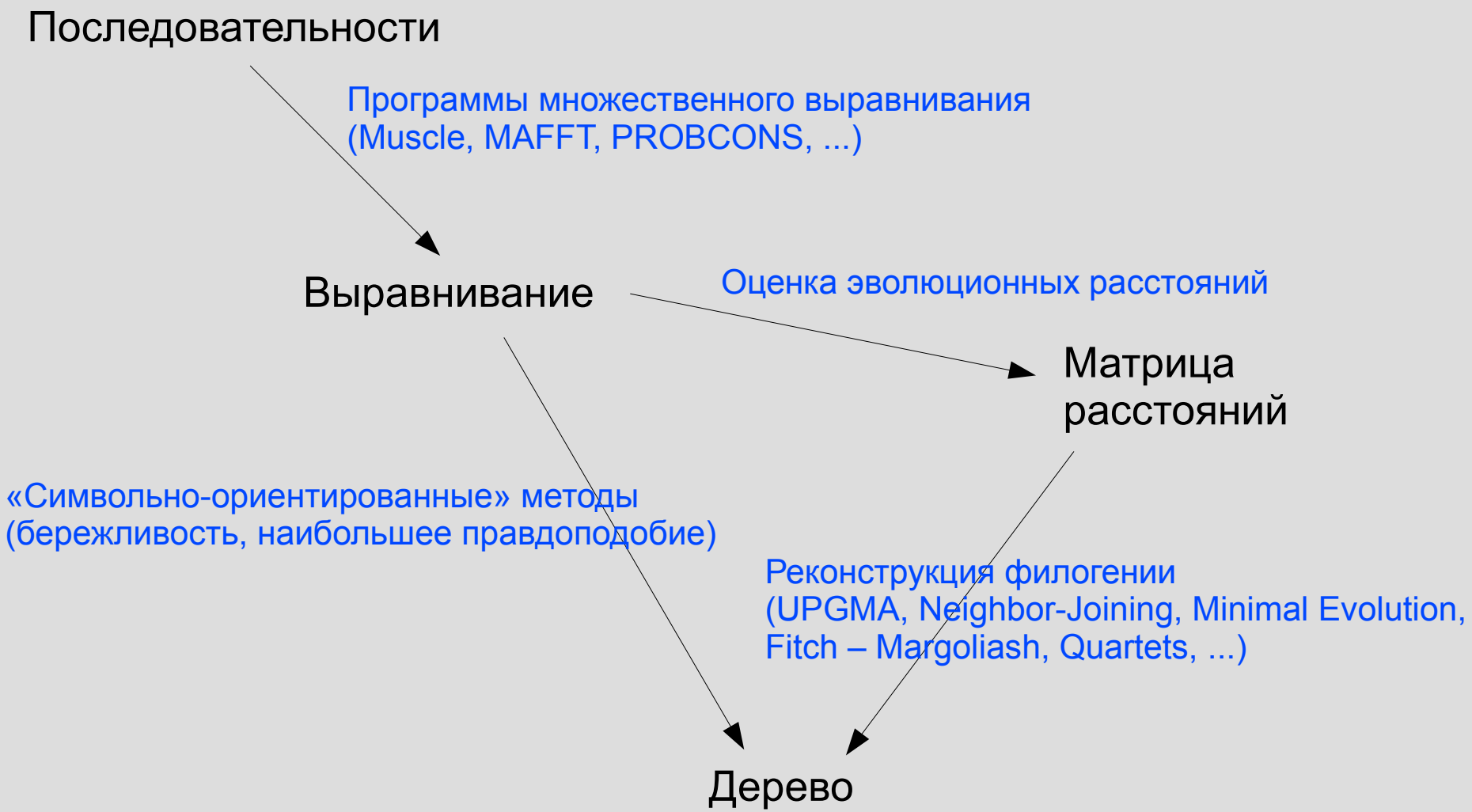
Алгоритмы реконструкции филогенетических деревьев

С.А.Спирин

2 марта 2010

ФББ МГУ

Схема реконструкции филогении по последовательностям



Классификация методов

Название метода	Переборный / эвристический	Предполагает молекулярные часы	Символьно ориентированный
UPGMA	Эвристический	Да	Нет
Neighbor-Joining	Эвристический	Нет	Нет
Наименьших квадратов	Переборный	Может	Нет
Фитча – Марголиаша	Переборный	Может	Нет
Минимальной эволюции	Переборный	Нет	Нет
Максимальной бережливости	Переборный	Нет	Да
Наибольшего правдоподобия	Переборный	Может	Да

Методы, предполагающие молекулярные часы, строят укоренённые ультраметрические деревья.

Методы, не предполагающие молекулярные часы строят, как правило, неукоренённые деревья.

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 square)

Пусть дана матрица расстояний и топология дерева;
i, j — две последовательности, тогда мы имеем расстояние $d(i,j)$ из матрицы, и, приписав ветвям длину, будем иметь расстояние $d'(i,j)$ «по дереву».
Подберём длины ветвей так, чтобы сумма $\left(d(i,j) - d'(i,j)\right)^2$ (по всем парам 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 (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

Все деревья перебрать (как правило) нельзя!

Число различных деревьев с N листьями равно:

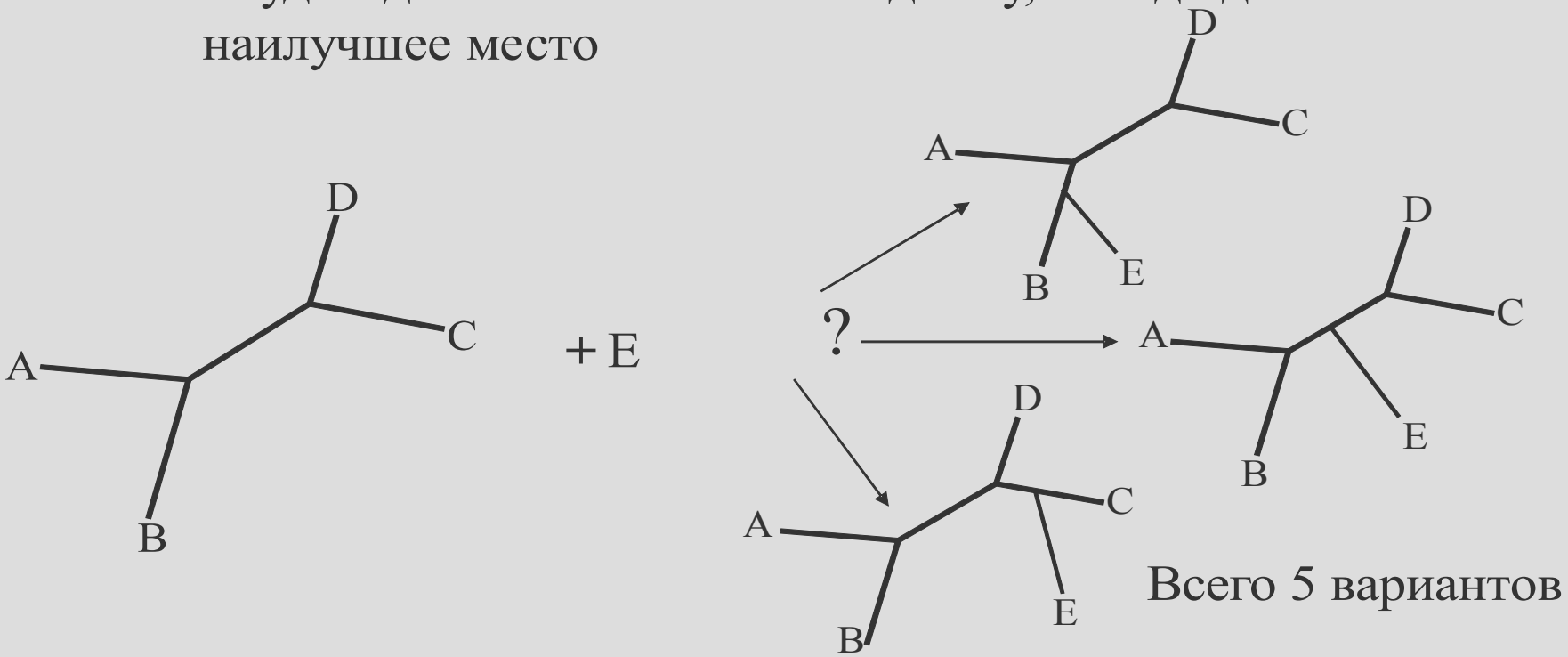
$$(2N-5)!! = 1 \cdot 3 \cdot 5 \cdot \ldots \cdot (2N-5)$$

Это число очень быстро растёт!

Полный перебор возможен, если число последовательностей не превышает 10–12

Поиск лучшего дерева: «выращивание»

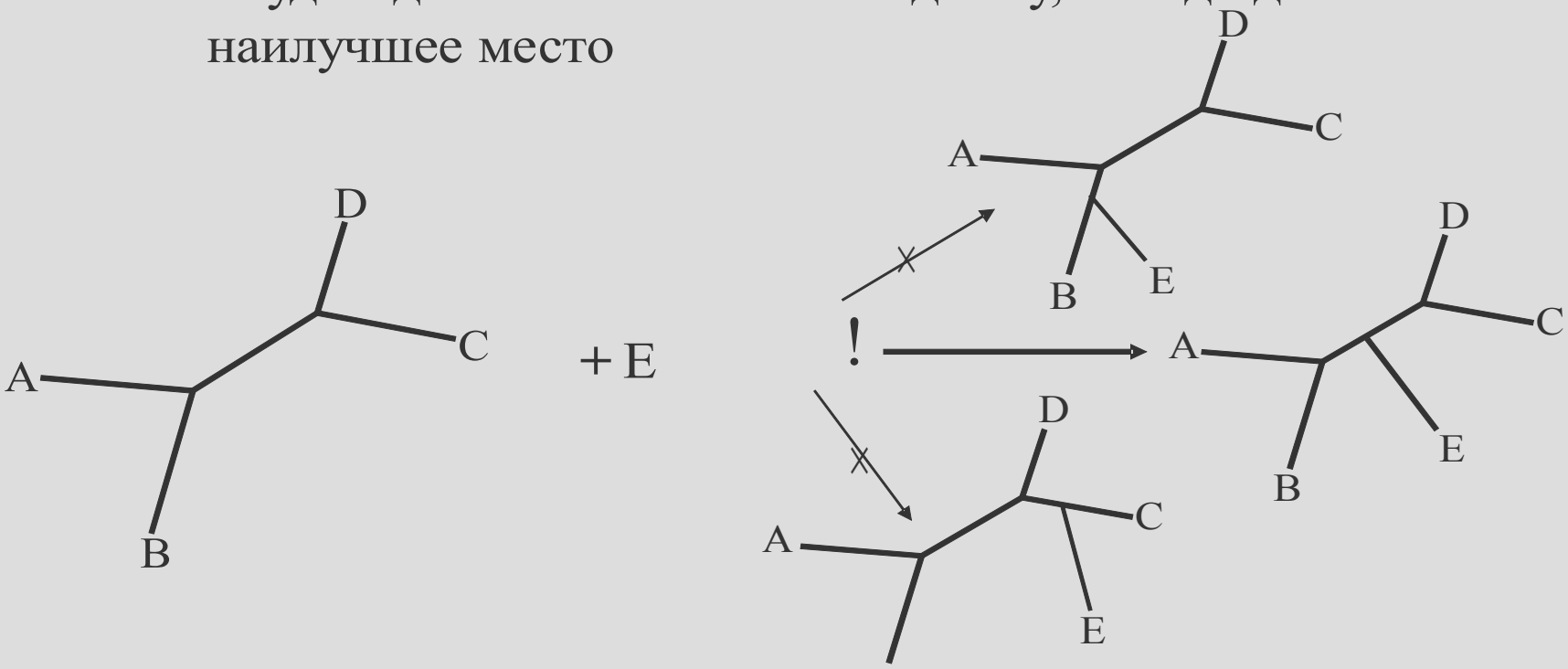
- Найдем лучшее дерево для части последовательностей
- Будем добавлять листья по одному, находя для них наилучшее место



Всего 5 вариантов

Поиск лучшего дерева: «выращивание»

- Найдем лучшее дерево для части последовательностей
- Будем добавлять листья по одному, находя для них наилучшее место



Поиск лучшего дерева: «выращивание»

Дерево с N листьями всегда имеет $2N–3$ ветви.
Поэтому, чтобы “вырастить” дерево с N листьями, надо проанализировать
$3 + 5 +...+ (2N – 5) = (N – 3)(N – 1)$ деревьев.
Уже для N=10 это число меньше числа всех возможных деревьев в 32175 раз!

Выращивание не гарантирует нахождение “лучшего” дерева, но при хороших данных не должно приводить к большим ошибкам.

Поиск лучшего дерева: просмотр соседних деревьев

Построим сначала «черновое» дерево, а затем попробуем его улучшить.

Черновое дерево можно построить одним из эвристических методов или «вырастить».

Улучшать будем, просматривая «соседние» деревья.

Поиск лучшего дерева: просмотр соседних деревьев

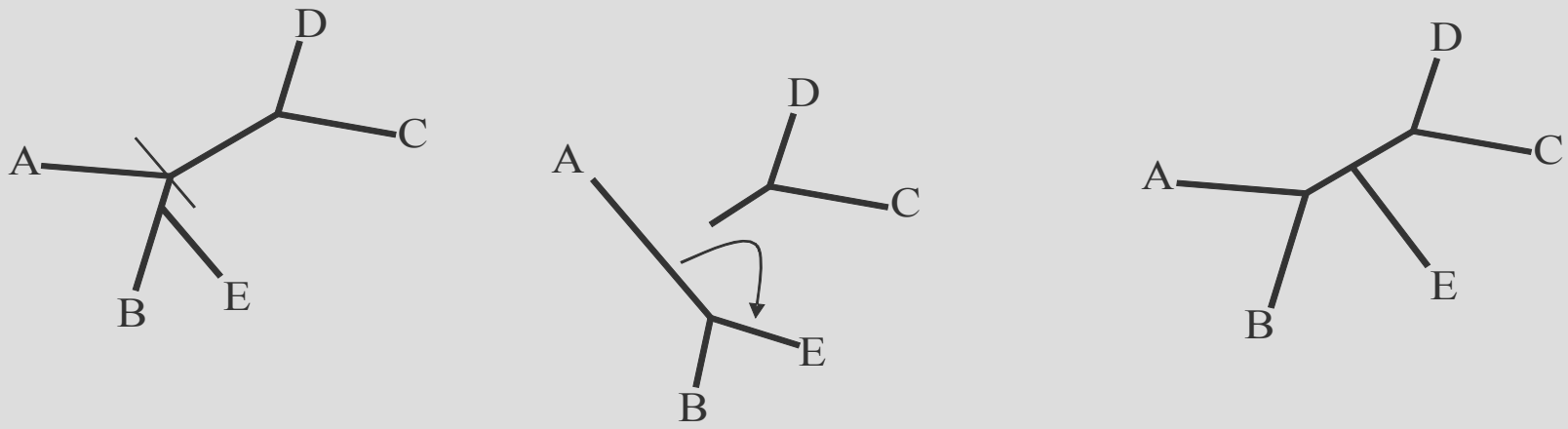
Что такое «соседние» деревья

- 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

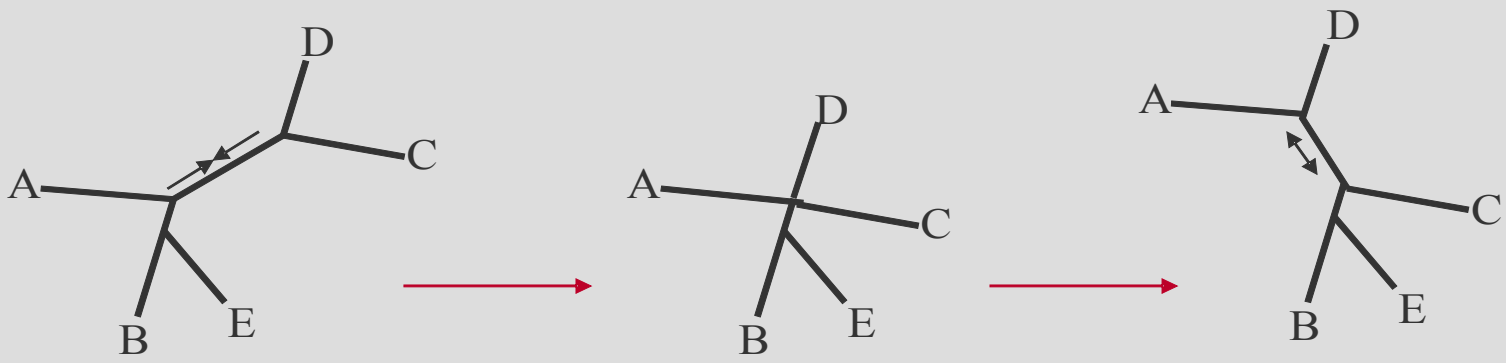


В пакете PHYLIP это называется “Global rearrangement”

Поиск лучшего дерева: просмотр соседних деревьев

Что такое «соседние» деревья

- Можно «схлопнуть» одну ветвь и заменить её другой



В пакете PHYLIP это называется “Local rearrangement”

Поиск лучшего дерева

Алгоритм поиска

- Строим черновое дерево
  (два варианта: эвристический метод или «выращивание» с использованием критерия качества).
- Анализируем соседние деревья;
  если находим среди соседей лучшее, берём за основу его.
- Повторяем предыдущий пункт, пока текущее дерево не окажется лучше всех своих соседей.

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 (на практике сводится к поиску «достаточно качественного» дерева).

Как правило, название метода совпадает с названием критерия.

Переборные методы

- Максимальной бережливости (maximum parsimony, MP)
- Наибольшего правдоподобия (maximal likelihood, ML)
- Наименьших квадратов (least squares, LS)
- Фитча – Марголиаша (Fitch – Margoliash, FM)

Все методы, кроме бережливости, допускают предположение о молекулярных часах (но чаще используются без этого предположения!).

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

Эвристические методы

- UPGMA = «Unweighted pair group method with arithmetic mean»
  Строит укоренённое ультраметрическое дерево
  Видимо, реально лучший из методов, предполагающих молекулярные часы.

- Neighbor-Joining
  Строит неукоренённое дерево. Если и уступает некоторым переборным алгоритмам, то не сильно.

Оба метода принимают на вход матрицу расстояний.

UPGMA – схема алгоритма

Укоренённое дерево строится «снизу вверх»

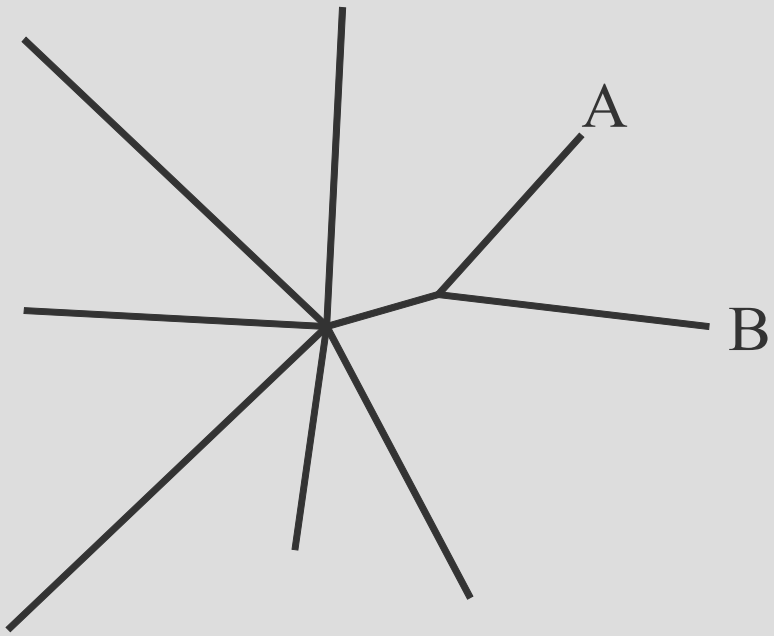
- Найдём в матрице расстояний наименьший элемент.
- Объединим два ближайших листа в кластер (это – узел дерева, соединённый ветвями с листьями, образовавшими его).
- Пересчитаем матрицу расстояний, рассматривая кластер как новый лист. Расстоянием до кластера будем считать **среднее арифметическое** расстояний до его элементов (отсюда название метода).
- Повторяем с начала, пока не останется всего два кластера.

К этому прибавляется способ вычисления длин ветвей.

Результат — укоренённое ультраметрическое дерево с длинами ветвей.

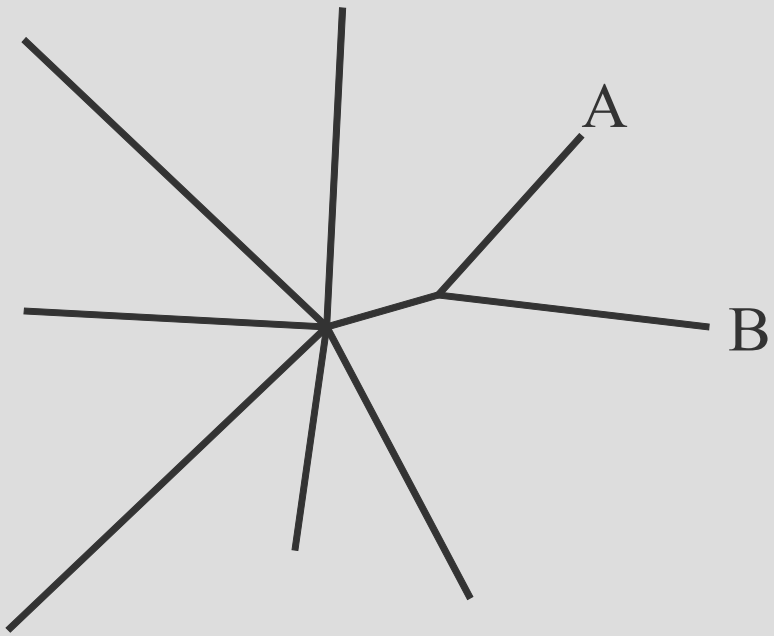
Neighbor-joining

- Выбираем пару последовательностей *A,B* , для которых наименьшее значение имеет величина $d(A,B) - m(A) - m(B)$, где d — расстояние из матрицы, а m — среднее расстояние до всех остальных последовательностей. Объединяем пару в кластер.



Neighbor-joining (продолжение)

- Повторяем объединение, пока не останется три кластера.



В отличие от UPGMA, даже при ультраметрической матрице «соседи» не обязательно объединяются снизу вверх! Полученное методом Neighbor-joining дерево — неукоренённое!

Сравнение методов
(результат одного из исследований)

	fitch	NJ	UPGMA	kitsch	protpars	proml	promlk		+	0	–
fitch		46:26	**54:26**	**62:16**	**90:4**	**88:8**	**94:5**		5	1	0
NJ	26:46		**52:31**	**49:26**	**85:7**	**82:9**	**94:4**		5	1	0
UPGMA	26:54	31:52		38:17	**73:19**	**78:19**	**88:7**		3	1	2
kitsch	16:62	26:49	17:38		**70:20**	**75:21**	**91:7**		3	1	2
protpars	4:90	7:85	19:73	20:70		42:39	**65:27**		1	1	4
proml	8:88	9:82	19:78	21:75	39:42		53:38		0	2	4
promlk	5:94	4:94	7:88	7:91	27:65	38:53			0	1	5

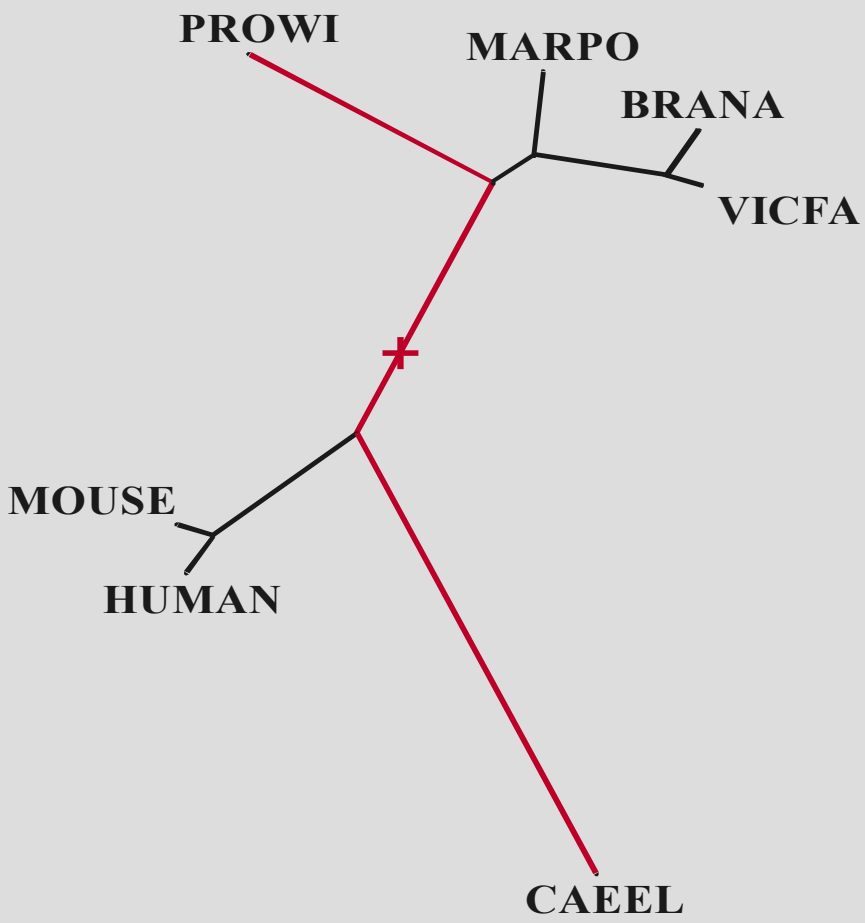
fitch — метод Фитча – Марголиаша без молекулярных часов
kitsch — метод Фитча – Марголиаша с молекулярными часами
NJ — метод Neighbor-joining
protpars — метод максимальной бережливости
proml — метод наибольшего правдоподобия без молекулярных часов
promlk — метод наибольшего правдоподобия с молекулярными часами

Счёт выделен **жирным**, если вероятность получить такой же или больший по случайным причинам — менее одной сотой.

Укоренение

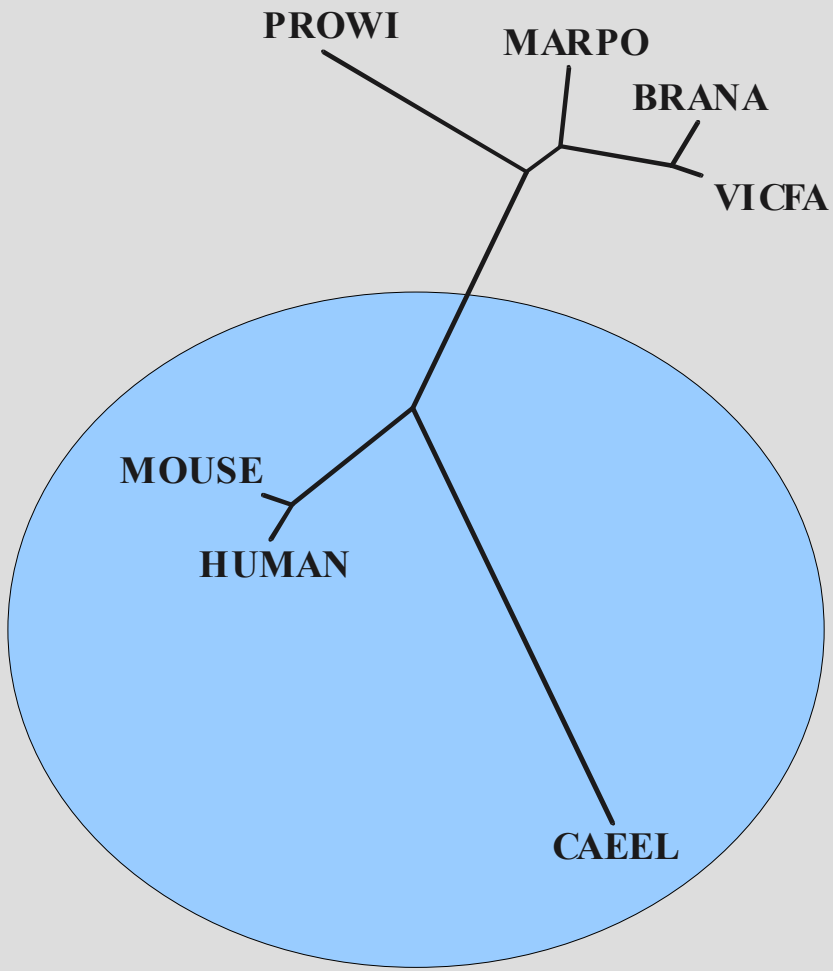
- В среднюю точку:

Находим на дереве самый длинный путь от листа к листу и за корень принимаем середину этого пути



Укоренение

- Используя «аутгруппу» (outgroup):



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

Сравнение деревьев

- Консенсусное (небинарное) дерево
- Максимальное общее поддерево
- Дерево из ветвей, поддержанных большинством (majority-rule tree)
- Меры сходства деревьев ("расстояние")

  i. Доля общих ветвей
  ii. Расстояние в "пространстве ветвей"
  iii. Доля общих четверок
  iv. Длина пути в пространстве деревьев

Бутстрэп-анализ

- создаём из входного выравнивания 100 «бутстрэп-реплик»;
- для каждой из реплик строим по дереву;
- из 100 деревьев строим дерево по методу большинства («majority-rule tree»).

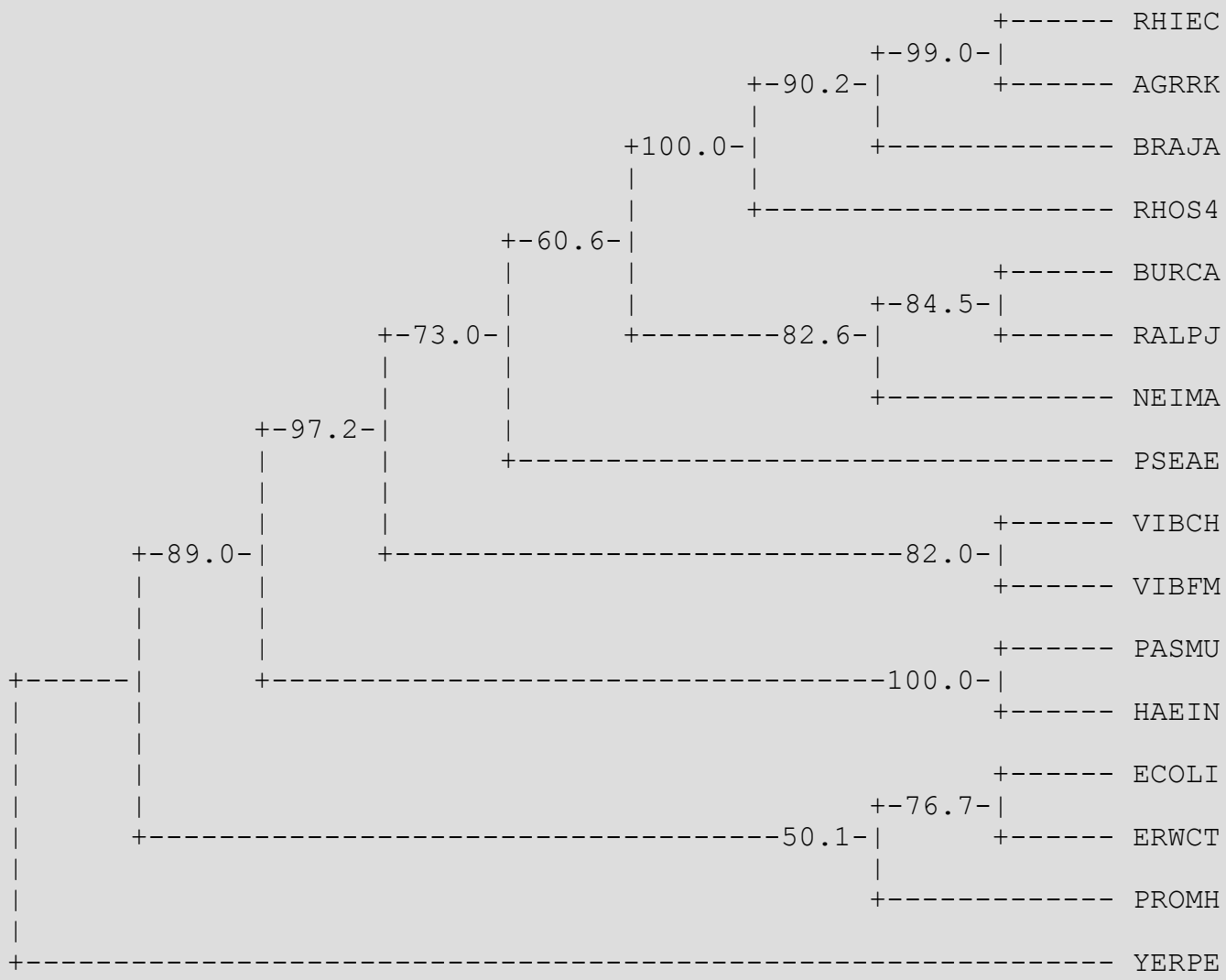
Помимо того, что (как правило) возрастает качество реконструкции, есть возможность оценить достоверность каждой ветви по т.н. «бутстрэп-поддержке», то есть проценту деревьев, в которых встретилась данная ветвь.

Бутстрэп-анализ

Каждая бутстрэп-реплика получается в результате случайного удаления половины столбцов из выравнивания с заменой их копиями других (тоже случайно выбранных) столбцов.

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

Бутстрэп-анализ
(пример результата)



Пакет PHYLIP

- Реализация методов UPGMA и Neighbor-Joining (программа *neighbor*), наименьших квадратов и Фитча – Марголиаша (*fitch* и *kitsch*), максимальной бережливости (*dnapars* и *protpars*), наибольшего правдоподобия (*dnaml*, *dnamlk*, *proml*, *promlk*)
- Оценка эволюционных расстояний: программы *dnadist* и *protdist*
- Сравнение деревьев: *consense, treedist, treedistpair*
- Редактура (включая укоренение в среднюю точку): *retree*
- Бутстрэп: *seqboot*
- Визуализация: *drawtree, drawgram*

Пакет PHYLIP

- Свободно распространяется, имеются версии для всех основных операционных систем (доступен для скачивания на сайте http://evolution.genetics.washington.edu/phylip.html)
- Имеется удобный веб-интерфейс: http://bioweb.pasteur.fr/phylogeny/intro-en.html
- В пакет EMBOSS в качестве дополнения включены варианты всех программ пакета PHYLIP, снабженные интерфейсом в стиле EMBOSS (отличаются буквой *f* в начале, например fprotpars вместо protpars)