# Parsimony

Genome 559: Introduction to Statistical and Computational Genomics

**Elhanan Borenstein**

# Who am I?

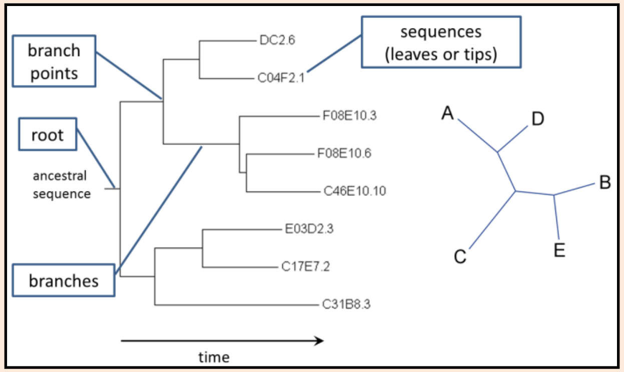
- Faculty at Genome Sciences
- Computational (systems) biologist
- Training: CS, physics, hi-tech, biology
- Research interests: Metagenomics and the Human Microbiome | Complex biological networks | Evolutionary Systems Biology

# What will change?

- **Not much!**
- **Informatics**: From sequence to genes and to systems
- **Programming:**
    - More emphasis on design and coding practices; Coding style
    - Tip of the day
- Website: http://elbo.gs.washington.edu/courses/GS_559_13_wi/

# A quick review

- Trees:
  - Represent sequence relationships
  - A sequence tree has a topology and branch lengths (distances)
  - The number of tree topologies grows very fast!



- Distance trees
  - Compute pairwise corrected distances
  - Build tree by sequential clustering algorithm (UPGMA or Neighbor-Joining).
  - These algorithms don't consider all tree topologies, so they are very fast, even for large trees.

# “Maximum Parsimony Algorithm”

*A fundamentally different method:*

Instead of ***reconstructing*** a tree,
we will ***search*** for the best tree.

*“Pluralitas non est ponenda sine necessitate”*

# (Maximum) Parsimony Principle

- *“Pluralitas non est ponenda sine necessitate”* (plurality should not be posited without necessity)
  William of Ockham

- Occam’s Razor: Of two equivalent theories or explanations, all other things being equal, the simpler one is to be preferred.

**William of Ockham**
(c. 1288 – c. 1348)

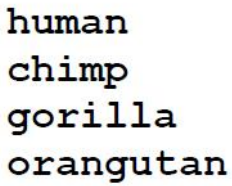
- "when you hear hoof beats, think horses, not zebras“
  Medical diagnosis
- The KISS principle: "Keep It Simple, Stupid!"
  Kelly Johnson, Engineer

- “Make everything as simple as possible, but not simpler”
  Albert Einstein

# Parsimony principle for phylogenetic trees

*Find the tree that requires the fewest evolutionary changes!*

# Consider 4 species



```
human
chimp
gorilla
orangutan
```

# Consider 4 species

Sequence data:

| | 123456 |
|---|---|
| human | agtctc |
| chimp | agagtc |
| gorilla | cggcag |
| orangutan | cgggac |

positions in alignment
(usually called "sites“)

- The same approach would work for any discrete property that can be associated with the various species:
    - Gene content (presence/absence of each gene)
    - Morphological features (e.g., “has wings”, purple or white flowers)
    - Numerical features (e.g., number of bristles)

# Consider 4 species

Sequence data:

```
                 123456
                 ______
human            agtctc
chimp            agagtc
gorilla          cggcag
orangutan        cgggac
```

positions in alignment
(usually called "sites“)

***Parsimony Algorithm***

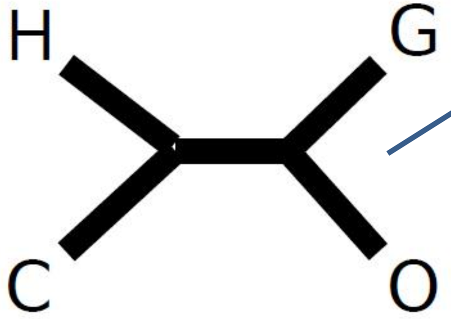
1) *Construct all possible trees*
2) ***For each site in the alignment and for each tree*** *count the minimal number of changes required*
3) *Add all sites up to obtain the total number of changes for each tree*
4) *Pick the tree with the lowest score*

# Consider 4 species

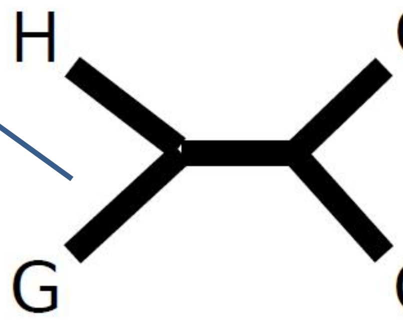
Sequence data:

```
           123456
human      agtctc
chimp      agagtc
gorilla    cggcag
orangutan  cgggac
```

All possible unrooted trees:

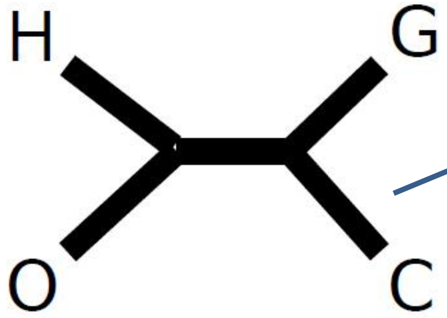


H closest to C



**or**

H closest to G



**or**

H closest to O

# Consider 4 species

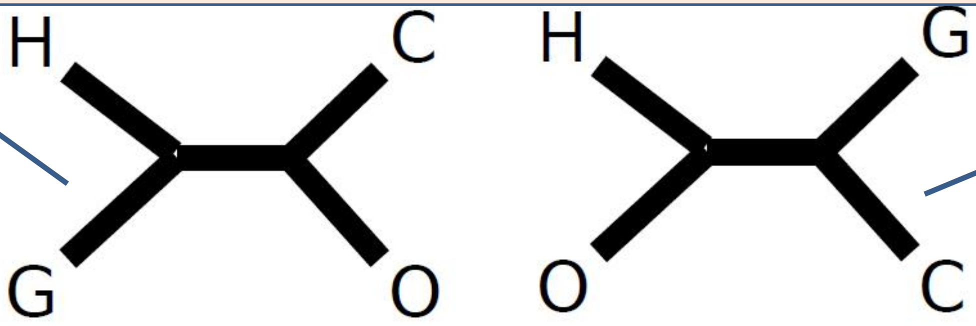
Sequence data:

```
                123456
human           agtctc
```

All

un

*For each site and for each tree count the minimal number of changes required:*



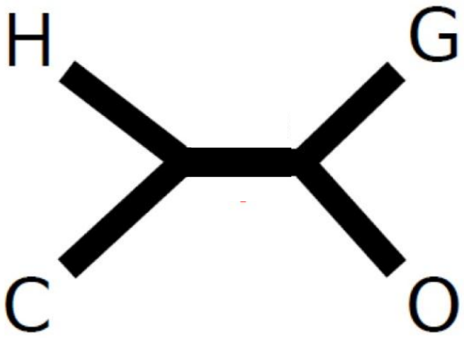
H closest to G

**or**

H closest to O

# Consider site 1

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |

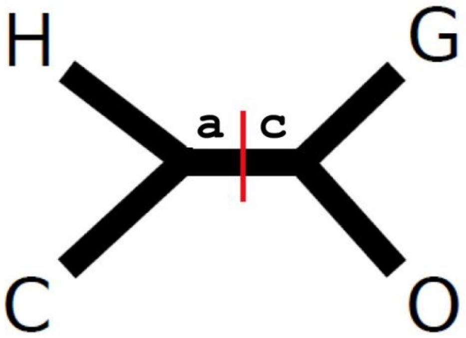


What is the minimal number of evolutionary changes that can account for the observed pattern?

***(Note: This is the “small parsimony” problem)***

# Consider site 1

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |

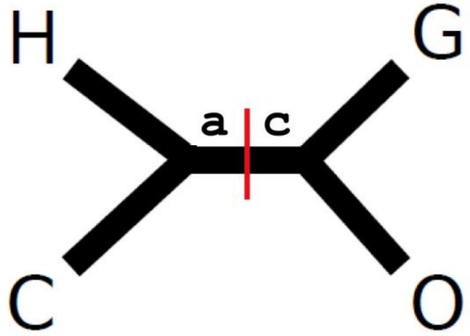


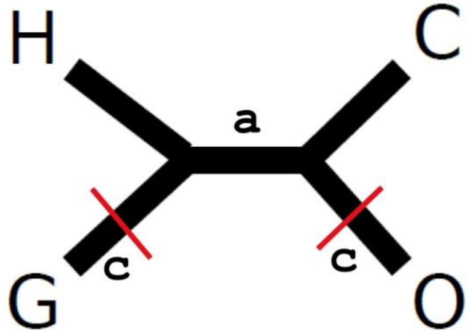
What is the minimal number of evolutionary changes that can account for the observed pattern?

***(Note: This is the “small parsimony” problem)***

# Consider site 1

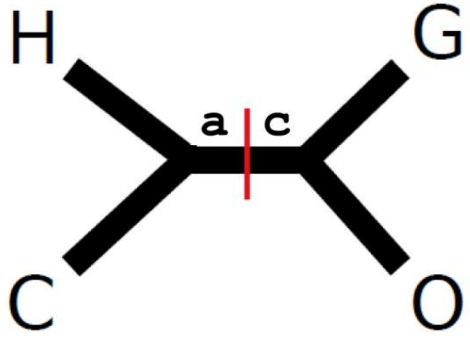
| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |

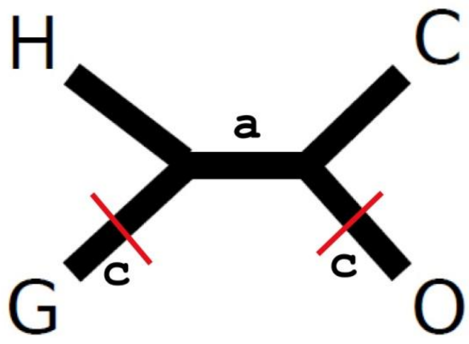


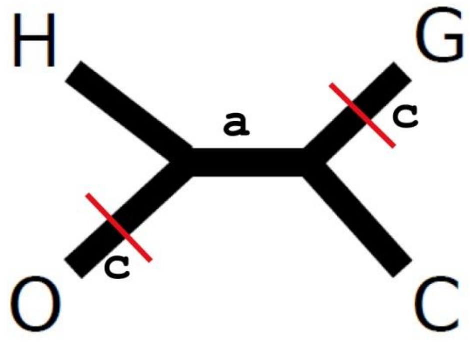


# Consider site 1

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |



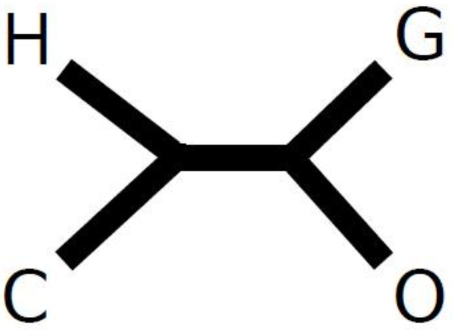


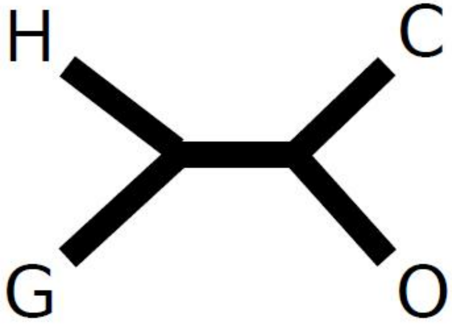


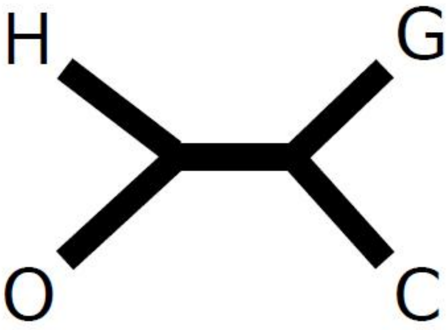
# Consider site 2

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| **human** | a | g | t | c | t | c |
| **chimp** | a | g | a | g | t | c |
| **gorilla** | c | g | g | c | a | g |
| **orangutan** | c | g | g | g | a | c |

Uninformative
(no changes)

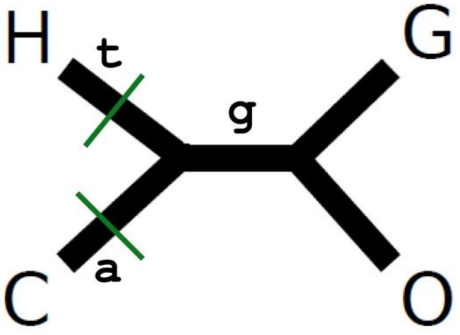


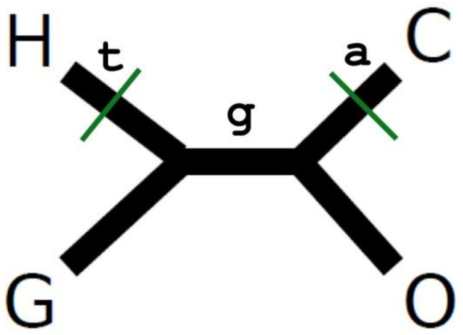


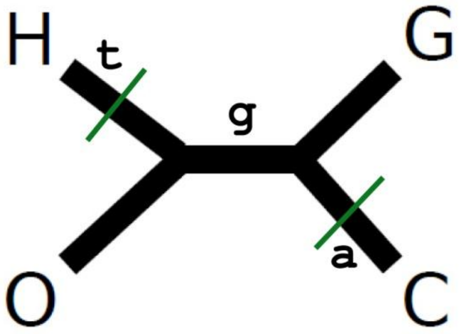


# Consider site 3

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |

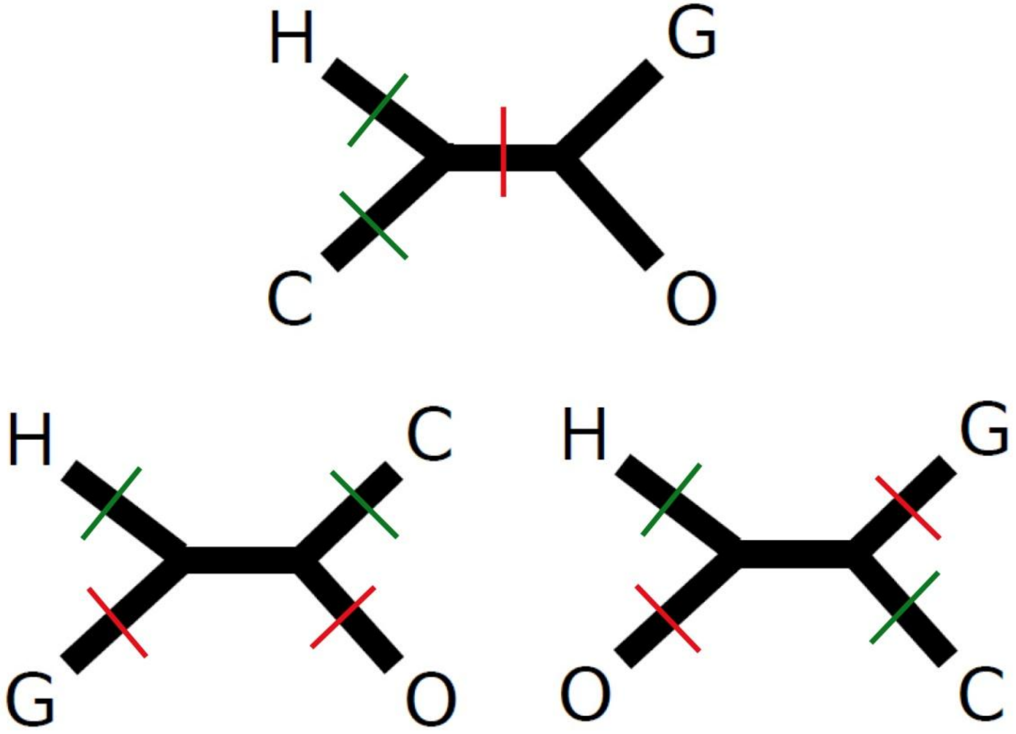






# Put sites 1 and 3 together

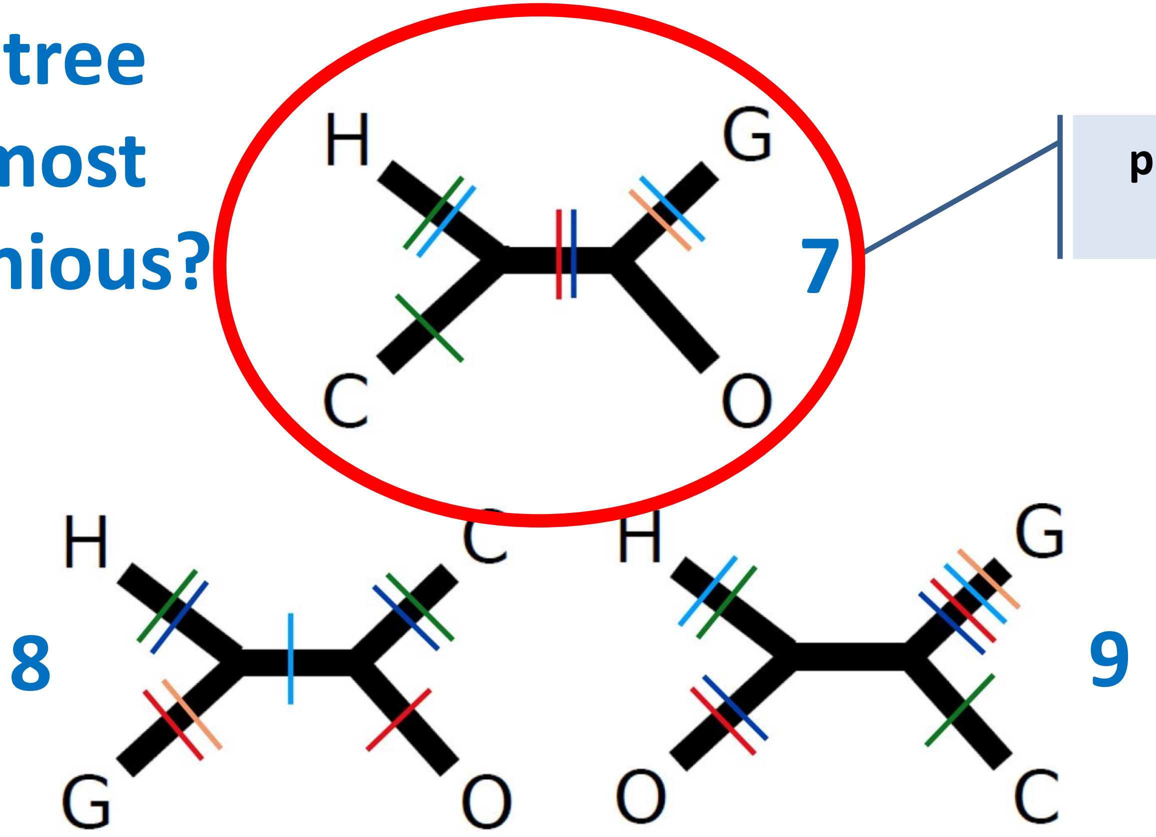
| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |



# Now put all of them together

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| human | a | g | t | c | t | c |
| chimp | a | g | a | g | t | c |
| gorilla | c | g | g | c | a | g |
| orangutan | c | g | g | g | a | c |

**Which tree is the most parsimonious?**



**parsimony score**

# The parsimony algorithm

1) *Construct all possible trees*

2) *For each site in the alignment and for each tree count the minimal number of changes required*

3) *Add all sites up to obtain the total number of changes for each tree*

4) *Pick the tree with the lowest score*

# The parsimony algorithm

**Too many!**

1) *Construct all possible trees*
2) *For each site in the alignment and for each tree count the minimal number of changes required*
3) *Add all sites up to obtain the total number of changes for each tree*
4) *Pick the tree with the lowest score*

# The parsimony algorithm

1) *Construct all possible trees*

**Too many!**

2) *For each site in the alignment and for each tree count the minimal number of changes required*

**How?**

3) *Add all sites up to obtain the total number of changes for each tree*

4) *Pick the tree with the lowest score*

# The parsimony algorithm

1) *Construct all possible trees*

**Too many!**

Search algorithm

2) *For each site in the alignment and for each tree count the minimal number of changes required*

**How?**

Fitch’s algorithm

3) *Add all sites up to obtain the total number of changes for each tree*

4) *Pick the tree with the lowest score*