Мотивы и профили (продолжение)

С.А. Спирин, Р.А. Ириоглов

10 апреля 2026

Информационное содержание одной позиции выравнивания

$$I_k = \sum_X f_k(X) \log_2 \frac{f_k(X)}{p(X)}$$

$p(\mathrm{X})$ — фоновая частота буквы X,
$f_k(\mathrm{X})$ — частота буквы X в позиции k (без псевдоотсчётов!)
В этой формуле считается, что $0 \cdot \log 0 = 0$

Смысл — мера отличия частот в колонке от фоновых частот

Свойства:

1. I_k всегда неотрицательно (это теорема)
2. Значение 0 означает, что все $f_k(\mathrm{X}) = p(\mathrm{X})$
3. Максимальное значение достигается, когда для X с самой малой фоновой частотой имеем $f_k(\mathrm{X}) = 1$, а для остальных 0

Для нуклеотидных мотивов обычно полагают
$p(\mathrm{A}) = p(\mathrm{C}) = p(\mathrm{G}) = p(\mathrm{T}) = 0{,}25$
В таком варианте I_k принимает значения от 0 до 2

Информационное содержание мотива

TTATGCC
ATCTTCA
GTATTAA

Выравнивание

	1	2	3	4	5	6	7
G	0.26	-1.3	-1.3	-1.3	0.26	-1.3	-1.3
A	0.26	-1.3	0.74	-1.3	-1.3	0.26	0.74
T	0.26	1.18	-1.3	1.18	0.74	-1.3	-1.3
C	-1.3	-1.3	0.26	-1.3	-1.3	0.74	0.26

PWM

$$I_k = \sum_X f_k(X) \log_2 \frac{f_k(X)}{p(X)}$$

$$I = \sum_k I_k$$

Информационное содержание (ИС, I) позволяет понять, как много похожих на мотив последовательностей мы найдем в наших данных по случайным причинам

Элементы PWM:

$$w_k(X) = \log \frac{f_k(X)}{p(X)}$$

$w_i(\mathrm{X})$ для буквы X в позиции i,
$p(\mathrm{X})$ — фоновая частота буквы X,
$f_k(\mathrm{X})$ — частота буквы X в позиции k
(с учётом псевдоотсчётов)

Здесь логарифм по любому основанию (как удобнее), разницы нет. Для ИС используется логарифм по основанию 2.

Информационное содержание мотива

$$I_k = \sum_X f_k(X) \log_2 \frac{f_k(X)}{p(X)}$$

$$I = \sum_k I_k$$

ИС позволяет понять, как много похожих на мотив последовательностей мы найдем в наших данных по случайным причинам

В случае ДНК и равных фоновых частот, если **все** позиции мотива строго консервативны, то ИС будет равно 2*n*, где *n* — длина мотива.

Вероятность найти такой мотив в заданном месте случайной последовательности равна $1/4^n = 1/2^I$

Приближённо можно считать, что эта формула верна всегда.

Тем самым:

В наборе последовательностей суммарной длины N в среднем найдётся $N/2^I$ случайных находок мотива с ИС, равным I

LOGO-диаграмма

FRYGREDLDVLGLTFRKDL
YRYGREEDEVMGVKFSKEL
YRYGREEDEVMGVKFSKEL
YRYGREEDEVMGVKFSKEL
FRYGRDDMELIGLSFRKDI
FRYGHDDLDVIGLTFRKDL
FRYGRDDLEVIGLTFRKDL
FRYGRDDLDVIGLTFRKDL
FRYGRDDMELIGLSFRKDI

Высота изображения в позиции = ИС этой позиции

Высота делится между буквами пропорционально их частотам

Generated with https://skylign.org/

Как быть с гэпами?

TTATTGCC
ATCT-TCA
GTAT-TAA

Как быть с гэпами — паттерны

```
TTATTGCC
ATCT-TCA
GTAT-TAA
```

В синтаксисе паттернов предусмотрена возможность вставки неопределённого количества любых букв:

```
[AGT]-T-[AC]-T-x(0,1)-[GT]-[AC](2)
```

Как быть с гэпами — профили

```
TTATTGCC
ATCT-TCA
GTAT-TAA
```

Казалось бы, можно построить что-то вроде PWM, но добавить дополнительную «букву» — гэп.

Но как искать такой мотив в новых последовательностях, чтобы учитывать частоту гэпа в колонке исходного выравнивания?

Как быть с гэпами — профили

```
TTATTGCC
ATCT-TCA
GTAT-TAA
```

Казалось бы, можно построить что-то вроде PWM, но добавить дополнительную «букву» — гэп.

Но как искать такой мотив в новых последовательностях, чтобы учитывать частоту гэпа в колонке исходного выравнивания?

Ответ: надо выравнивать профиль с последовательностью, допуская гэпы только в последовательности. Штраф за гэп должен зависеть от частоты гэпа в пропускаемых позициях профиля.

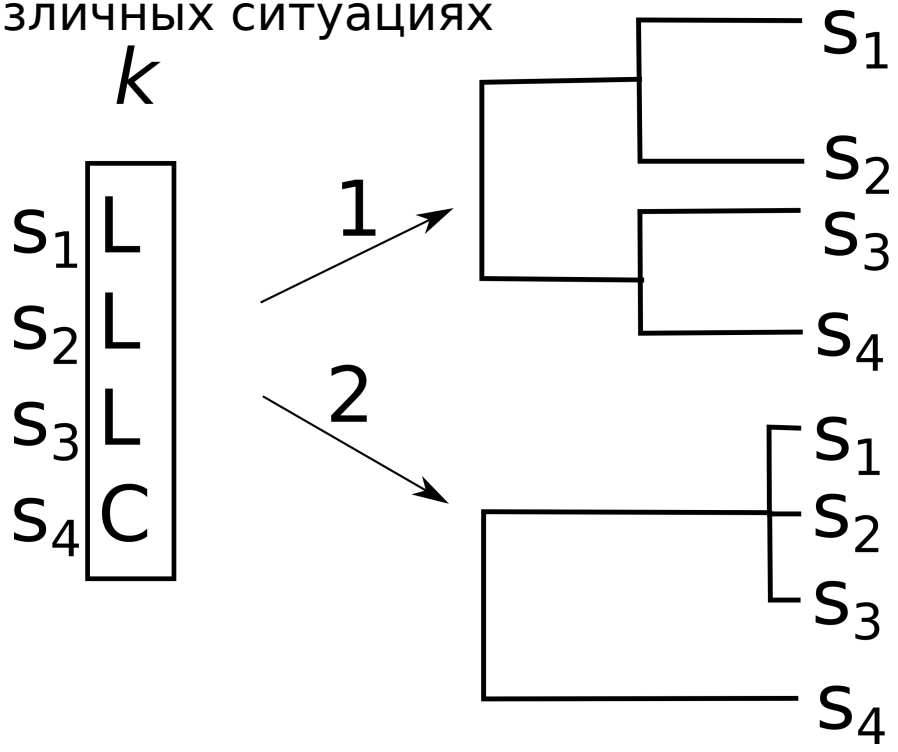
То, что получается, принято называть PSSM — position-specific scoring matrix

PSSM — position-specific scoring matrix

- PSSM отличается от PWM учётом возможности гэпов (и, как следствие, способом поиска мотива в новых последовательностях)
- Как правило, PSSM используется при работе с семействами белков, а PWM — при работе с мотивами в нуклеиновых кислотах

Чрезмерный вклад родственных последовательностей

Одинаковая с виду колонка в двух различных ситуациях



- Последовательности, из которых состоит выравнивание, могут быть очень близкородственны, как последовательности s_1-s_3 в случае 2. Частота L в данной колонке равна ¾, но можем ли мы в обоих случаях считать, что L более вероятна в данной позиции, чем C?
- Нет, это предположение верно для случая 1, а для случая 2 три L являются скорее следствием близости содержащих L белков.
- Если в нашем выравнивании много близкородственных последовательностей, они будут ухудшать качество профиля. Чтобы этого избежать, частоты остатков в колонке оценивают с учетом весов (weights) последовательностей, которые содержат эти остатки в данной колонке.
  Веса последовательностей рассчитывают так, чтобы суммарный вес группы близкородственных последовательностей был ненамного больше веса последовательности, не имеющей близких родственников.

Для PWM (сигнал в геноме) такой проблемы обычно нет, а для PSSM (принадлежность белка семейству) она почти всегда актуальна. Стоит подумать, почему.

Оценка частоты остатка в позиции с учетом веса последовательности

Придумали такой способ: присвоить каждой последовательности вес (weight) так, чтобы у последовательностей, имеющих много родственников, он был маленьким, а у «одиноких» последовательностей — большой. При расчете частоты буквы X в позиции k используются веса последовательностей w_s:

$$f_k(X) = \frac{\sum_{s:a_{sk}=X} w_s + \psi(X)}{\sum_s w_s + \sum_X \psi(X)}$$

Если все веса последовательностей равны, то получится обычная частота.
Здесь a_{sk} — буква последовательности *s* в позиции *k*,
$\psi(\mathrm{X})$ — псевдоотсчёт для остатка X.

Внимание: слово «вес» имеет два разных значения

- Вес = Score, вес выравнивания двух последовательностей или последовательности относительно профиля (PWM или PSSM или HMM), обычно обозначается *S*.
- Вес = Weight, вес последовательности, используемый при построении PSSM по множественному выравниванию, обычно обозначается *w*.

Дополнительная информация: как получается вес последовательности?

Как получить такой вес последовательности (w_s), чтобы он был маленьким у последовательностей, имеющим в нашем выравнивании близких родственников, и большим у одиноких последовательностей?
Например, так
(а всего около десятка только популярных способов).

- Дано выравнивание: последовательность s содержит букву $a_{s,k}$ в позиции k.
  Сначала припишем вес каждой **букве** выравнивания:
  - Пусть в *k*-ой позиции выравнивания встречается *r*(*k*) типов аминокислотных остатков. Сделаем так, чтобы суммарный вес каждого типа был равен 1/*r*(*k*) (то есть суммарный вес всех аланинов равнялся суммарному весу лейцинов и т.д.)
  - Для этого посмотрим, в скольких последовательностях содержится каждый тип остатка. Пусть такой же остаток, как $a_{s,k}$, встречается в нашей позиции *k* всего в $n(a_{s,k}, k)$ разных последовательностях. Припишем букве $a_{s,k}$ вес $w_{s,k} = 1/r(k)n(a_{s,k}, k)$.
- Вес последовательности будет равен сумме весов всех её букв:

$$w_s = \sum_k w_{s,k}$$

Добавление псевдоотсчётов

Используются по крайней мере три способа избавиться от 0 в матрице частот:

1. Добавление 1 к наблюдаемому количеству каждой буквы (правило Лапласа). Не учитываем разную частоту встречаемости разных остатков. Стандартный вариант для нуклеотидов (PWM).
2. Добавление фоновой частоты остатка в банке белковых последовательностей — лучше, но не учитываем свойства остатков.

Например, если в данной позиции выравнивания стоит Leu, то вероятность появления в этой позиции похожего по свойствам остатка (например, Met) должна быть больше фоновой, а непохожего (например, Asp) — меньше фоновой.

3. Учёт $q(X,Y)$: частот замен из «образцовых» выравниваний (тех же, что использовались при создании матриц замен остатков, например BLOSUM62)

Матрица BLOSUM62

	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	4																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

$$s(X,Y) = \frac{1}{\lambda} \log \frac{q(X,Y)}{p(X)p(Y)}$$

здесь $S(X,Y)$ — элемент матрицы замен, $p(X)$, $p(Y)$ — фоновые частоты остатков X и Y, $q(X,Y)$ — частота сопоставления X с Y в «образцовых» выравниваниях (из базы BLOCKS).

Расчет ожидаемой частоты остатка

$$\psi_k(X) = \sum_Y n_k(X) \frac{q(X,Y)}{p(Y)}$$

$$Q_k(X) = \alpha n_k(X) + \beta \psi_k(X)$$

Здесь $n_k(\mathrm{X})$ — «эффективное число встреч» буквы X, то есть сумма весов последовательностей, которые в позиции *k* содержат X
$q(X,Y)$ — частота сопоставления X с Y в «образцовых» выравниваниях
$p(\mathrm{Y})$ — фоновая частота буквы Y
α и β — некоторые коэффициенты.
В PSI-BLAST (см. далее) полагают β =10, а α равным числу **различных** букв в колонке

Теперь вместо частоты $f_k(\mathrm{X})$ используют $Q_k(\mathrm{X})/\Sigma_{\mathrm{X}} Q_k(\mathrm{X})$

Использование PSSM

PSSM можно «выравнять» с белковой последовательностью и получить вес, аналогично весу выравнивания двух последовательностей.

PSSM используется при поиске в банке данных программой PSI-BLAST

PSI-BLAST (Position-Specific Iterative BLAST) — разновидность BLASTP, использующий PSSM, благодаря чему он способен находить дальних родственников заданного белка.

Алгоритм PSI-BLAST

- На входе — последовательность и порог по e-value, на выходе — набор найденных последовательностей и построенный по ним PSSM.
- 1. На первом этапе запускается обычный BLASTP входной последовательности против выбранного банка последовательностей
- 2. Для находок со значениями e-value лучше заданного порога строится множественное выравнивание.
- 3. Это выравнивание используется для получения PSSM.
- 4. На следующем шаге опять происходит запуск BLAST для исходной последовательности против того же банка последовательностей, но вместо матрицы замен остатков используется PSSM, полученная на предыдущем шаге.
- 5. Повторяем шаги 2-4, пока не перестанут добавляться новые последовательности.

Дополнительные возможности PSI-BLAST

- Можно вручную включать/исключать последовательности, которые используются для построения PSSM
- Можно использовать PSSM, созданную на основе поиска в одном банке, для поиска в другом банке.