

# SESSION 8. EVOLUTION

## **Resolving a criminal case**



## A fatal injection with HIV and C-Hepatitis

To show the relationship b/w different HIV isolates in a criminal case, a **phylogenetic tree analysis** can be conducted.

- *The case took place in Lafayette in Louisiana about 30 years ago.*
- *Maria Jones (Victim), a married nurse of age 20 had a relationship with a married gastroenterologist named Robert White of age 34.*
- *Maria in turn divorced her husband with Robert's promise to divorce his wife but followed through with it.*
- *Maria did give birth to a child after three times of abortions.*
- *Robert eventually became exceedingly jealous and controlling to Maria.*
- *They broke up in July 1994. Before that, Robert had regularly given Maria Vitamin B-12 injection. One day in August 1994, Robert forced to proceed injection.*

## A fatal injection with HIV and C-Hepatitis

- *Two weeks later, Maria consulted a physician with a symptom of her lymph nodes were swollen with a viral infection.*
- *In Dec. She got annual check-up including HIV test and was found HIV positive and Hepatitis C positive.*
- *She turned to Police in Jan. 1995 and did not deal the accusations seriously.*
- *It turned out all seven men Maria reported as sexual contacts for the period 1984-1995 were tested and found to be HIV negative.*
- *During examining Robert's patients' records, Police found a blood sample from a patient with AIDS and another sample from a patient with Hepatitis C.*
- *Prosecution wanted evidence that the HIV carried by Maria was the same as – or very closely related to – the AIDS patient's HIV. In other word, it was important to exclude the possibility that Maria received the infection from some other sources.*

# Molecular phylogeny

## Computational methods of molecular phylogeny

- This criminal case discussed here was the **first time molecular phylogeny was used in court in the USA.**
- Molecular phylogeny – the science of constructing phylogenetic trees using molecular sequence data.
- Tree construction methods rely on analysis of a multiple sequence alignment (MSA).
- Two major approaches: **distance-based** and **character-based**

# Molecular phylogeny methods

## Distance-based approach (computationally simple)

- Neighbor-joining : based on all pairwise comparison of seqs and minimize the total branch length

## Character-based approach (more biologically relevant)

- Maximum-parsimony: an optimality criterion under which the phylogenetic **tree** that minimizes the total number of character-state changes is to be preferred.
- Maximum-likelihood: maximizes tree likelihood given specific parameter values

# Neighbor-joining (NJ)

## □ Building Phylogenetic Trees by Neighbor-Joining:

### □ Algorithm (Given a distance matrix):

Iterate Until 2 Nodes are left:

- For each node find

$$U_i = \sum_{k=1}^N d_{i,k}$$

- Choose pair (i, j) with smallest

$$\delta_{ij} = d_{ij} - \frac{U_i + U_j}{N - 2}$$

- Merge two nodes i and j with a new internal node Y, and find branch lengths by

$$b_{iY} = \frac{1}{2} \left( d_{ij} + \frac{U_i - U_j}{N - 2} \right) \quad b_{jY} = d_{ij} - b_{iY}$$

- Update the distance matrix using

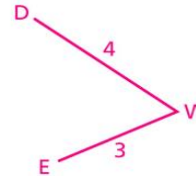
$$d_{Yk} = \frac{1}{2} (d_{ik} + d_{jk} - d_{ij})$$

# Neighbor-joining (NJ)

STEP 1 ( $N = 5$ )

	$d_{ij}$				$U_i$	$3\delta_{ij}$				
	B	C	D	E		B	C	D	E	
A	5	4	9	8	26	-40	-36	-32	-32	A
B		5	10	9	29		-36	-32	-32	B
C			7	6	22			-34	-34	C
D				7	33				-42	D
E					30					E

D and E are neighbors through internal node W with  $d_{DW} = \frac{1}{2} \left( 7 + \frac{33-30}{3} \right) = 4$  and  $d_{EW} = 7 - 4 = 3$ .



$U_a = 26$   $U_b = 29$   $U_c = 22$   $U_d = 33$   $U_e = 30$

$\Delta(a,b) = 5 - (26+29)/(5-2) = 5 - 18.3 = -13.3$   
 $\Delta(a,c) = 4 - (26+22)/3 = 12$   
 $\Delta(a,d) = 9 - (26+33)/3 = 9 - 19.67 = 10.67$   
 $\Delta(a,e) = 8 - (26+30)/3 = 8 - 18.67 = 10.67$   
 $\Delta(b,c) = 5 - (29+22)/3 = 5 - 17 = -12$   
 $\Delta(b,d) = 10 - (29+33)/3 = 10 - 20.67 = 10.67$   
 $\Delta(b,e) = 9 - (29+30)/3 = 9 - 19.67 = 10.67$   
 $\Delta(c,d) = 7 - (22+33)/3 = 7 - 18.33 = 11.33$   
 $\Delta(c,e) = 6 - (22+30)/3 = 6 - 17.33 = 11.33$   
 $\Delta(d,e) = 7 - (33+30)/3 = 7 - 21 = 14$

$D(dw) = \frac{1}{2}(7 + (33-30)/3) = 4$   $D(ew) = 7 - 4 = 3$

$D(wa) = \frac{1}{2}(9 + 8 - 7) = 5$   
 $D(wb) = \frac{1}{2}(10 + 9 - 7) = 6$   
 $D(wc) = \frac{1}{2}(7 + 6 - 7) = 3$

$$U_i = \sum_{k=1}^N d_{i,k}$$

$$\delta_{ij} = d_{ij} - \frac{U_i + U_j}{N - 2}$$

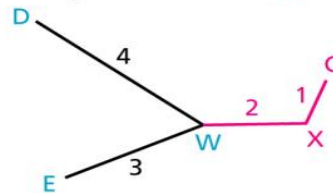
$$b_{iY} = \frac{1}{2} \left( d_{ij} + \frac{U_i - U_j}{N - 2} \right) \quad b_{jY} = d_{ij} - b_{iY}$$

$$d_{YK} = \frac{1}{2} (d_{ik} + d_{jk} - d_{ij})$$

# Neighbor-joining (NJ)

STEP 2 (N = 4)				$d_{ij}$	$U_i$	$2\delta_{ij}$			
	B	C	W			B	C	W	
A	5	4	5	14	-20	-18	-18		A
B		5	6	16		-18	-18		B
C			3	12			-20		C
W				14					W

C and W are neighbors through internal node X with  $d_{CX} = \frac{1}{2} \left( 3 + \frac{12-14}{2} \right) = 1$  and  $d_{WX} = 3 - 1 = 2$ .



$$U_i = \sum_{k=1}^N d_{i,k}$$

$$\delta_{ij} = d_{ij} - \frac{U_i + U_j}{N - 2}$$

$$b_{iY} = \frac{1}{2} \left( d_{ij} + \frac{U_i - U_j}{N - 2} \right) \quad b_{jY} = d_{ij} - b_{iY}$$

$$d_{YK} = \frac{1}{2} (d_{ik} + d_{jk} - d_{ij})$$



# Neighbor-joining (NJ)

STEP 3 ( $N = 3$ )		$d_{ij}$		$U_i$	$\delta_{ij}$	
	B	X			B	X
A	5	3	8	-12	-12	A
B		4	9		-12	B
X			7			X

Three alternatives (of which here we choose one of the two with an internal node):

A and X are neighbors through internal node Y with  $d_{AY} = 2$  and  $d_{XY} = 1$  or

B and X are neighbors through internal node Y with  $d_{BY} = 3$  and  $d_{XY} = 1$ .

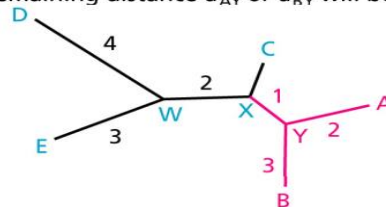
Whichever is chosen, the remaining distance  $d_{AY}$  or  $d_{BY}$  will be found in the next  $d_{ij}$  matrix.

$$U_i = \sum_{k=1}^N d_{i,k}$$

$$\delta_{ij} = d_{ij} - \frac{U_i + U_j}{N - 2}$$

$$b_{iY} = \frac{1}{2} \left( d_{ij} + \frac{U_i - U_j}{N - 2} \right) \quad b_{jY} = d_{ij} - b_{iY}$$

$$d_{Yk} = \frac{1}{2} (d_{ik} + d_{jk} - d_{ij})$$



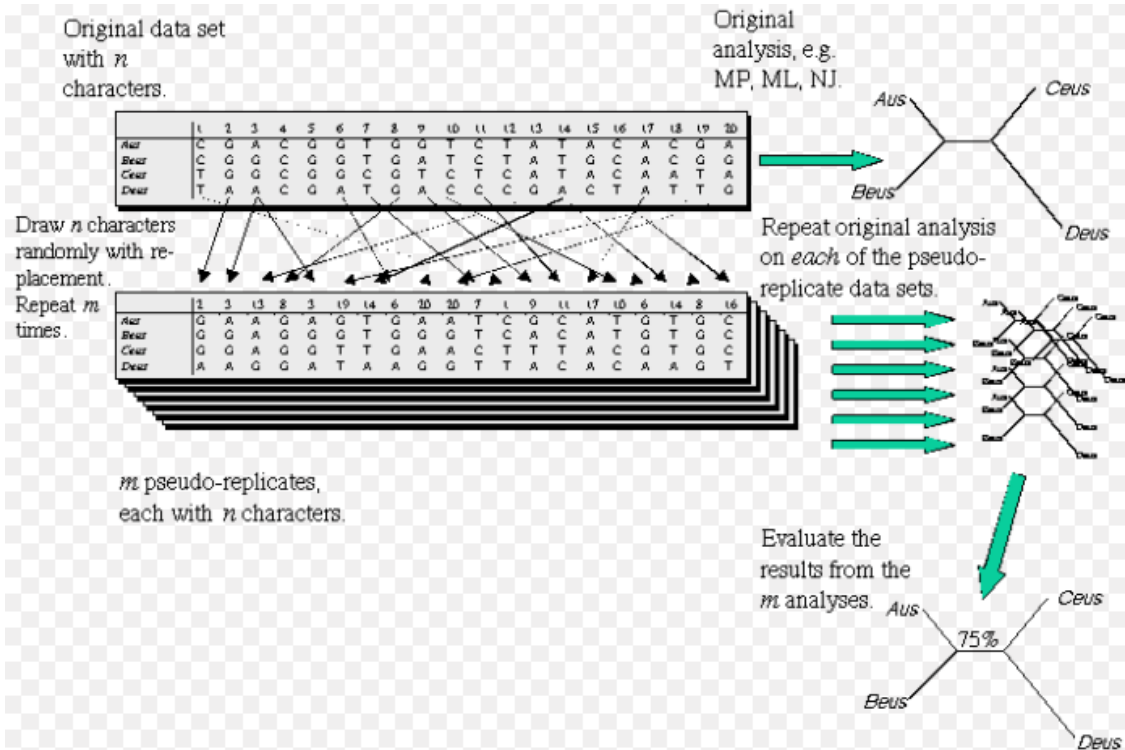
## Bootstrapping (Accuracy and Robustness of tree)

**Bootstrapping** refers to any test or metric that relies **on random sampling with replacement** that allows assigning measures of accuracy

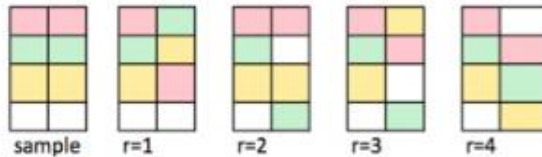
Sampling from the original sample of size  $N$  to form a new sample (called a 'resample' or bootstrap sample) that is also of size  $N$ .

**Repeats 1000-10000 times** of resampling and makes a histogram of means.

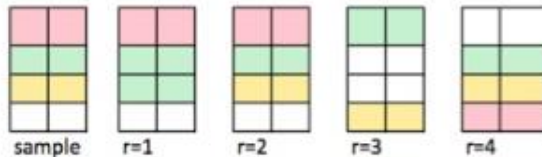
# Bootstrapping (Accuracy and Robustness of tree)



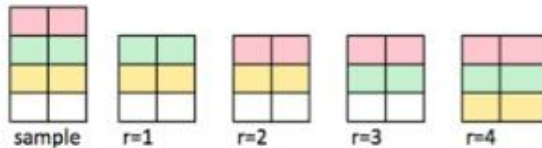
# Other randomization and sampling methods



Permutation  
Randomization test



Bootstrap

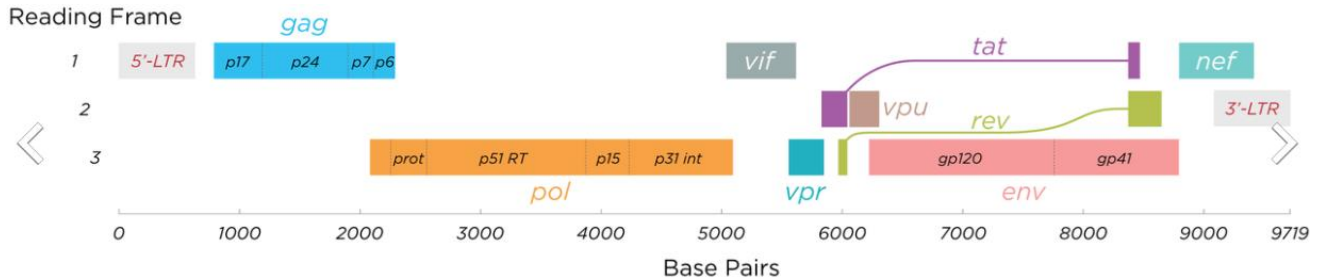


Jackknife



cross validation

# HIV has RNA genome encoding 20 genes



- HIV RNA genome comprises 9749 nt
- Two genes, **env** and **RT** are often used for phylogenetic analysis
- **Env** encodes gp160 -> (gp120 + gp41) presenting in a surface
- **RT** encodes reverse transcriptase that makes dsDNA from RNA.

# Phylogenetic analysis of HIVs

DNA samples from Victim, patients, and controls (from Lafayette regions)

PCR DNAs with primers of env and RT and sequenced them.

Computational analysis of phylogenetic trees with the sequences using ClustalW

Data is publicly available in NCBI Entrez (AY156734-AY156907)

- 132 env sequences
- 42 RT sequences

# Phylogenetic analysis of HIVs

clustalw2 rt.fa

```
Group 34: Sequences: 16      Score:12713
Group 35: Sequences: 17      Score:12690
Group 36: Sequences: 37      Score:12479
Group 37: Sequences: 38      Score:12695
Group 38: Sequences: 39      Score:12760
Group 39: Sequences: 40      Score:12754
Group 40: Sequences: 41      Score:12802
Group 41: Sequences: 42      Score:12609
Alignment Score 3678234
```

```
CLUSTAL-Alignment file created [rt.aln]
```

clustalw2 rt.aln -tree

```
Sequence 34: gi|24209997|gb|AY156791.1| 805 bp
Sequence 35: gi|24209965|gb|AY156775.1| 805 bp
Sequence 36: gi|24209991|gb|AY156788.1| 805 bp
Sequence 37: gi|24209987|gb|AY156786.1| 805 bp
Sequence 38: gi|24210001|gb|AY156793.1| 805 bp
Sequence 39: gi|24209963|gb|AY156774.1| 805 bp
Sequence 40: gi|24209981|gb|AY156783.1| 805 bp
Sequence 41: gi|24209971|gb|AY156778.1| 805 bp
Sequence 42: gi|24209957|gb|AY156771.1| 805 bp
```

```
Phylogenetic tree file created: [rt.ph]
```





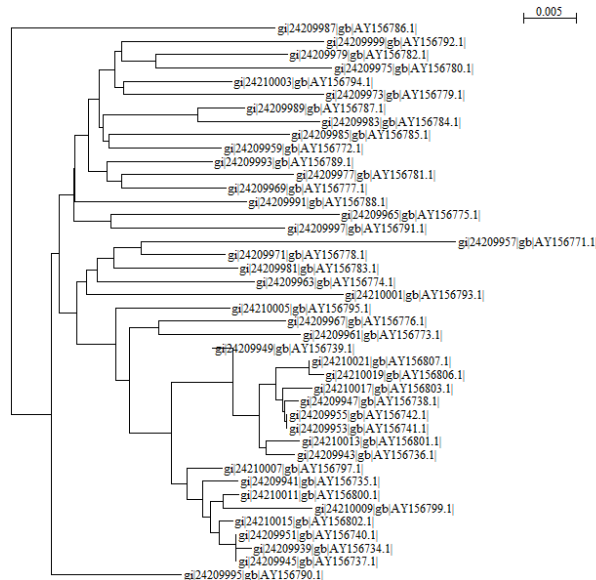
# Phylogenetic analysis of HIVs

clustalw2 rt.aln -bootstrap=1000

Sequence 40: gi|24209981|gb|AY156783.1| 805 bp  
Sequence 41: gi|24209971|gb|AY156778.1| 805 bp  
Sequence 42: gi|24209957|gb|AY156771.1| 805 bp

Bootstrap output file created: [rt.phb]

Njplot  rt.phb



## Reformat\_giline.py

>gi|24209939|gb|AY156734.1| HIV-1 clone P1.BCM.RT from USA reverse transcriptase (pol) gene, partial cds

P = patient, V=Victim, LA = Lafayette area control



>P1.BCM.RT

## Reformat\_giline.py

```
#!/usr/bin/python

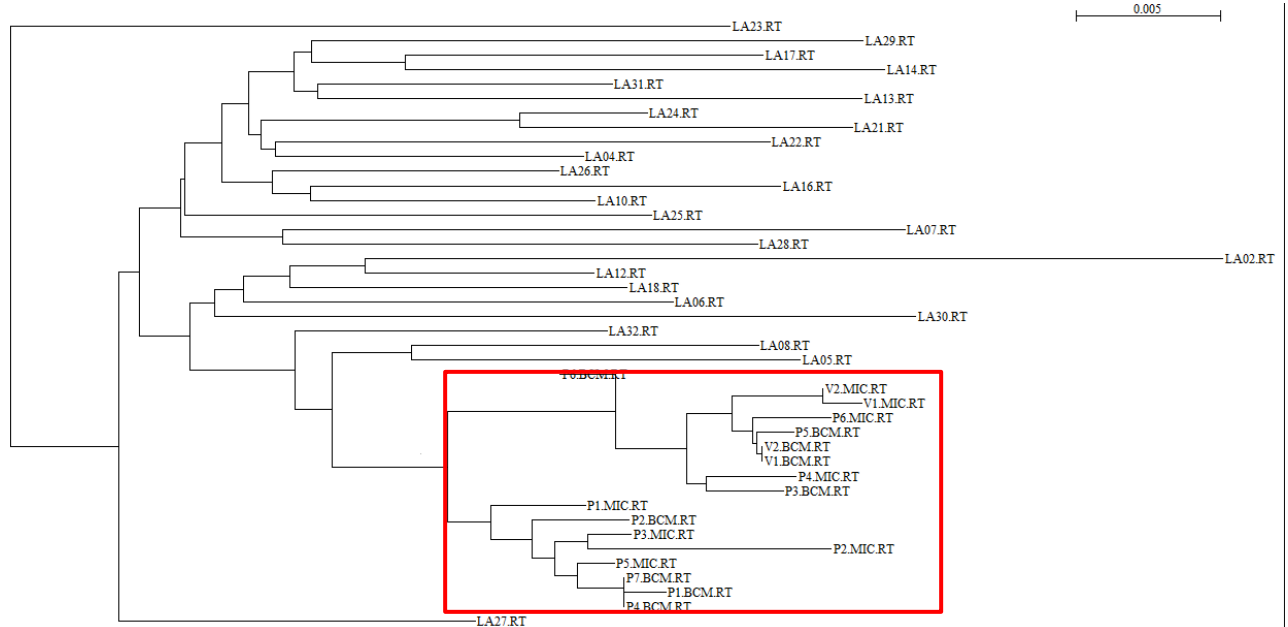
import re
import sys

for line in open('rt.fa'):
    line = line.rstrip()
    match = re.search('>.*clone (\S+) ', line)
    if match:
        sys.stdout.write('>')
        print match.group(1)
    else:
        print line
```

`python reformat_giline.py rt.fa >rt_reformatted.fa`

# Phylogenetic analysis with a new format

clustalw2 rt\_reformat.aln -bootstrap=1000



# Pseudocode for NJ method

```
#####  
### Pseudocode for neighbor-joining method  
### This pseudocode was designed for the computational biology practice  
### Jin-Wu Nam, April 25th 2016  
#####  
  
def alnToDMatrix(filename):  
    # upload aln file  
    # count nt differences b/w any pairs  
    # create a distance matrix  
    return DMatrix  
  
def calcTotalDistances(DMatrix):  
    # create a dictionary that contains total distance of each node  
    return TotalDistanceD  
  
def calcDeltaMatrix(DMatrix, TotalDistanceD):  
    # calculate deltaMatrix using DMMatrix minus Total Distance Matrix  
    return deltaMatrix  
  
def findMinimumDistPair(deltaMatrix):  
    # chrSet = ['S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']  
    # newNodeCharacter = chrSet[len(deltaMatrix)]  
    # please find a pair that has the minimum delta value  
    return minimumPair, newNodeCharacter  
  
def distancesFromNewNode(minimumPair, DMatrix, TotalDistanceD):  
    # minimumPair = findMinimumDistPair  
    # using the minimum pair and create a new node.  
    # calculate all distances from the new node to other nodes  
    # create a new DMMatrix  
    return newDMatrix  
  
DMatrix = alnToDMatrix(aln_filename)  
while len(DMatrix) > 1:  
    TotalDistanceD = calcToDistances(DMatrix)  
    deltaMatrix = calcDeltaMatrix(DMatrix, TotalDistanceD)  
    minimumPair = findMinimumDistPair(deltaMatrix)  
    DMatrix = distancesFromNewNode(minimumPair, DMatrix, TotalDistanceD)
```