Реконструкция филогении (окончание)

ФББ, IV семестр,
весна 2025

С.А.Спирин

Переборные методы

Алгоритм, реализующий переборный метод, должен включать:

а) критерий сравнения деревьев (какая из двух топологий лучше соответствует исходным данным?)

б) алгоритм поиска лучшего по критерию дерева.

Пример критерия
(метод наименьших квадратов, OLS — ordinary least squares)

Пусть дана матрица расстояний и топология дерева;
i, j — две последовательности, тогда мы имеем расстояние $d(i,j)$ из матрицы. Приписав ветвям дерева длину, будем иметь расстояние $d'(i,j)$ «по дереву».

Подберём длины ветвей так, чтобы сумма величин $(d(i,j) - d'(i,j))^2$ (по всем парам листьев i,j) была наименьшей.

Это наименьшее значение и будет критерием качества: будем считать ту топологию лучшей, для которой это значение получится меньшим.

Поиск лучшего дерева

Имеется единственная топология (бинарного и неукоренённого) дерева с тремя листьями, три разных топологии деревьев с четырьмя листьями,
15 топологий деревьев с пятью листьями,
… … … …
~ 2 млн. топологий деревьев с десятью листьями,
… … … …
~ 8 трлн. топологий деревьев с 15 листьями,
… … … …

Триллионы проверок компьютер будет делать слишком долго.
А ведь приходится строить деревья и с сотней листьев…

Поиск лучшего дерева

Имеется единственная топология (бинарного и неукоренённого) дерева с тремя листьями, три разных топологии деревьев с четырьмя листьями,
15 топологий деревьев с пятью листьями,
… … … …
~ 2 млн. топологий деревьев с десятью листьями,
… … … …
~ 8 трлн. топологий деревьев с 15 листьями,
… … … …

Триллионы проверок компьютер будет делать слишком долго.
А ведь приходится строить деревья и с сотней листьев…

Поэтому программы, реализующие переборные методы, практически никогда не включают **полный** перебор всех возможных деревьев

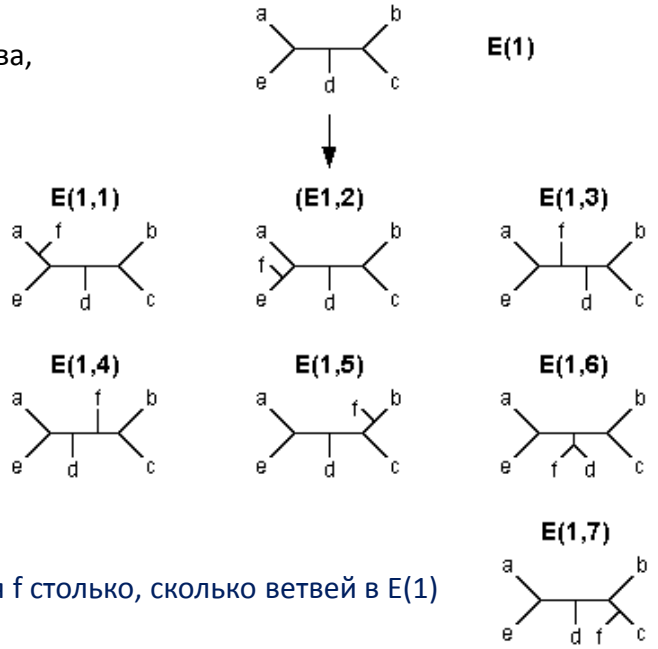
Поиск лучшего дерева: «выращивание»

Найдём лучшее дерево для небольшой части последовательностей.
Будем добавлять последовательности по одной, каждый раз находя лучшее место в уже построенном дереве.

Реализовано в FastME для двух критериев качества дерева, в help'е называется "TaxAdd"

Дерево с *N* листьями всегда имеет 2*N*–3 ветви. Поэтому, чтобы вырастить дерево с *N* листьями, нужно проанализировать $3 + 5 + \ldots + (2N–5) = (N–3)(N–5)$ деревьев.

Уже для *N*=10 это число меньше числа всех деревьев в 32175 раз!



Вариантов для f столько, сколько ветвей в E(1)

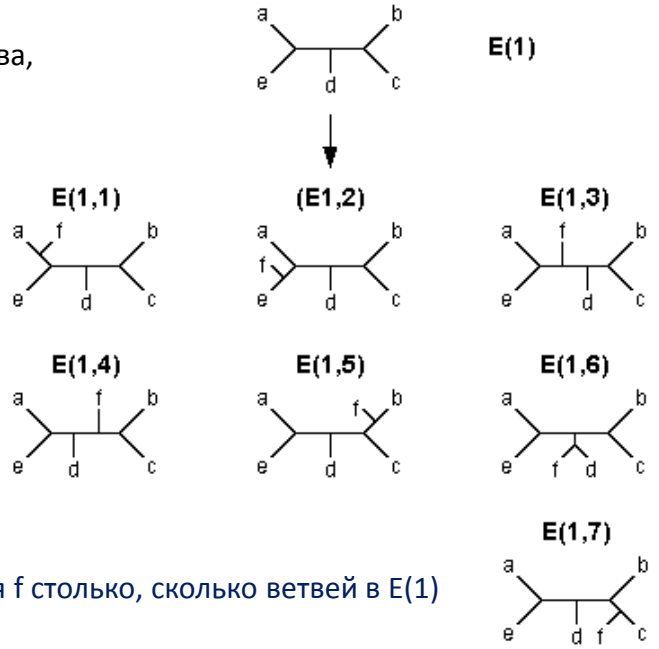
Fig. 5.3 in https://www.megasoftware.net/mega1_manual/Phylogeny.html

Поиск лучшего дерева: «выращивание»

Найдём лучшее дерево для небольшой части последовательностей.
Будем добавлять последовательности по одной, каждый раз находя лучшее место в уже построенном дереве.

Реализовано в FastME для двух критериев качества дерева, в help'е называется "TaxAdd"

Выращивание ("taxon addition", "greedy algorithm") не гарантирует нахождение лучшего по критерию дерева, но при хороших данных не приводит к большим ошибкам.



Вариантов для f столько, сколько ветвей в E(1)

Fig. 5.3 in https://www.megasoftware.net/mega1_manual/Phylogeny.html

Поиск лучшего дерева: просмотр соседних деревьев

Построим сначала черновое дерево, а затем попробуем его улучшить.

Черновое дерево можно: (а) взять случайным образом; (б) «вырастить» (в том числе с применением более быстрого критерия); (в) взять из результатов другого алгоритма.

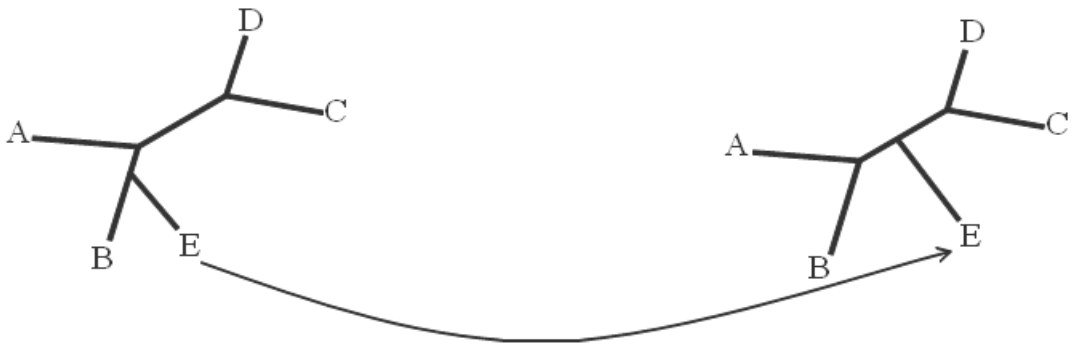
Улучшать будем, просматривая соседние деревья.

Если соседнее дерево по критерию лучше данного, то перейдём к нему и проанализируем его соседей и т.д., пока не получим дерево, которое лучше всех своих соседей (локальный максимум)

Поиск лучшего дерева: просмотр соседних деревьев

Что такое «соседние деревья»?

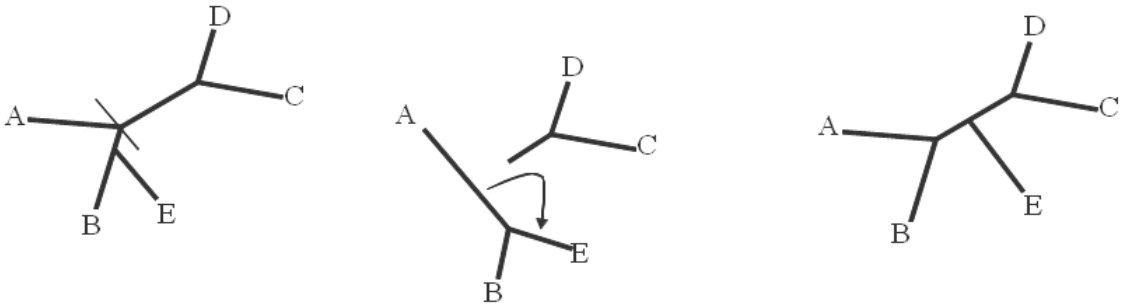
- Оторвём один лист и «привьём» его на другую ветвь



Поиск лучшего дерева: просмотр соседних деревьев

Что такое «соседние» деревья

- Можно проделать аналогичную операцию с целой кладой



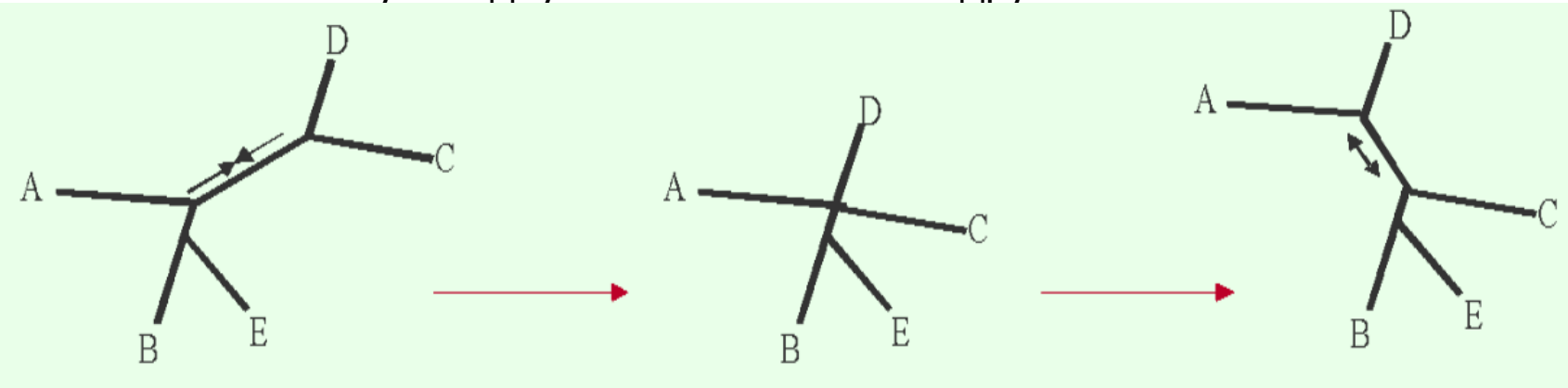
Такая операция обычно называется “SPR” : Subtree Pruning and Regrafting

Сколько таких соседей у дерева?

Поиск лучшего дерева: просмотр соседних деревьев

Что такое «соседние деревья»?

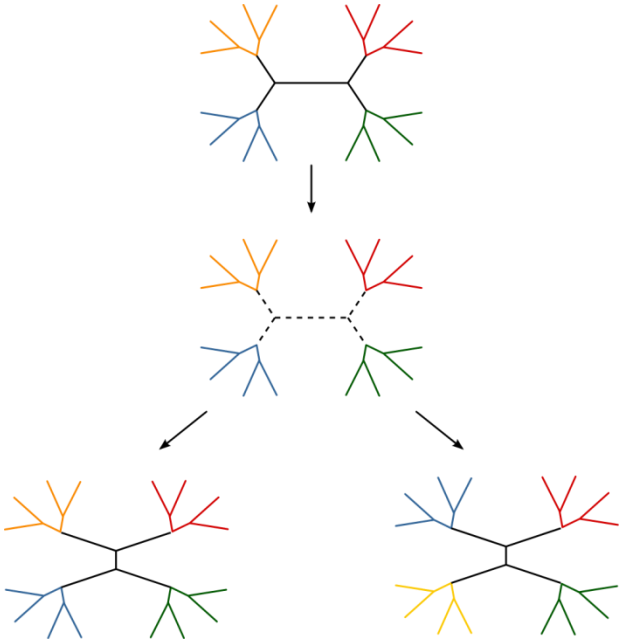
Можно «схлопнуть» одну ветвь и заменить её другой



Это называется NNI = Nearest Neighbor Interchange

Сколько таких соседей у дерева?

По два NNI каждую нетривиальную ветвь



https://en.wikipedia.org/wiki/Tree_rearrangement

Поиск лучшего дерева

- Строим черновое дерево:
  - прямым методом *или*
  - выращиванием с использованием того же критерия качества *или*
  - выращиванием с использованием другого критерия (вычисляемого быстрее, например максимальной экономии при основном критерии наибольшего правдоподобия)
- Анализируем соседние деревья (NNI или SPR),

если находим среди соседей лучшее дерево, берём за основу его

- Повторяем предыдущий пункт, пока текущее дерево не окажется лучше всех своих соседей

Переборные методы

Название метода совпадает с названием **критерия качества**

Алгоритмы поиска одни и те же для всех

- Максимальной экономии
(или «бережливости», **maximum parsimony**, MP)
- Наибольшего правдоподобия (**maximum likelihood**, ML)
- Наименьших квадратов (**least squares**, LS)
- Фитча – Марголиаша (**Fitch – Margoliash**, FM)
- Минимальной эволюции (**minimum evolution**, ME)

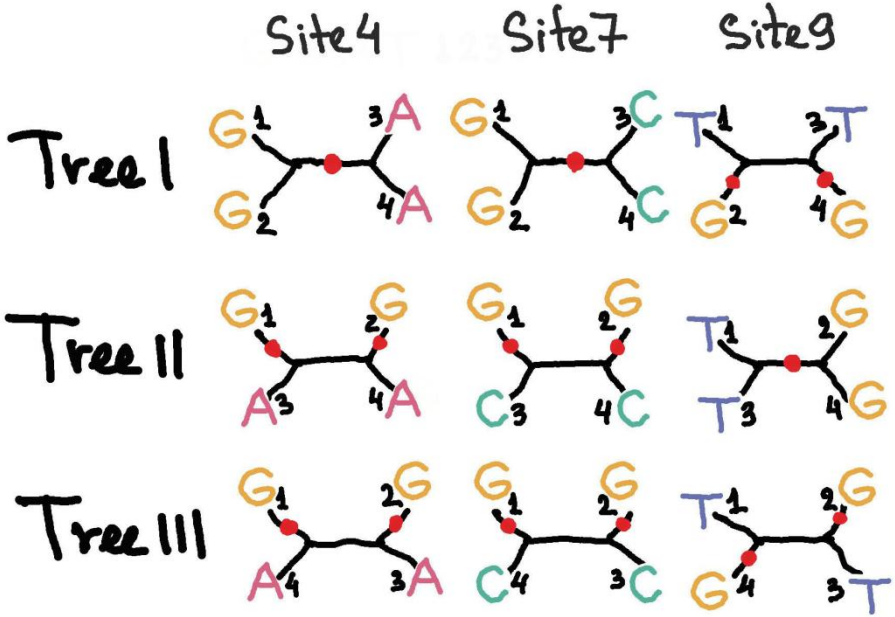
Все методы, кроме максимальной экономии, допускают предположение о молекулярных часах (но чаще используются без этого предположения!) и оценивают длины ветвей.

Методы MP и ML — символьно-ориентированные, LS, FM, ME и многие другие принимают на вход матрицу расстояний.

Метод максимальной экономии

```
Seq1 ATTGTCGTT
Seq2 ATTGTCGTG
Seq3 ATTATCCTT
Seq4 ATTATCCTG
```

	Поз.4	Поз.7	Поз.9	Сумма
Tree I	1	1	2	4
Tree II	2	2	1	5
Tree III	2	2	2	6



По дереву I получается наименьшее число замен

Метод максимального правдоподобия

Оцениваем причину по последствиям!

Нас интересует самое вероятное дерево (топология + длины ветвей), которое можно получить из наших данных: **P (дерево | выравнивание, модель).**
Но мы не знаем, как его искать!
Давайте для каждого дерева оценим обратную вероятность — правдоподобие (насколько вероятно из такого дерева получить наше выравнивание):
P (выравнивание | дерево, модель) = правдоподобие дерева

Это и будет критерием сравнения деревьев — дерево с бóльшим правдоподобием будем считать лучшим

Метод максимального правдоподобия

P (выравнивание | дерево, модель):
дерево = топология + длины ветвей

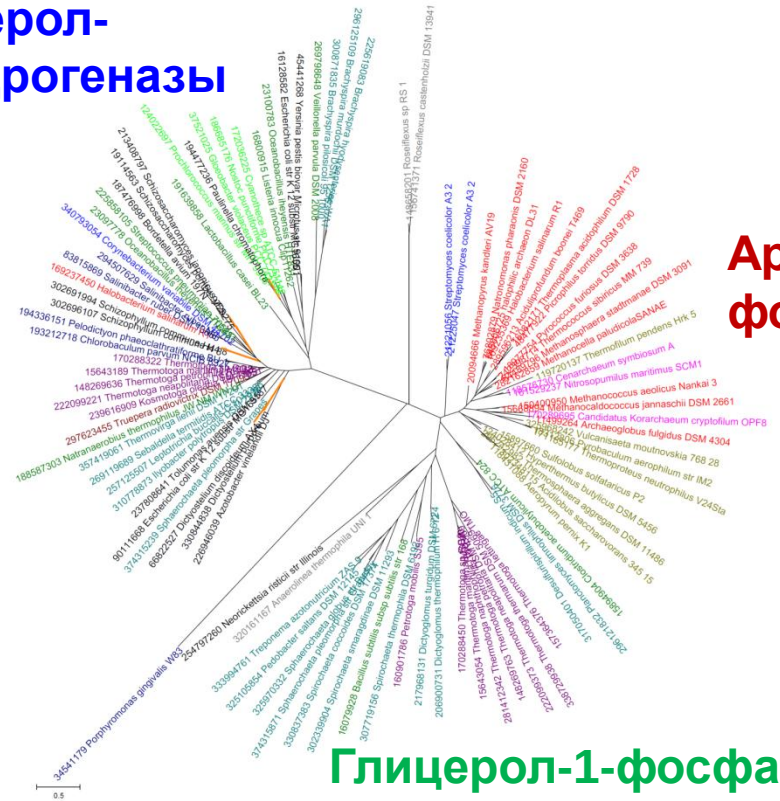
Зная длину ветви и задав модель эволюции, можно посчитать вероятность изменения нуклеотида/а.к. в каждом сайте на каждой ветви дерева

Предполагая независимость эволюции во всех сайтах, можно подсчитать вероятность того, что такое дерево породило такое выравнивание *(как?)*

УКОРЕНЕНИЕ

Пример

Глицерол-дегидрогеназы

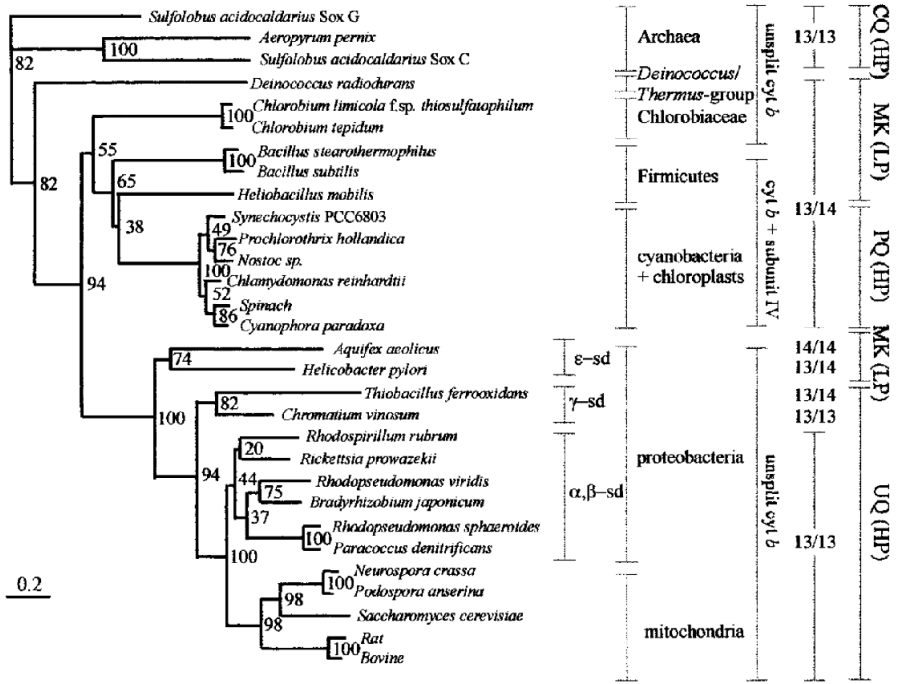


От положения корня зависит, как видится эволюция фермента!

Архейные глицерол-1-фосфат дегидрогеназы

Глицерол-1-фосфат-дегидрогеназы

Ещё пример

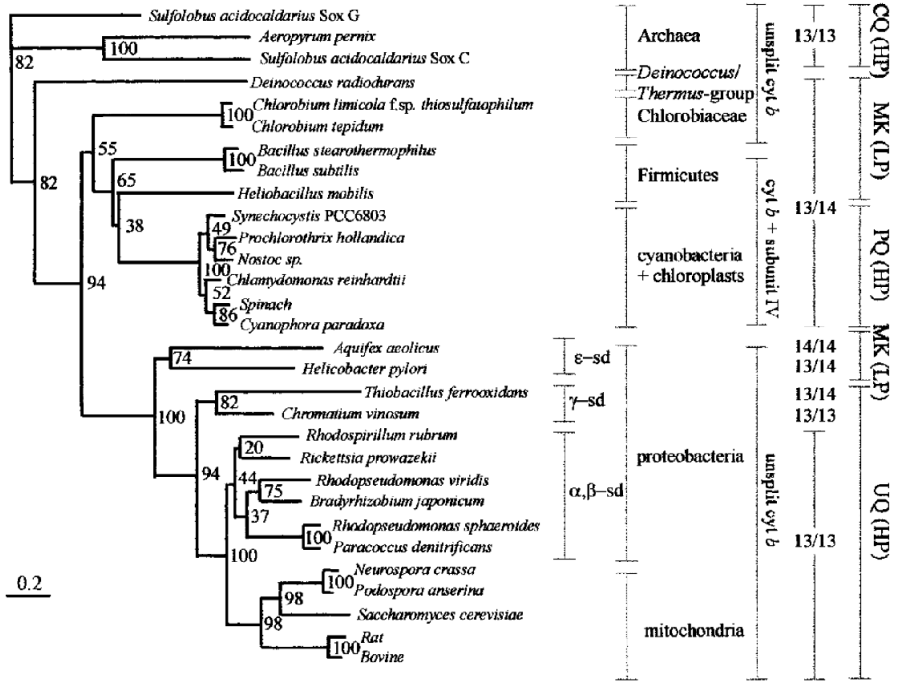


(Schütz *et al.*, 2000)

Для подтверждения своей гипотезы о древности некоторого фермента авторы работы укоренили дерево так, чтобы корневая ветвь разделяла археи и бактерии.

Однако такое укоренение ничем не обосновано!

Ещё пример



(Schütz *et al.*, 2000)

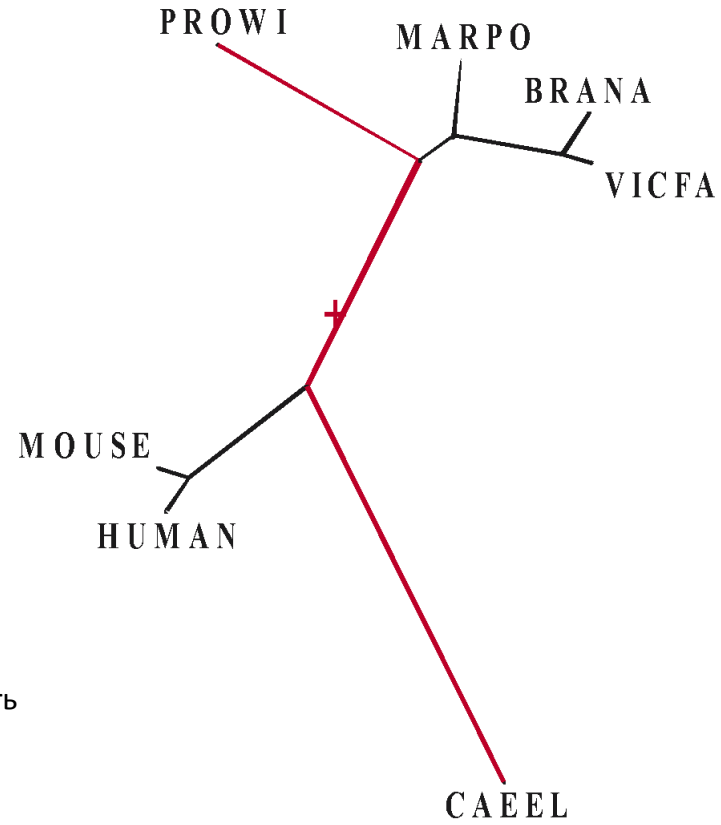
Для подтверждения своей гипотезы о древности некоторого фермента авторы работы укоренили дерево так, чтобы корневая ветвь разделяла археи и бактерии.

Однако такое укоренение ничем не обосновано!

Не менее вероятная альтернативная гипотеза: корень находится в другом месте, а три архейные последовательности – результат горизонтального переноса от предков эубактерии *Deinococcus* к предкам архей *Sulfolobus* и *Aeropyrum*.

Укоренение в среднюю точку

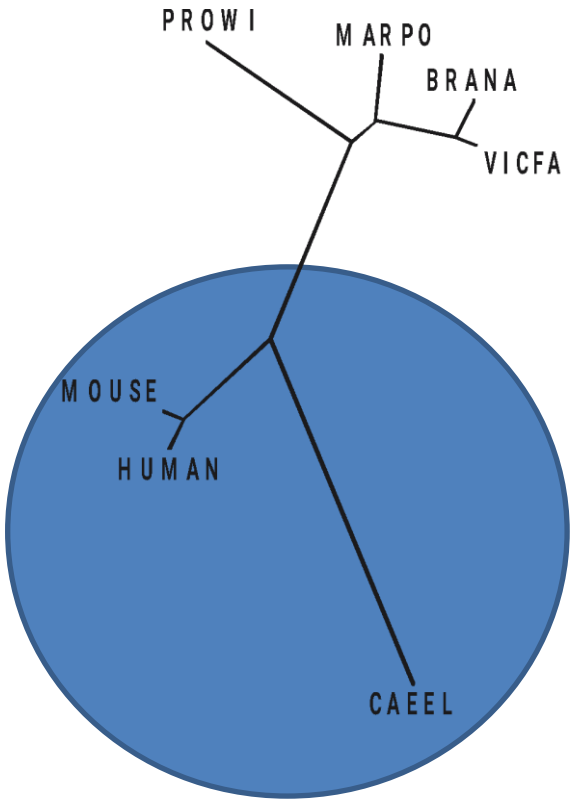
Находим на дереве самый длинный путь от листа к листу, ставим корень в середину пути



Этот способ предполагает хотя бы приблизительно точные молекулярные часы.

Но в любом случае на дерево, укоренённое в среднюю точку, смотреть легче, чем на укоренённое в случайно выбранное место (что обычно выдают программы реконструкции)

Укоренение с использованием внешней группы



В данном случае укоренено дерево четырёх растений, для чего пришлось построить дерево с участием внешней группы — трёх животных (в синем круге)

Подбор внешней группы – непростое дело. Нужно, чтобы все последовательности внешней группы достоверно являлись внешними (имели более раннего общего предка с нашими, чем общий предок наших); при этом они не должны быть слишком далёкими, иначе топология дерева (а тем самым и укоренение) будет недостоверным.

СРАВНЕНИЕ ДЕРЕВЬЕВ

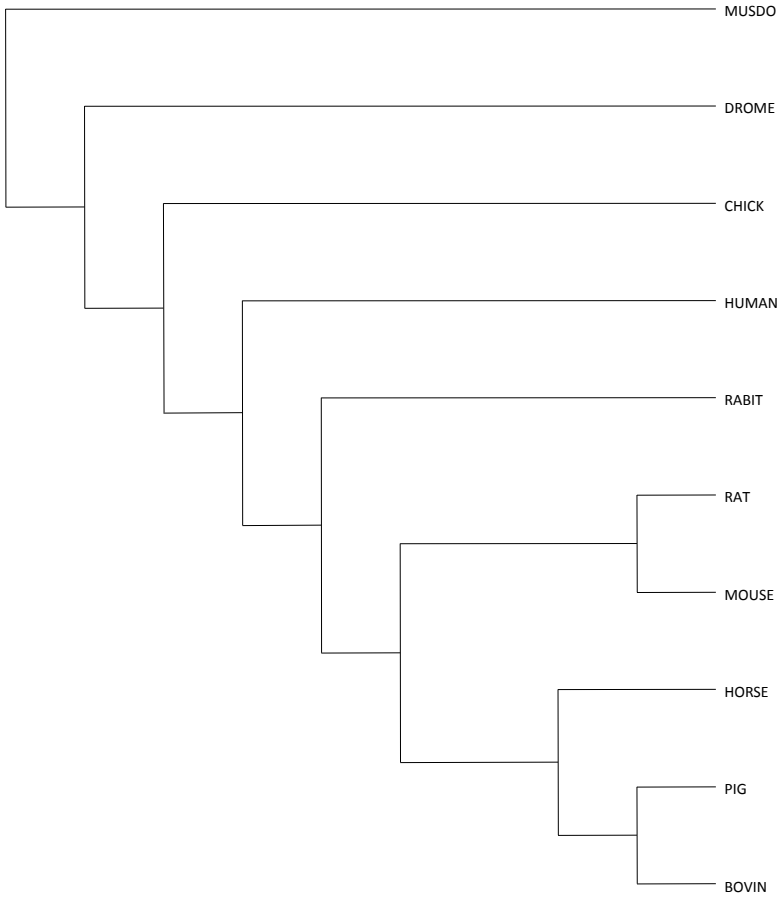
Сравнение деревьев

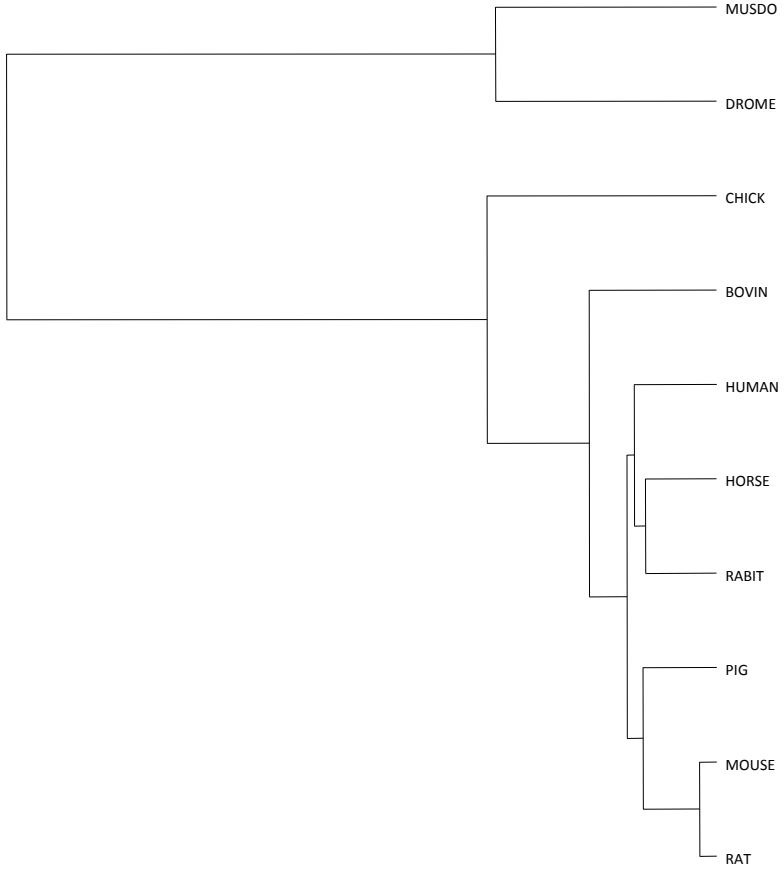
Программы реконструкции филогении так же ненадёжны, как и любые другие компьютерные программы предсказания биологических фактов.

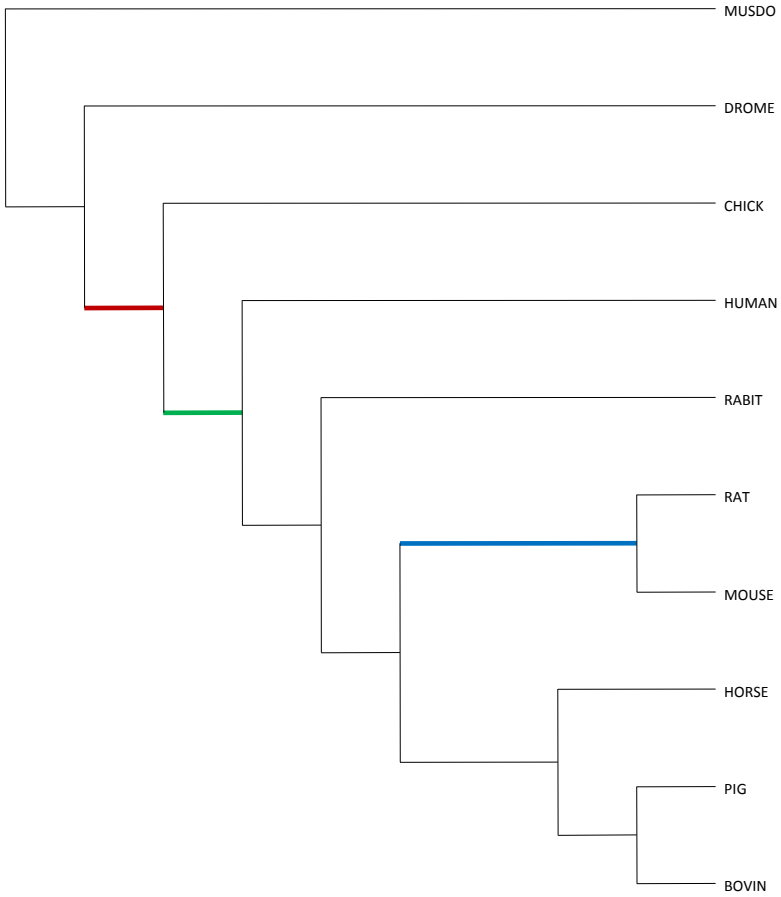
Поэтому (в частности) возможны различные варианты реконструкции по одним и тем же данным. Аналогичная ситуация: реконструируем филогению организмов по разным группам ортологов.

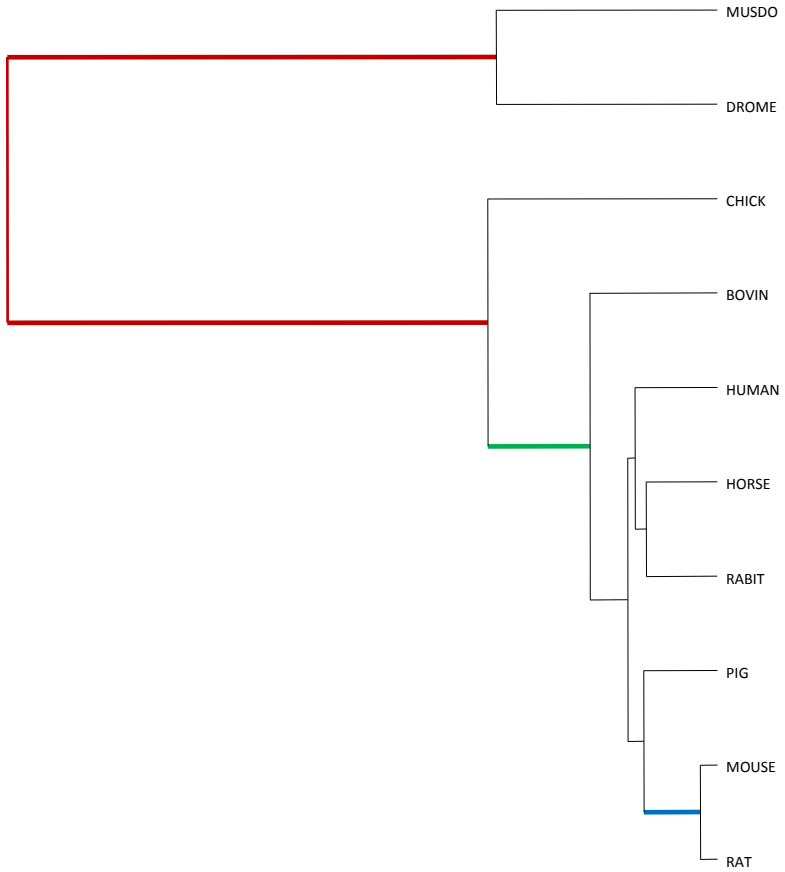
Встаёт задача сравнения различных деревьев с одним и тем же набором листьев.

Что общего у этих двух деревьев?

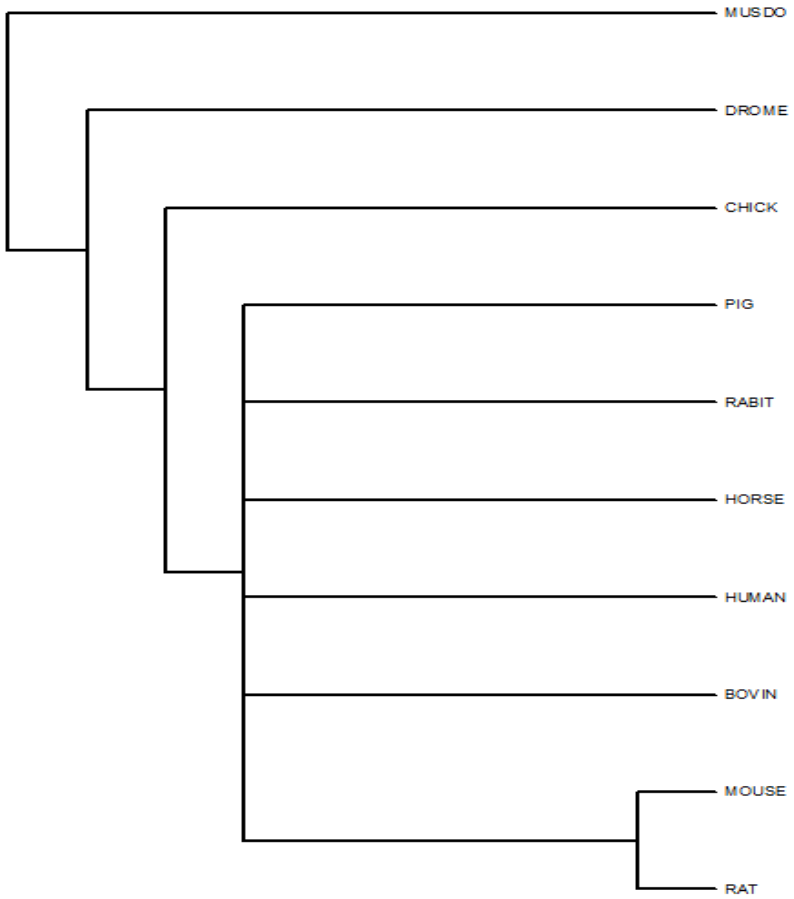





MUSDO
DROME
CHICK
HUMAN
RABIT
RAT
MOUSE
HORSE
PIG
BOVIN


MUSDO
DROME
CHICK
BOVIN
HUMAN
HORSE
RABIT
PIG
MOUSE
RAT

Консенсусное дерево



Совмещение данных от многих деревьев

По набору деревьев с одинаковым множеством листьев:

1. Консенсус (consensus)
   Включает только те ветви, которые встретились **во всех** деревьях исходного набора
2. Дерево большинства (majority-rule tree)
   Включает только те ветви, которые встретились **в большинстве** деревьев исходного набора
3. Дерево расширенного большинства (extended majority-rule tree)
   К дереву большинства добавляются ветви, не противоречащие уже имеющимся, начиная с наиболее «поддержанных»

Какие ветви не противоречат друг другу?

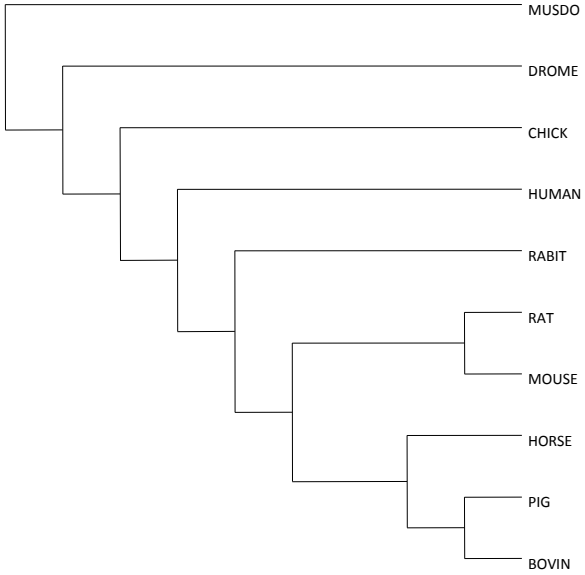
Не любые два разбиения множества листьев могут соответствовать двум ветвям одного и того же дерева.

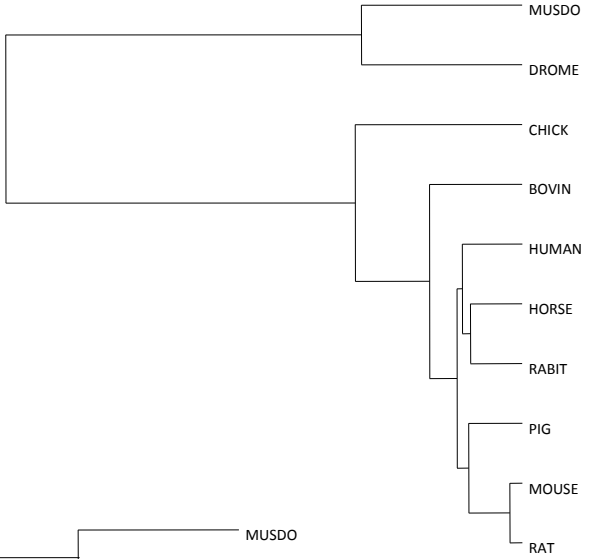
Рассмотрим четыре пересечения двух частей одного разбиения с двумя частями другого. Два разбиения совместимы (могут быть ветвями на одном дереве), только если из этих четырёх пересечений одно пусто.

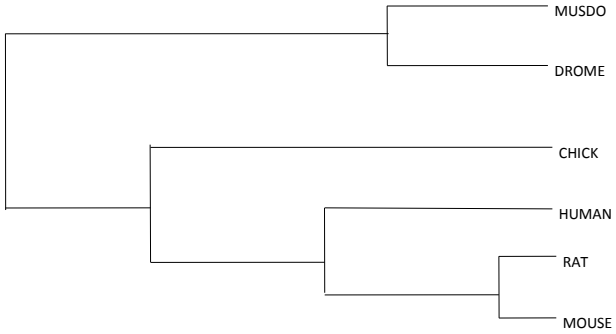
Эквивалентная формулировка: одна из частей одного разбиения должна лежать целиком внутри одной из частей другого разбиения.

Другой вариант консенсуса

Наибольшее общее поддерево







Н.О.П. определено не однозначно!

Как оценить достоверность отдельных ветвей дерева?

БУТСТРЭП

Bootstrap

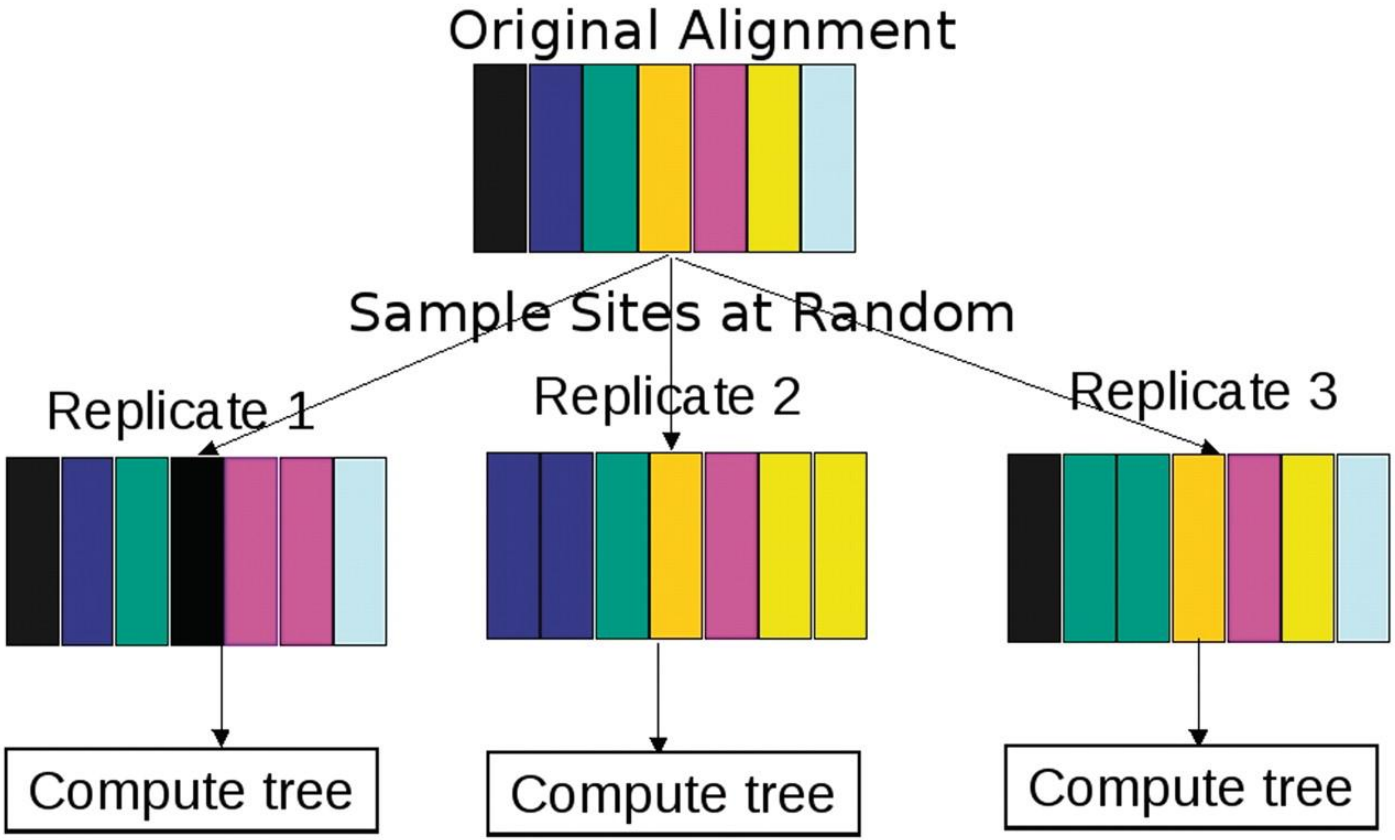
Бутстрэп-анализ

Из входного выравнивания делается много (например, 100) так называемых «бутстрэп-реплик».

Каждая бутстрэп-реплика получается в результате случайного выбора столбцов из выравнивания в том же количестве, что в исходном выравнивании. В результате некоторые столбцы выбираются один раз, некоторые дважды, некоторые ни разу.

Смысл в том, чтобы построить дерево по части данных и затем сравнить результаты от по разному выбранных частей.

Outline of the phylogenetic bootstrap procedure

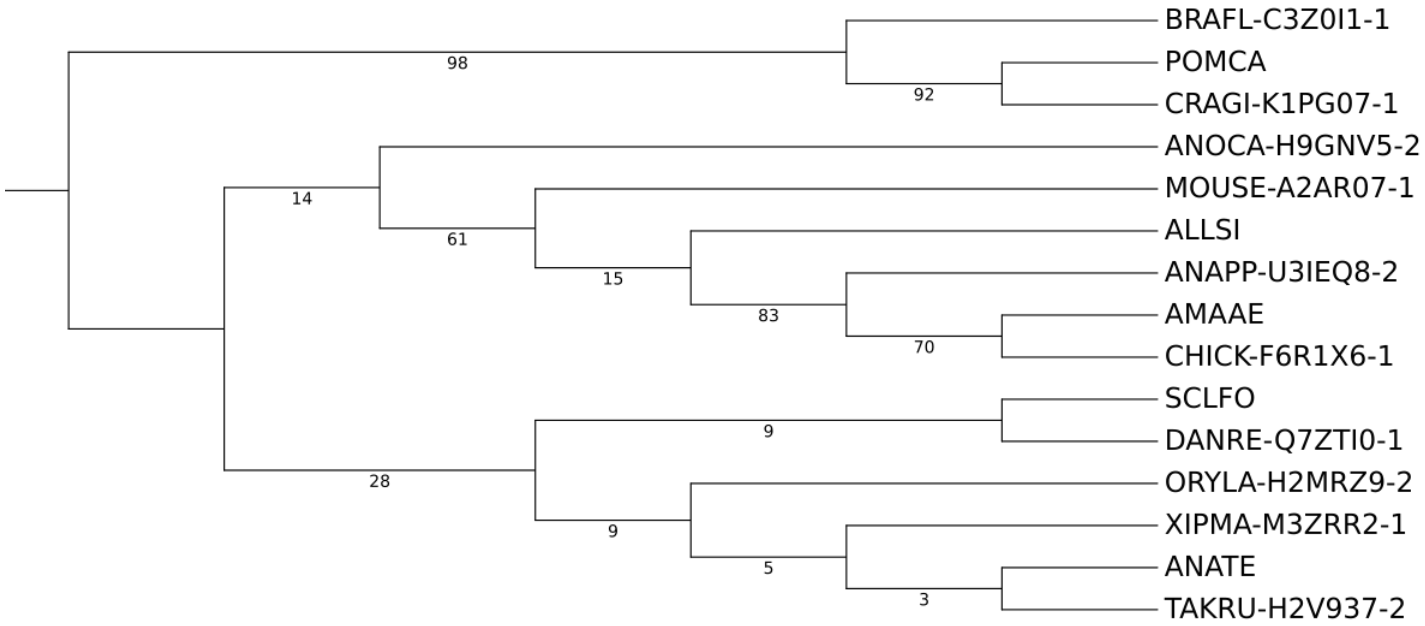


Stamatakis, Izquierdo-Carrasco. Brief Bioinform 2011;12:270-279

Бутстрэп-анализ

- Создаём из входного выравнивания 100 бутстрэп-реплик
- Для каждой из реплик строим по дереву
- Два варианта:
  – строим дерево как обычно, по входному выравниванию, но для каждой ветви отмечаем, в каком проценте деревьев, построенных по репликам, встретилась эта ветвь. По этим числам ветвях («бутстрэп-поддержке») судим о достоверности каждой ветви
  – из 100 деревьев строим дерево по методу расширенного большинства («Extended majority-rule tree»)

Бутстреп-анализ (пример результата)



Особенности работы с нуклеотидными последовательностями

- Убедитесь, что все последовательности гомологичны **друг другу**, а не своим комплементарным вариантам!

  Вообще всегда стоит посмотреть на выравнивание, прежде чем начинать строить по нему дерево

- Помните, что участки ДНК, представленные в базе данных, могут иметь довольно произвольные границы — удаляйте негомологичные концы

  Или используйте опцию –r в fastme и аналогичные: игнорирование колонок выравнивания, в которых есть гэпы

- Если ДНК кодирующая, то подумайте, не лучше ли строить дерево по соответствующим аминокислотным последовательностям?

  Если последовательности далёкие, то точно лучше: больше информации. Если близкие, то у белков может быть слишком мало неконсервативных позиций, тогда наоборот, в генах информации больше, за счёт вырожденности кода

  Если всё же по генам, то стоит делать выравнивание генов по выравниванию белков, будет меньше ошибок выравнивания

  Например, используйте PAL2NAL: https://www.bork.embl.de/pal2nal/