Neighbor-Joining: minimiza la longitud total de las ramas

4. Actualizar matriz de distancia con el nuevo par

$$d(u, c) = \frac{1}{2}[d(a, c) + d(b, c) - d(a, b)] = \frac{9 + 10 - 5}{2} = 7$$

$$d(u, d) = \frac{1}{2}[d(a, d) + d(b, d) - d(a, b)] = \frac{9 + 10 - 5}{2} = 7$$

$$d(u, e) = \frac{1}{2}[d(a, e) + d(b, e) - d(a, b)] = \frac{8 + 9 - 5}{2} = 6$$

	u	c	d	e
u	0	7	7	6
c	7	0	8	7
d	7	8	0	3
e	6	7	3	0

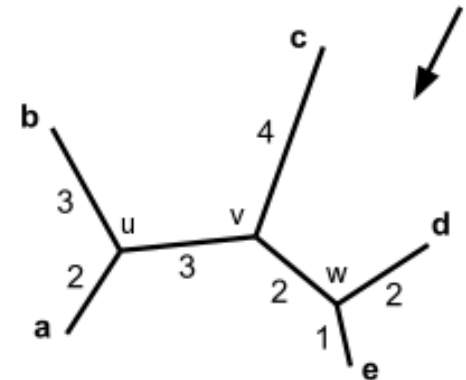
5. Segundo ensamblaje de vecinos: convertir valores a matriz Q_2 y escoger el par con valores más bajos

	u	c	d	e
u		-28	-24	-24
c	-28		-24	-24
d	-24	-24		-28
e	-24	-24	-28	

6. Segunda estimación de long. de ramas

$$\delta(u, v) = \frac{1}{2}d(u, c) + \frac{1}{2(4-2)} \left[\sum_{k=1}^4 d(u, k) - \sum_{k=1}^4 d(c, k) \right] = \frac{7}{2} + \frac{20 - 22}{4} = 3$$

$$\delta(v, c) = d(u, c) - \delta(u, v) = 7 - 3 = 4$$



MÉTODOS DE DISTANCIA

Método 2: Neighbor-Joining: minimiza la longitud total de las ramas

7. Actualización final de matriz de distancia con el nuevo par

$$d(v, d) = \frac{1}{2}[d(u, d) + d(c, d) - d(u, c)] = \frac{7 + 8 - 7}{2} = 4$$

$$d(v, e) = \frac{1}{2}[d(u, e) + d(c, e) - d(u, c)] = \frac{6 + 7 - 7}{2} = 3$$

	v	d	e
v	0	4	3
d	4	0	3
e	3	3	0

8. Tercer ensamblaje de vecinos: convertir valores a matriz Q_3 y escoger el par con valores más bajos

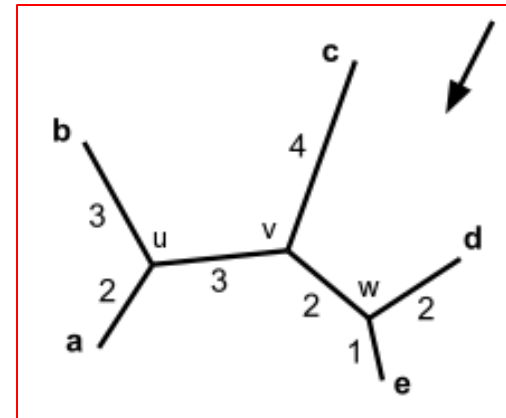
	v	d	e
v		-10	-10
d	-10		-10
e	-10	-10	

9. Última estimación de long. de ramas

$$\delta(v, w) = \frac{1}{2}d(v, d) + \frac{1}{2(3-2)} \left[\sum_{k=1}^3 d(v, k) - \sum_{k=1}^3 d(d, k) \right] = \frac{4}{2} + \frac{7-7}{2} = 2$$

$$\delta(w, d) = d(v, d) - \delta(v, w) = 4 - 2 = 2$$

$$\delta(w, e) = d(v, e) - \delta(v, w) = 3 - 2 = 1$$



MÉTODOS DE DISTANCIA

Método 2: Neighbor-Joining: minimiza la longitud total de las ramas

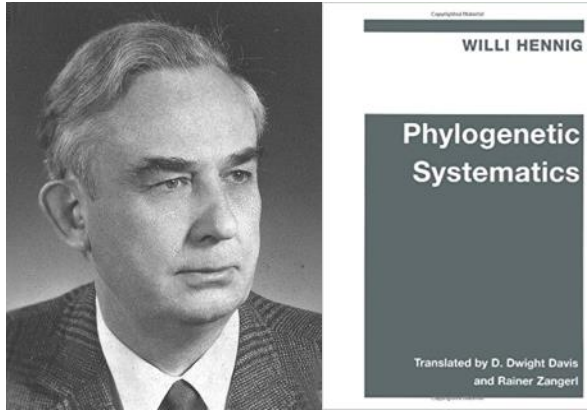
VENTAJAS:

- Muy rápido para matrices muy grandes
- Usa distancias evolutivas y estima longitud de ramas
- Buen estimativo de árbol inicial para otros análisis

DESVENTAJAS:

- No se basa en evolución de caracteres individuales
- No permite evaluar la calidad del árbol

SISTEMÁTICA FILOGENÉTICA



Objetivo: reconstruir las relaciones de parentesco entre taxones y proveer una clasificación concordante usando clados monofiléticos

En la naturaleza hay un orden jerárquico

Los caracteres permiten reconstruir ese orden

Especies y taxones superiores pueden definirse como monofiléticos si y solo si comparten una novedad evolutiva única (sinapomorfía)

Argumentación Hennigiana

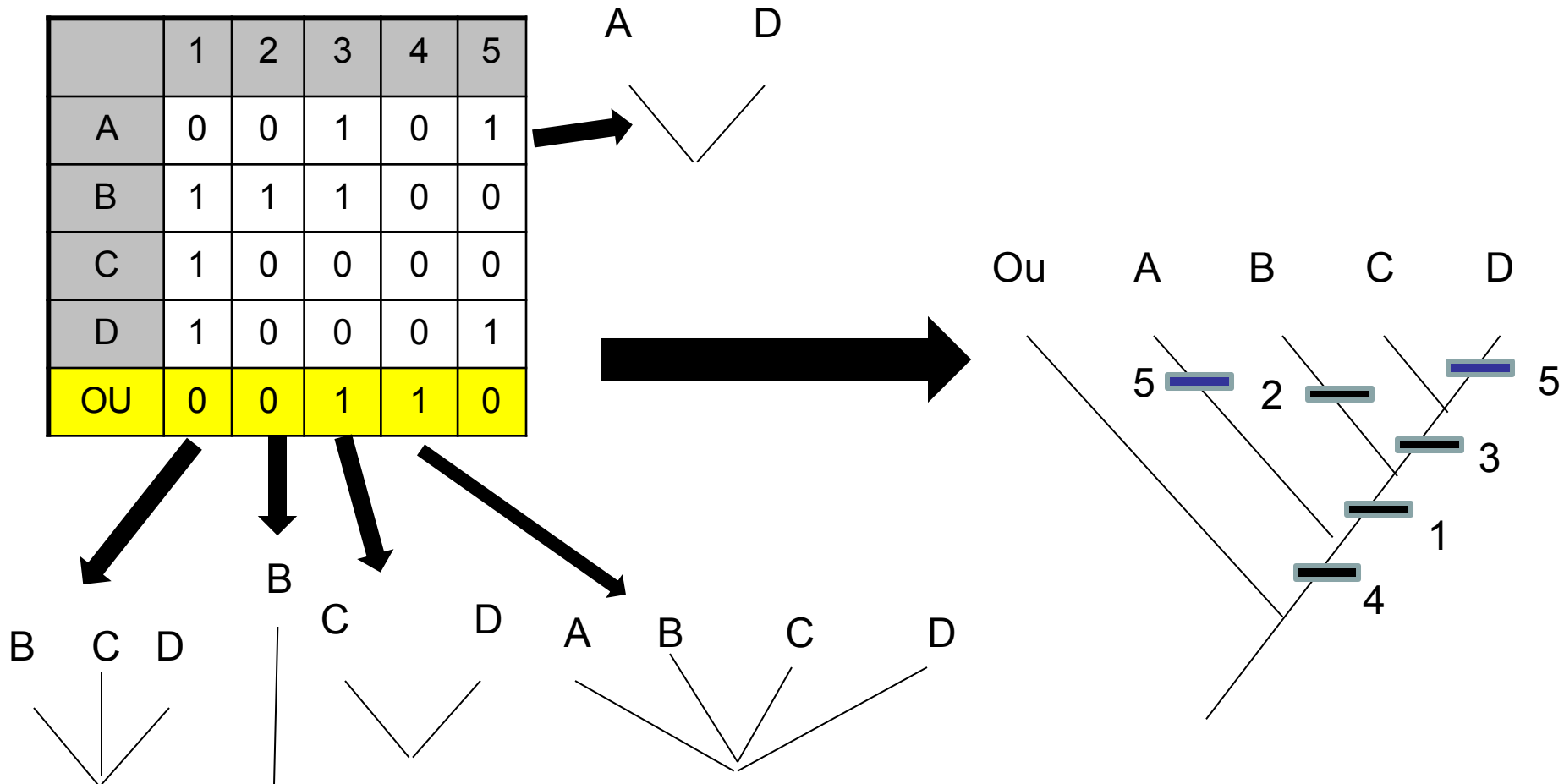
- Determinar homología primaria
- Determinar **polaridad** (Grupo ajeno)
- Identifique congruencias
- Construya el o los árboles

Supuestos sobre homología y polaridad

En ausencia de evidencia en contra, asuma que los caracteres compartidos son resultado de ancestralidad: Homología *a priori*

En ausencia de evidencia en contra, asuma que los caracteres **derivados** compartidos permiten reconocer relaciones

Argumentación Hennigiana



Algunos problemas con la argumentación Hennigiana

- Polarización *a priori* de caracteres (grupo ajeno = ancestral)
- El método asume que **no hay homoplasia** (poco realista)
 - Caracteres inconsistentes con otros violan el modelo Hennigiano
- Imposible evitar errores o malas interpretaciones al codificar caracteres

CRITERIO DE OPTIMALIDAD

- Medida que permite decidir, con base en un conjunto de datos, cuales hipótesis (árboles) son mejores y cuales son peores

1. MÁXIMA PARSIMONIA

Cuchilla de Occam: la mejor hipótesis para explicar un proceso es aquella que requiere el menor número de suposiciones

En inferencia filogenética: el mejor árbol es aquel que explica los datos observados con la menor cantidad de homoplasia posible (menos transformaciones)

1. MÁXIMA PARSIMONIA

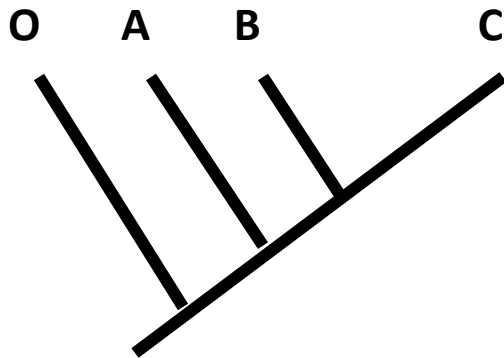
IMPLEMENTACIÓN

1. Construir matriz con base en homología primaria
2. Contar el mínimo número de cambios (pasos) de cada caracter en un árbol determinado
3. Sumar todos los números de pasos para determinar la LONGITUD DEL ÁRBOL
4. Repetir en los otros árboles alternativos y escoger aquel con la menor longitud com el ÁRBOL MÁS PARSIMONIOSO

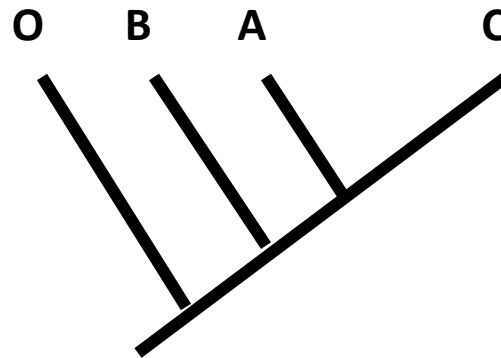
1. MÁXIMA PARSIMONIA

EJEMPLO

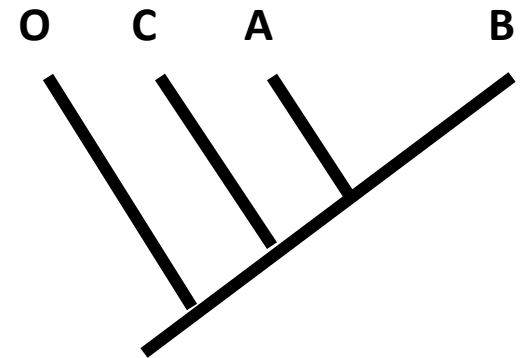
	1	2	3	4	5	6	7	8
O	0	0	0	0	0	0	0	0
A	0	1	0	0	0	1	1	0
B	1	1	0	1	1	1	1	1
C	0	0	1	1	0	0	0	0



Árbol 1



Árbol 2



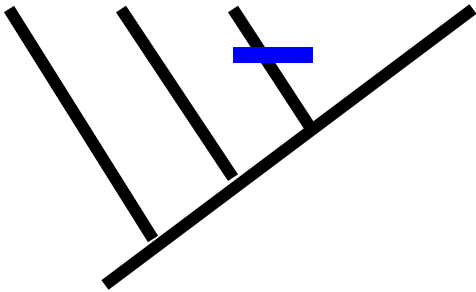
Árbol 3

1. MÁXIMA PARSIMONIA

Árbol 1

	1	2	3	4	5	6	7	8
O	0	0	0	0	0	0	0	0
A	0	1	0	0	0	1	1	0
B	1	1	0	1	1	1	1	1
C	0	0	1	1	0	0	0	0

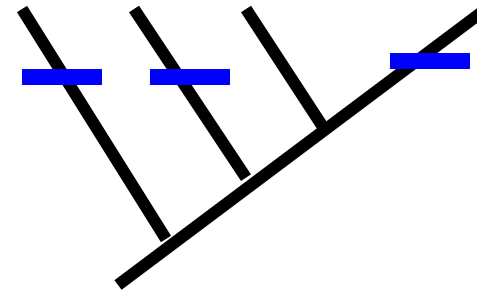
O = 0 A = 0 B = 1 C = 0



0 → 1

1 pasos

O = 0 A = 0 B = 1 C = 0



1 → 0

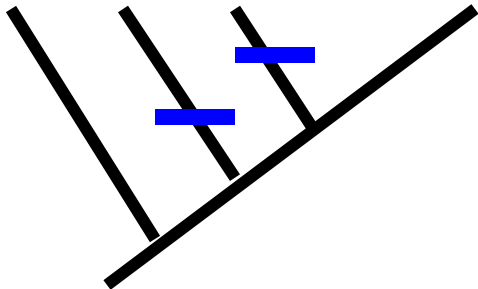
3 pasos

1. MÁXIMA PARSIMONIA

Árbol 1

	1	2	3	4	5	6	7	8
O	0	0	0	0	0	0	0	0
A	0	1	0	0	0	1	1	0
B	1	1	0	1	1	1	1	1
C	0	0	1	1	0	0	0	0

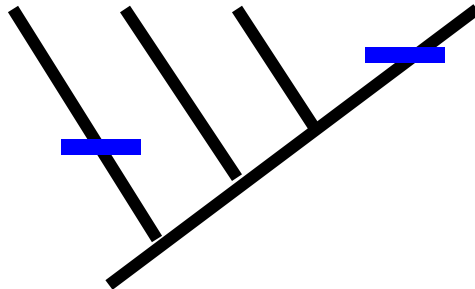
O = 0 A = 1 B = 1 C = 0



0 → 1

2 pasos

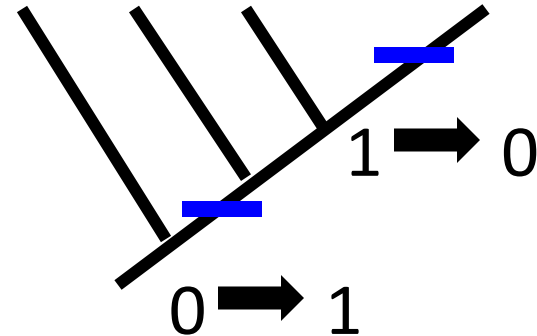
O = 0 A = 1 B = 1 C = 0



1 → 0

2 pasos

O = 0 A = 1 B = 1 C = 0

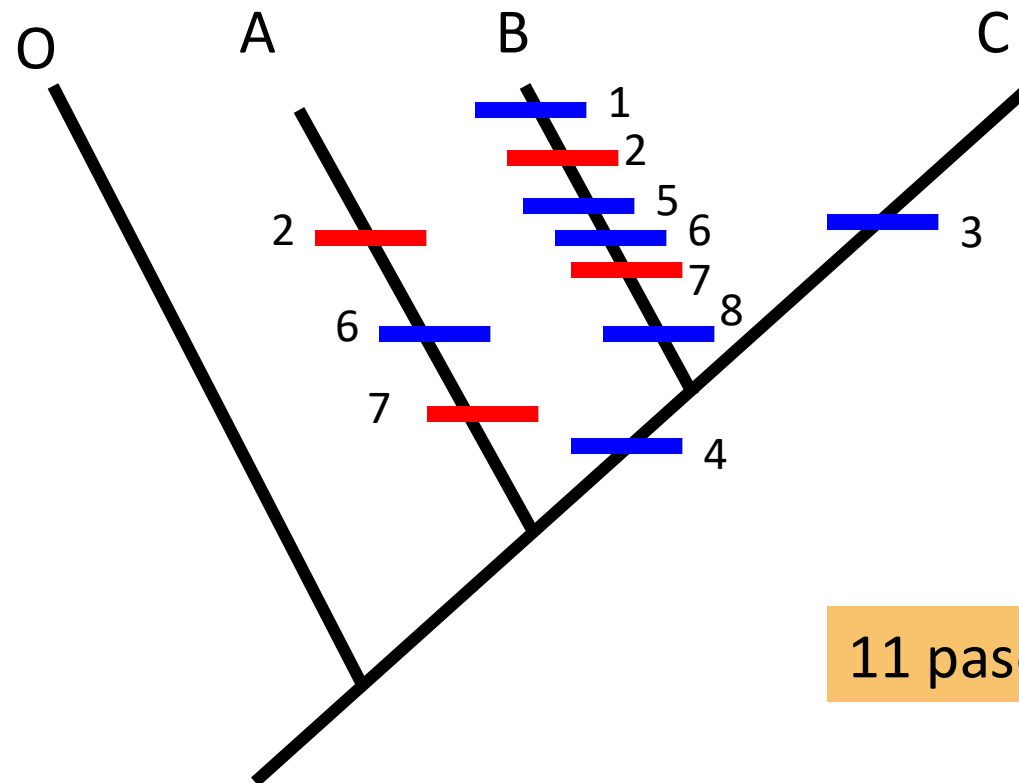


2 pasos

1. MÁXIMA PARSIMONIA

Árbol 1

	1	2	3	4	5	6	7	8
O	0	0	0	0	0	0	0	0
A	0	1	0	0	0	1	1	0
B	1	1	0	1	1	1	1	1
C	0	0	1	1	0	0	0	0

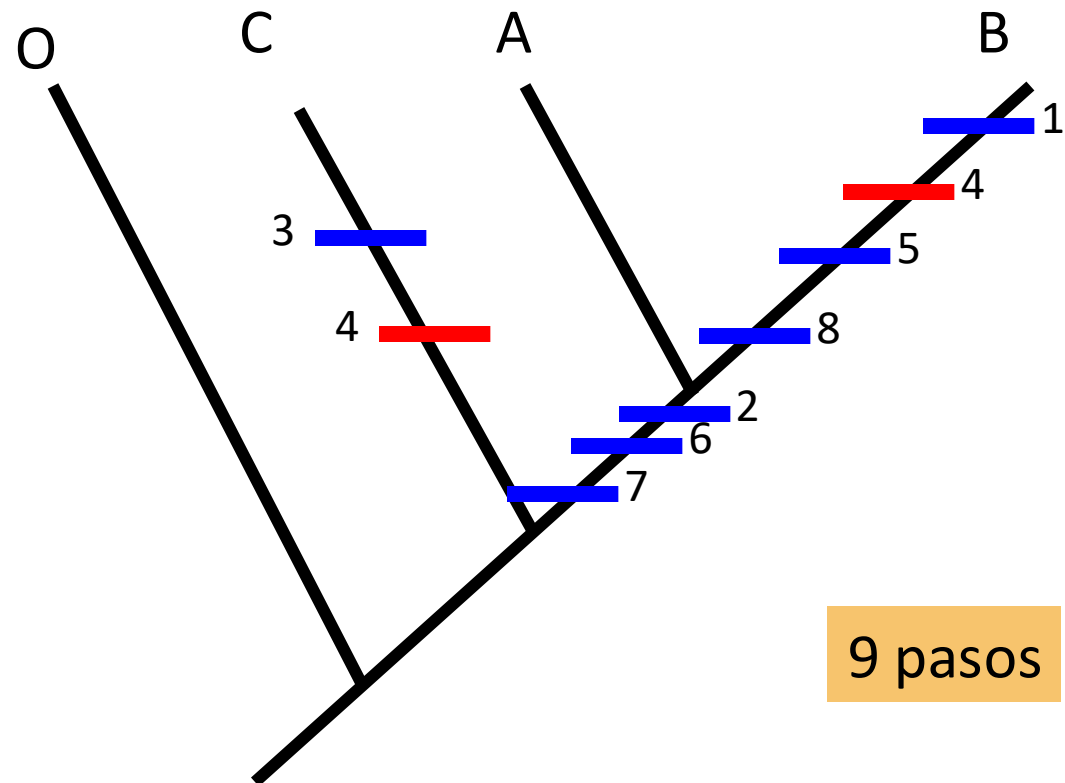


11 pasos

1. MÁXIMA PARSIMONIA

Árbol 3

	1	2	3	4	5	6	7	8
O	0	0	0	0	0	0	0	0
A	0	1	0	0	0	1	1	0
B	1	1	0	1	1	1	1	1
C	0	0	1	1	0	0	0	0



1. MÁXIMA PARSIMONIA

	1	2	3	4	5	6	7	8	
O	0	0	1	0	1	1	0	0	
A	0	1	1	0	1	0	1	0	
B	1	1	1	1	0	0	1	1	
C	0	0	0	1	1	1	0	0	Longitud Total

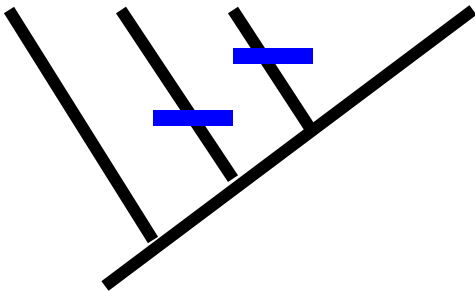
Longitud árbol 1	1	2	1	1	1	2	2	1	11	
Longitud árbol 2	1	2	1	2	1	2	2	1	12	
Longitud árbol 3	1	1	1	2	1	1	1	1	9	Árbol más parsimonioso

Caracteres informativos y no informativos para parsimonia

1. MÁXIMA PARSIMONIA

ACCTRAN y DELTRAN

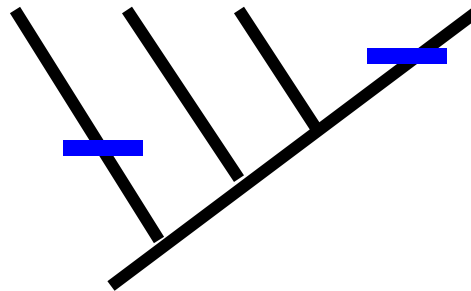
O = 0 A = 1 B = 1 C = 0



0 → 1

2 pasos

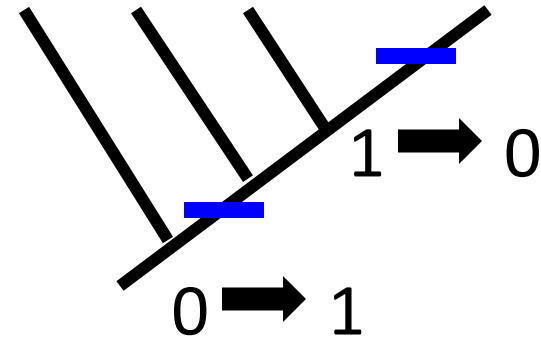
O = 0 A = 1 B = 1 C = 0



1 → 0

2 pasos

O = 0 A = 1 B = 1 C = 0



2 pasos

- ACCTRAN: Transformación acelerada (favorece reversiones)
- DELTRAN: Transformación retrasada (favorece los paralelismos)