Лекция 6. Программа BLAST (26 марта).

1.1 Напоминание лекции 3.

1.1.1 Выравнивание биологических последовательностей.

Определение. Выравнивание — установление соответствий между буквами разных последовательностей. Обычно требуется, чтобы при сопоставлении сохранялся порядок букв.

В биоинформатике смысл выравнивания заключается в сопоставлении гомологичных аминокислотных остатков или нуклеотидов.

Такое выравнивание записывается в виде таблицы, строки которой содержат буквы исходных последовательностей. Гэпы (прочерки в последовательности) показывают, что для данного аминокислотного остатка нет гомологичного в другом белке. Биологическая причина — инсерции и делеции, закрепившиеся в эволюции.

Выделяют программы локального и глобального выравнивания. Программа глобального выравнивания только расставляет гэпы, выравнивая последовательности по всей длине. Программа локального выравнивания выбирает в каждой последовательности по участку, а затем выравнивает между собой эти участки.

Глобальное выравнивание используется, когда есть уверенность в том, что последовательности гомологичны. Если последовательности содержат только гомологичные участки или если про последовательности неизвестно ничего, то используется локальное выравнивание.

1.1.2 Примеры алгоритмов выравнивания:

- парное глобальное выравнивание (global alignment of two sequences) — алгоритм Нидлмана–Вунша (Needleman&Wunsch); вот ссылка на визуализацию работы алгоритма,
- парное локальное выравнивание (local alignment of two sequences) — алгоритм Смита–Уотермена (Smith&Waterman),
- множественное выравнивание (multiple alignment), когда последовательностей больше двух, — ClustalW, Muscle, MAFFT, T-Coffee и пр.

Программы выравнивания действуют формально и выдадут выравнивание, даже если на вход им подать негомологичные белки. Смысла такое выравнивание иметь не будет.

Даже если белки гомологичны, в выравнивании, выданном программой, не обязательно будут сопоставлены гомологичные аминокислотные остатки или нуклеотиды. Алгоритмы выравнивания сделаны так, чтобы максимизировать долю правильных сопоставлений. Но невозможно придумать такой алгоритм, который бы выдавал биологически правильный ответ всегда. Выравнивание, выданное программой, — это всего лишь реконструкция причины (эволюции) по последствиям (т.е. современным последовательностям).

Выравнивание необходимо, чтобы

- предсказать функцию белка на основании установленной гомологии с экспериментально изученными белками,
- выяснить, какие аминокислотные остатки консервативны (следовательно, важны для функции белка),
- оценить время расхождения последовательностей, а если последовательностей много, то реконструировать их эволюционную историю.

1.2 Постановка задачи.

Дана последовательность белка (query) и база данных (например, UniProt). Надо найти белки, гомологичные заданной последовательности (полностью или частично). Критерий завершения — последовательности должны иметь статистически значимое сходство (не случайное). Критерием качества парного выравнивания является его вес (score). Основные подходы к решению:

- локальное выравнивание: используется для поиска участков сходства. Алгоритм Смита–Уотермана точен, но медленен для больших баз данных,
- эвристический алгоритм BLAST: быстрый, но менее точный. Основан на поиске высокоскоростных совпадений коротких слов.

1.3 BLAST.

Определение. BLAST[1] — программа поиска последовательностей, похожих на данную.

Данная программа использует локальное выравнивание исходной последовательности с последовательностями из банка и оценивает значимость совпадений через E-value (вероятность случайного совпадения).

Критерий гомологии:

- процент идентичности $>$ 20–25% на длинном участке,
- низкий E-value (например, < 0.001) указывает на статистическую значимость.

1.3.1 Принцип работы BLAST:

- индексация базы данных: создается таблица всех слов длины W[2] (по умолчанию $W = 3$) с их позициями в последовательностях,
- поиск похожих слов: для входной последовательности генерируются все слова длины W. Добавляются слова, похожие на них (с заданным порогом на вес T[3], например, $T = 13$),
- отбор кандидатов: ищутся последовательности с двумя совпадениями на одной диагонали на расстоянии $\leq A$[4] (по умолчанию $A = 20$),
- расширение выравнивания: динамическое программирование применяется локально для расширения выравнивания в обе стороны. Остановка при снижении веса ниже порога X.

[1] **B**asic **L**ocal **A**lignment **S**earch **T**ool

[2] Увеличение W ускоряет поиск, но снижает чувствительность, уменьшение W повышает чувствительность, но замедляет работу.

[3] T определяет минимальный вес для учета совпадения.

[4] Максимальный разрыв между совпадениями на диагонали.

1.3.2 Достоинства и недостатки BLAST.

- Достоинства:
  - высокая скорость благодаря индексации и эвристикам,
  - возможность работы с огромными базами данных.
- Недостатки:
  - потеря чувствительности при больших W,
  - риск пропуска слабых, но биологически значимых совпадений.

1.3.3 Интерпретация результатов BLAST.

- E-value — вероятность случайного совпадения (чем меньше, тем значимее),
- вес в битах (Bit score) — нормализованный вес выравнивания (не зависит от матрицы замен),
- Identity/Positives — процент идентичности/сходства аминокислот,
- Query cover — процент покрытия запроса выравниванием.

1.4 Практические рекомендации.

Для большинства задач используйте локальное выравнивание (BLAST). Настраивайте параметры (W, T, A) в зависимости от цели. Преследуете высокую точность: уменьшайте W, снижайте T. Важен быстрый поиск: увеличивайте W, повышайте T. Интерпретируйте E-value в контексте биологической значимости.

Замечание. Всё предыдущее относилось к белковому BLAST'у (BLASTP), а есть ещё BLASTN и поиск с формальной трансляцией (BLASTX, TBLASTN и TBLASTX).