

Парное выравнивание

Алгоритмы в биоинформатике

Антон Елисеев

eliseevantoncoon@gmail.com

Что было на прошлой лекции?

- Транскрипция и трансляция.
- Свойство локальности ДНК.
- Можно считать расстояние между строками и делать выводы о свойствах организмов.
- Сравнивать участки генома можно достаточно эффективно.

Что будет на этой лекции?

- Определение выравнивания и веса выравнивания.
- Неравнозначные замены. Матрицы замен BLOSUM и PAM.
- Проблема гэпов. Определение аффинных штрафов за гэпы.

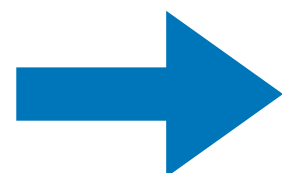
Расстояние и выравнивание

GATTACA

GATTACA

Расстояние и выравнивание

GATTACA GATTACA
GATT~~A~~CA GATTCA



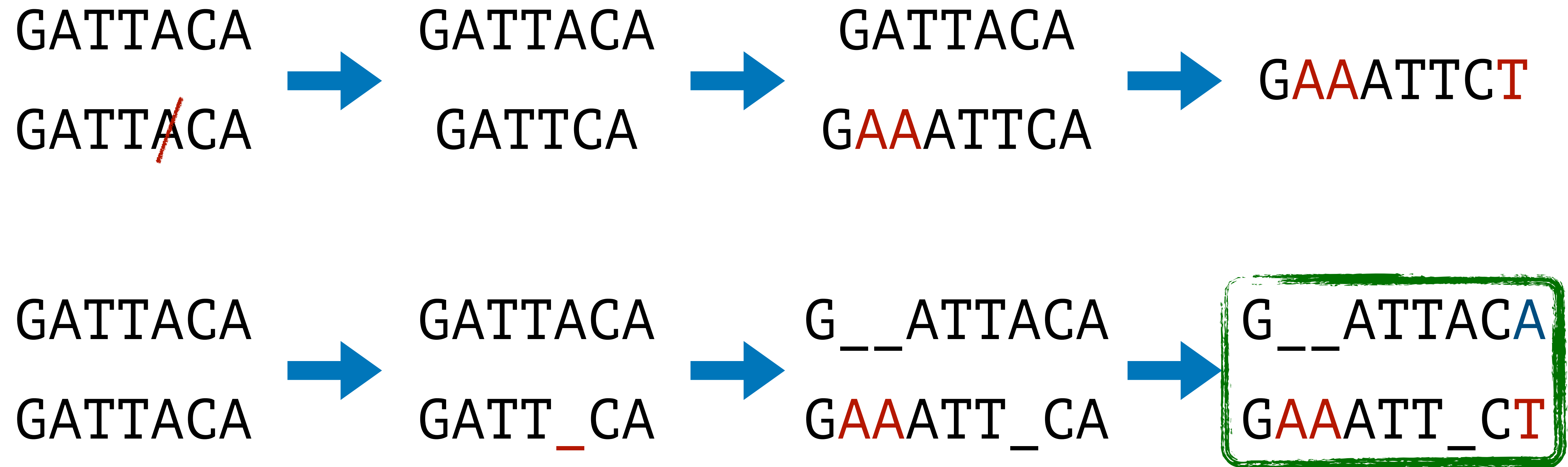
Расстояние и выравнивание

GATTACA → GATTACA → GATTACA
GATT~~A~~CA → GATTCA → GAATTC A

Расстояние и выравнивание

GATTACA → GATTACA → GATTACA → GAAATTCT
GATTACA GATTCA GAAATTCA

Расстояние и выравнивание



Выравнивание



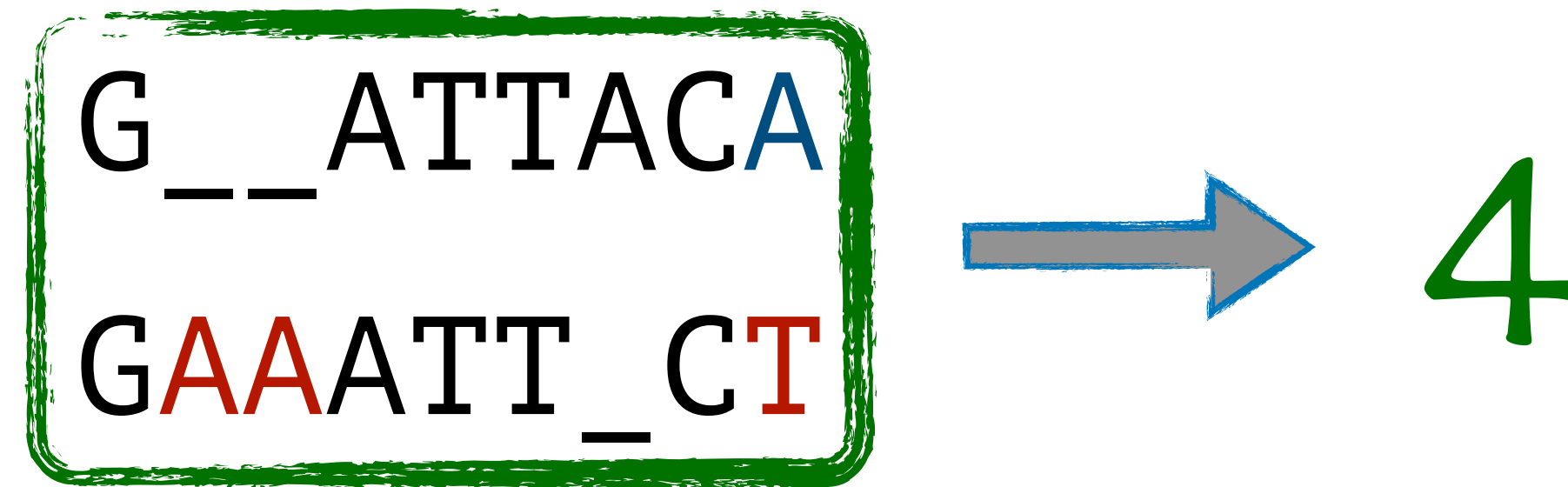
G__ATTACA
GAATT_CT

Рассмотрим пару строк (a, b) где $a_i, b_i \in \mathbb{A}$

Выравнивание — такая пара строк (a^*, b^*) где $a_i^*, b_i^* \in (\mathbb{A} \cup \{__\})$, что

1. $|a^*| = |b^*|$
2. $a_i^* \neq __$ или $b_i^* \neq __$
3. При удалении всех гэпов из a^*, b^* получаем a, b

Стоимость выравнивания

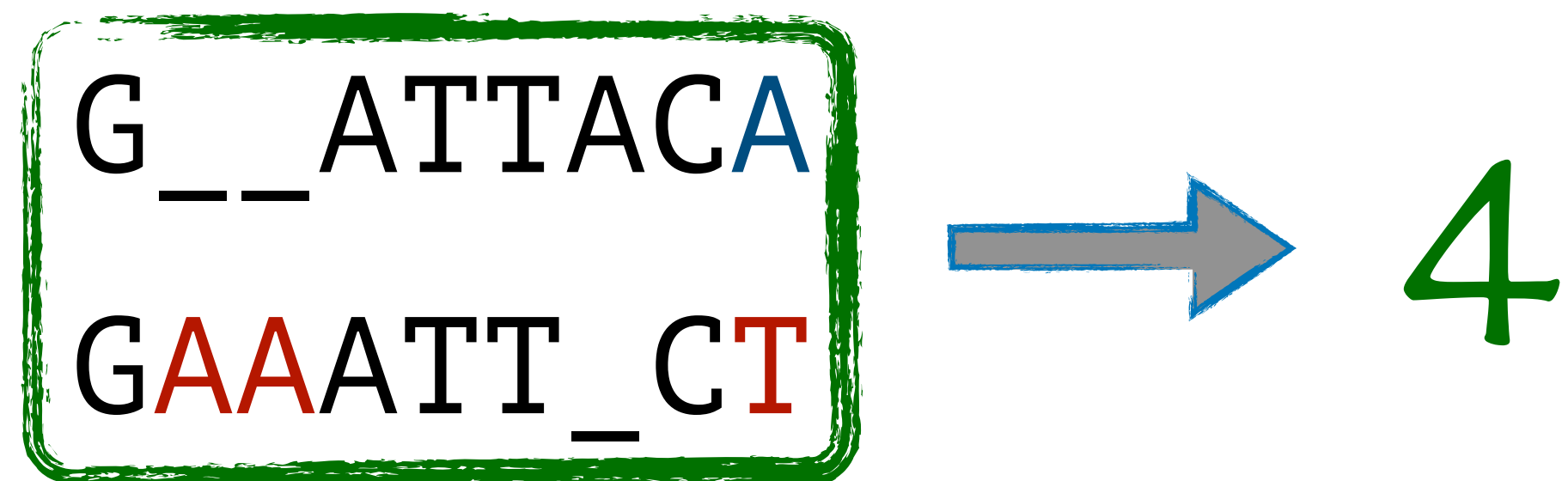


Стоимостью (весом) выравнивания будем называть

$$W(a^*, b^*) = \sum_{i=1}^{|a^*|} w(a_i^*, b_i^*)$$

Где $w(a_i^*, b_i^*)$ функция $(\mathbb{A} \cup \{_ \})^2 \rightarrow \mathbb{R}$

Оптимальное выравнивание и расстояние



Оптимальным будем называть выравнивание, вес которого минимален!

А расстоянием выравнивания — вес оптимального выравнивания.

$$D(a, b) = \min_{a^*, b^*} W(a^*, b^*)$$

Где a^*, b^* — это выравнивание a, b

Редактирование и выравнивание

Для заданной $w(a_i^*, b_i^*)$ расстояние редактирования $d_w(a, b)$ равно расстоянию выравнивания $D_w(a, b)$.

Идея:

Редактирование и выравнивание

Для заданной $w(a_i^*, b_i^*)$ расстояние редактирования $d_w(a, b)$ равно расстоянию выравнивания $D_w(a, b)$.

Идея:

- $d_w(a, b) \leq D_w(a, b)$: выравнивание кодирует последовательность операций редактирования.
- $d_w(a, b) \geq D_w(a, b)$: последовательность операций редактирования порождает выравнивание такое же по весу либо меньше.

Расстояние выравнивания



--GATTACA
AAGATC_

Расстояние выравнивания

Чтобы посчитать $D_w(a, b)$, где $|a| = n$, $|b| = m$ построим матрицу D , $Dim(D) = (n + 1, m + 1)$ по следующим правилам:

- $D_{0,0} = 0$
- $D_{i,0} = D_{i-1,0} + w(a_i, _)$
- $D_{0,j} = D_{0,j-1} + w(_, b_j)$
- $D_{i,j} = \min \begin{cases} D_{i-1,j-1} + w(a_i, b_j) \text{ (замена)} \\ D_{i-1,j} + w(a_i, _) \text{ (удаление)} \\ D_{i,j-1} + w(_, b_j) \text{ (вставка)} \end{cases}$

Замены не равноценны!

Рассмотрим выравнивание аминокислотных последовательностей

Заряженные: **D** (аспарагиновая кислота), **E** (глутаминовая кислота)

Гидрофобные: **I** (Изолейцин), **V** (Валин)

◦ $D \rightarrow E - ?$

◦ $I \rightarrow V - ?$

◦ $D \rightarrow V - ?$

Замены не равноценны!

Рассмотрим выравнивание аминокислотных последовательностей

Заряженные: **D** (аспарагиновая кислота), **E** (глутаминовая кислота)

Гидрофобные: **I** (Изолейцин), **V** (Валин)

- $D \rightarrow E$ — правдоподобно
- $I \rightarrow V$ — правдоподобно
- $D \rightarrow V$ — не очень то и правдоподобно

Замены не равноценны!

Как быть?

Замены не равноценны!

Как быть?

Хотелось бы отличать случайные матчи от вероятных

Замены не равноценны!

Как быть?

Рассмотрим выравнивание (a^*, b^*) и предположим что a^* и b^* не зависят друг от друга. Случайная модель R .

$$P(a, b | R) = \prod_{i=1}^{|a^*|} p_{a_i^*} \prod_{j=1}^{|b^*|} p_{b_j^*} = \prod_{i=1}^{|a^*|} p_{a_i^*} p_{b_i^*}$$

Предположим, пары встречаются не независимо. Модель сопоставления M .

$$P(a, b | M) = \prod_{i=1}^{|a^*|} p_{a_i^*, b_i^*}$$

Замены не равноценны!

Родственные к неродственным

$$\frac{P(a, b \mid M)}{P(a, b \mid R)} = \frac{\prod_{i=1}^{|a^*|} p_{a_i^*, b_i^*}}{\prod_{i=1}^{|a^*|} p_{a_i^*} p_{b_i^*}} = \prod_{i=1}^{|a^*|} \frac{p_{a_i^*, b_i^*}}{p_{a_i^*} p_{b_i^*}}$$

Хотелось бы аддитивную весовую функцию

Замены не равноценны!

Родственные к неродственным

$$\frac{P(a, b \mid M)}{P(a, b \mid R)} = \frac{\prod_{i=1}^{|a^*|} p_{a_i^*, b_i^*}}{\prod_{i=1}^{|a^*|} p_{a_i^*} p_{b_i^*}} = \prod_{i=1}^{|a^*|} \frac{p_{a_i^*, b_i^*}}{p_{a_i^*} p_{b_i^*}}$$

Хотелось бы аддитивную весовую функцию

$$S(a^*, b^*) = \sum_{i=1}^{|a^*|} s(a_i^*, b_i^*), \text{ где } s(a_i^*, b_i^*) = \log \left(\frac{p_{a_i^*, b_i^*}}{p_{a_i^*} p_{b_i^*}} \right)$$

Откуда узнать вероятности?

PAM и BLOSUM

1. База выравниваний BLOCS. Белки, разбитые на блоки
[The Blocks Database—A System for Protein Classification. 1992]

2. Кластеризация. $s_1, s_2 \in C \Leftrightarrow \frac{\#(s_{1,i} = s_{2,i})}{|s_1|} > L$

3. Частоты встречаемости.

Рассмотрим выравнивание a, b из разных кластеров.

Пусть $a \in C_n, b \in C_m$, вычислим $A_{a,b} = \frac{\#(pos(a) = pos(b))}{|C_n||C_m|}$

4. Как пользуясь $A_{a,b}$ вычислить вероятности $p_a, p_{a,b}$?

РАМ и BLOSUM

1. База выравниваний BLOCS.

2. Кластеризация.

3. Частоты встречаемости.

4. Вероятности $p_a, p_{a,b}$

$$p_a = \frac{\sum_b A_{a,b}}{\sum_{cd} A_{c,d}} \text{ — символ } a \text{ выравнился для разных } C_n$$

$$p_{a,b} = \frac{A_{a,b}}{\sum_{cd} A_{c,d}} \text{ — часть тех выравниваний где выравнились } a, b$$

РАМ и BLOSUM

1. База выравниваний BLOCS.
2. Кластеризация.
3. Частоты встречаемости.
4. Вероятности $p_a, p_{a,b}$
5. Воспользуемся функцией $s(a, b)$, чтобы получить матрицу замен!

$$s(a, b) = \log \left(\frac{p_{a,b}}{p_a p_b} \right)$$

BLOSUM. Замечания.

1. Существует BLOSUM65, BLOSUM50. В чем разница?

BLOSUM. Замечания.

1. Существует BLOSUM62, BLOSUM50. В чем разница?
Параметр L используемый для кластеризации.
2. Чему соответствуют меньшие/большие значения L ?

BLOSUM. Замечания.

1. Существует BLOSUM62, BLOSUM50. В чем разница?
Параметр L используемый для кластеризации.
2. Чему соответствуют меньшие/большие значения L ?
Меньшие значения L соответствуют большим эволюционным временам.
3. BLOSUM50 работает для выравниваний с разрывами лучше чем BLOSUM62. [Paerson 1996]

Штрафы за гэпы!



G__ATTACA
GAAATT_CT



G_A_TTACA
GAAATT_CT

- В примерах выше цена выравнивания одинаковая.
- Но первое “биологически адекватнее”! Два маленьких гэпа происходят менее вероятно чем один, но длинны 2.
- Что делать?

Штрафы за гэпы!



G__ATTACA
GAAATT_CT



G_A_TTACA
GAAATT_CT

- В примерах выше цена выравнивания одинаковая.
- Но первое “биологически адекватнее”! Два маленьких гэпа происходят менее вероятно чем один, но длины 2.
- Что делать? Использовать аффинный штраф за гэп!

Штрафы за гэпы!

1. Нам нужна субаддитивная функция штрафа за гэпы:

$$g : \mathbb{N} \rightarrow \mathbb{R}, \text{ причем } g(n + m) \leq g(n) + g(m)$$

2. \nless выравнивание (a^*, b^*) и мультимножество подстрок в нем, содержащих только гэпы Δ .

Вес выравнивания со штрафом за гэпы g и функцией веса замен w

$$W_{w,g}(a^*, b^*) = \sum_{i=1, a_i \neq (), b_i \neq ()}^{|a^*|} w(a_i^*, b_i^*) + \sum_{x \in \Delta} g(|x|)$$

На предыдущем примере:

$(G_A_TTACA, GAATT_CT) \Rightarrow (_, _, _) - \text{мультимножество гэпов.}$

Штрафы за гэпы

Чтобы посчитать $D_{w,g}(a, b)$, где $|a| = n$, $|b| = m$ построим матрицу D , $\dim(D) = (n + 1, m + 1)$ так:

- $D_{0,0} = 0$

- $D_{i,0} = g(i)$

- $D_{0,j} = g(j)$

- $$D_{i,j} = \min \begin{cases} D_{i-1,j-1} + w(a_i, b_j) \text{ (замена)} \\ \min_{k=1}^i D_{i-k,j} + g(k) \text{ (удаление } k \text{ символов)} \\ \min_{k=1}^j D_{i,j-k} + g(k) \text{ (вставка } k \text{ символов)} \end{cases}$$

Штрафы за гэпы

- Что хорошего в предыдущем алгоритме?

Штрафы за гэпы

- Что хорошего в предыдущем алгоритме?
Можно использовать вообще для любых w, g
- Что плохого?

Штрафы за гэпы

- Что хорошего в предыдущем алгоритме?
Можно использовать вообще для любых w, g
- Что плохого?
Сложность $O(n^3)$
:(

Аффинные штрафы за гэпы.

- Используем аффинную функцию g
 $g(k) = \alpha + \beta k$, штраф за начало гэпа α , а за его продолжение β
- Можно использовать алгоритм Гота (Gotoh).
Сложность $O(n^2)$

Алгоритм Гота

Кроме матрицы D , $Dim(D) = (n + 1, m + 1)$ добавим еще матрицы A, B такого же размера.

- $A_{i,j}$ — цена лучшего выравнивания $a_{1..i}, b_{1..j}$, которое заканчивается удалением.

- $B_{i,j}$ — цена лучшего выравнивания $a_{1..i}, b_{1..j}$, которое заканчивается вставкой.

- $$A_{i,j} = \min \begin{cases} A_{i-1,j} + \beta & \text{(расширение удаления)} \\ D_{i-1,j} + g(1) & \text{(начало удаления)} \end{cases}$$

- $$B_{i,j} = \min \begin{cases} B_{i,j-1} + \beta & \text{(расширение вставки)} \\ D_{i,j-1} + g(1) & \text{(начало вставки)} \end{cases}$$

- $$D_{i,j} = \min \begin{cases} D_{i-1,j-1} + w(a_i, b_i) & \text{(замена)} \\ A_{i,j} \\ B_{i,j} \end{cases}$$

Алгоритм Гота

- Сложность $O(n^2)$ по времени и памяти.

Резюмируем

- Выравнивания последовательностей дают наглядное представление об эволюции.
- Важно то, как именно вычислять стоимость замен.
- Выравнивание с аффинными гэпами вычислять не более трудно, чем с обычными.