Правительство Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Санкт-Петербургский государственный университет»

Кафедра Системного Программирования

Атаманова Анна Михайловна

# Скрытые Марковские модели переменного порядка для анализа данных ChIP-seq

Бакалаврская работа

Допущена к защите.
Зав. кафедрой:
д. ф.-м. н., профессор Терехов А. Н.

Научный руководитель:
д. ф.-м. н., профессор Терехов А. Н.

Рецензент:
Тузова Е. А.

Санкт-Петербург
2015

SAINT-PETERSBURG STATE UNIVERSITY

Chair of Software Engineering

Anna Atamanova

# Variable-length hidden Markov models for ChIP-seq data analysis

Bachelor's Thesis

Admitted for defence.
Head of the chair:
professor Andrey Terekhov

Scientific supervisor:
professor Andrey Terekhov

Reviewer:
Ekaterina Tuzova

Saint-Petersburg
2015

# Оглавление

**Введение** **4**

**1. Постановка задачи** **7**

**2. Обзор существующих решений** **8**

2.1. Основные понятия и определения . . . . . 8

2.2. Скрытые Марковские модели . . . . . 10

2.3. Обучение модели СММПП . . . . . 11

2.3.1. Инициализация . . . . . 12

2.3.2. EM (Expectation–Maximization algorithm) . . . . . 12

2.3.3. Подрезание дерева . . . . . 14

2.4. Обучение на нескольких выборках . . . . . 15

2.5. Сравнение . . . . . 15

**3. Реализация** **18**

**4. Применение** **19**

4.1. Применение к симулированным данным . . . . . 19

4.1.1. Пуассоновская смесь . . . . . 19

4.1.2. СММ . . . . . 20

4.1.3. СММПП, не являющаяся СММ фиксированного порядка . . . . 21

4.2. Применение к реальным данным . . . . . 21

**Заключение** **24**

**Список литературы** **25**

# Введение

## Предметная область

Дезоксирибонуклеиновая кислота (ДНК) — молекула, обеспечивающая хранение генетического кода, который определяет развитие и функционирование живых организмов. ДНК хранит наследственную информацию, информацию о структуре РНК и белков. Белки выполняют структурные, сигнальные, механические и другие функции. Соединения ДНК с конкретным белком могут влиять на конформацию ДНК, на внутренние механизмы управления клетки, поэтому изучение ДНК-белковых взаимодействий крайне важно и актуально.

Геном — это совокупность всех молекул ДНК в клетке. Каждая ДНК состоит из двух цепей нуклеотидов, поэтому длина генома измеряется в парах нуклеотидов (пн).

В данной работе рассматривается задача нахождения позиций связывания ДНК и конкретного белка, то есть нахождения позиций ДНК-белковых взаимодействий при заранее выбранном белке.

## ChIP-seq

ChIP-seq (иммунопреципитация хроматина с последующим секвенированием, англ. chromatin immunoprecipitation sequencing) — биологический эксперимент, позволяющий получить информацию о наличии или отсутствии взаимодействия ДНК с заданным белком.

Эксперимент проводится на множестве одинаковых клеток и включает в себя следующие стадии:

1. фиксация всех обнаруженных белков на ДНК;
2. расщепление ДНК на фрагменты;
3. вылавливание фрагментов, связанных с исследуемым белком (с помощью специфичного к исследуемому белку антитела);
4. считывание концов фрагментов (называемых ридами или прочтениями) до тех пор, пока каждый фрагмент с высокой вероятностью не будет прочитан несколько раз.

Далее для каждого полученного рида ищется соответствующий ему участок последовательности генома (рис. 1). Обычно риды, которым может соответствовать более одного участка в геноме, исключают из рассмотрения.

Результаты эксперимента представляют в виде массива длины генома, в позиции которого стоит 1, если в соответствующей позиции генома начиналось хотя бы одно прочтение и 0 в обратном случае.

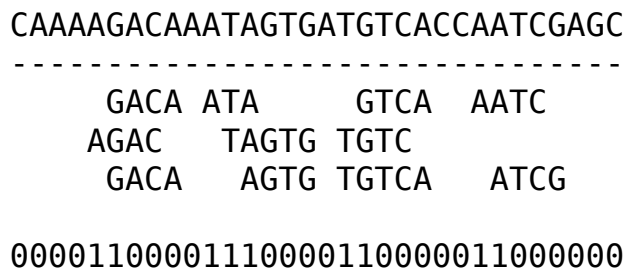



Рис. 1: Схематичное изображение выравнивания прочтений секвенатора (под чертой) на известную последовательность генома (над чертой).

Соединение белка с ДНК происходит не точечно, а на некотором участке ДНК, поэтому, для дальнейшего анализа, полученный массив разбивается на отрезки заранее выбранной длины, называемые окнами (обычно 200 пн). Значение в окне определяется как сумма единиц в нем.

Эксперимент ChIP-seq (как и большинство биологических экспериментов) не исключает наличие ошибок в результатах. Недостаточная специфичность антитела, наличие ошибок секвенирования, нестабильность положения белка на ДНК приводят к возникновению сигнала, не зависящего от наличия взаимосвязи. Поэтому, для дальнейшего анализа результатов эксперимента, требуется построение вероятностной модели, способной отделять ошибки, а также выявлять зависимости соединений и, по возможности, описывать их структуру.

Большинство существующих моделей ([8], [5]) для данных ChIP-seq основано на аппарате скрытых Марковских моделей (СММ) первого порядка [4] с Пуассоновскими испусканиями. Использование распределения Пуассона для моделирования покрытия опирается на предположение о том, что в каждой позиции генома в среднем начинается одинаковое количество прочтений. Марковский процесс, как правило, имеет два состояния «1» — сигнал есть и «0» — сигнала нет. Первый порядок модели означает, что состояние некоторого окна зависит только от состояния его прямого предшественника. Использование моделей первого порядка объясняется тем, что количество параметров модели, а также сложность её обучения и использования экспоненциально зависят от порядка. Так, СММ порядка $m$ для каждой цепочки из $m$ состояний содержит распределение на следующее состояние ($2^m$ вероятностных распределений). В связи с этим, неправильный выбор $m$ при обучении сильно усложняет модель и способствует ее переобучению. Переобучение — это одна из основных проблем машинного обучения, при которой модель слишком сильно подгоняется под обучающую выборку и находит в ней случайные закономерности, которые не характерны для данных генеральной совокупности.

Скрытые Марковские модели переменного порядка менее склонны к переобучению в силу того, что они не фиксируют длину строки, порождающей следующее состояние,

и стараются ее уменьшить.

# 1. Постановка задачи

Целью данной дипломной работы является построение скрытой Марковской модели переменного порядка для анализа данных ChIP-seq.

Для достижения цели были определены следующие задачи:

1. реализовать скрытую Марковскую модель переменного порядка;
2. проанализировать эффективность работы модели на синтетических данных;
3. применить к данным ChIP-seq, сравнить с более простыми моделями (СММ первого порядка).

# 2. Обзор существующих решений

Марковские модели переменного порядка (не скрытые) обучаются путем построения контекстного дерева переходов [2]. Скрытые Марковские модели фиксированного порядка обучаемы алгоритмом Баума-Велша [4]. В работе [7] было предложено совмещение этих двух идей для обучения скрытых Марковских моделей переменного порядка (СММПП).

В данной работе алгоритмом обучения СММПП был выбран модифицированный под поставленную задачу алгоритм из [7]. Модификация заключается в следующем: наблюдения итоговой модели будут порождаться из соответствующих состояний, т.е. распределение значений для каждого окна задается скрытым состоянием, которое определяет, была ли там взаимосвязь с белком или нет. В работе [7] такие распределения определялись всем контекстом. Алгоритм был дополнен недостающей информацией об обучении контекстных деревьев из статей [2], [3].

## 2.1. Основные понятия и определения

Путь $S = \{0, 1\}$ — множество состояний (в рамках рассматриваемой задачи, «1» означает наличие связи, «0»— ее отсутствие), $X_0, X_1, \ldots$ — последовательность случайных величин (дискретный случайный процесс), значения которых лежат в S, а $x_0, x_1, \ldots$ — некоторая реализация случайных величин $X_0, X_1, \ldots$.

**Определение 1.** $\{X_i\}_{i \in Z_+}$ называется *Марковским процессом порядка* $m$, если

$$\begin{aligned}
&\forall t, t' \in N,\ t, t' \geq m,\ \forall \overrightarrow{x} \in S^{t+1} \\
&P(X_t = x_t | X_{t-1} = x_{t-1}, X_{t-2} = x_{t-2}, \ldots, X_0 = x_0) \\
&= P(X_t = x_t | X_{t-1} = x_{t-1}, X_{t-2} = x_{t-2}, \ldots, X_{t-m} = x_{t-m}) \\
&= P(X_{t'} = x_t | X_{t'-1} = x_{t-1}, X_{t'-2} = x_{t-2}, \ldots, X_{t'-m} = x_{t-m}) \\
&= P(X_{t'} = x_t | X_{t'-1} = x_{t-1}, X_{t'-2} = x_{t-2}, \ldots, X_0 = x_0) \qquad (1)
\end{aligned}$$

Далее, для Марковских процессов, вероятности вида

$$P(X_{t'} = x_t | X_{t'-1} = x_{t-1}, X_{t'-2} = x_{t-2}, \ldots, X_{t'-m} = x_{t-m})$$

где $t' \geq m$, будем записывать как $P(x_t | x_{t-1} \ldots x_{t-m})$. Запись корректна, в силу независимости такой вероятности от $t'$.

Для удобства будем считать, что наш процесс растет справа налево

$$\ldots x_t,\ x_{t-1},\ x_{t-2} \ldots$$

Так, если цепь $\ldots x_t, x_{t-1}, x_{t-2} \ldots$ была порождена процессом порядка 2, то

$$P(x_t|x_{t-1}, x_{t-2} \ldots) = P(x_t|x_{t-1}, x_{t-2})$$

**Определение 2.** *Марковская модель порядка* $m$ — это вероятностная модель, описывающая марковский процесс порядка $m$. Параметрами модели являются множество вероятностных распределений переходов $A = \{a(q; x^m)\}_{q \in S, x^m \in S^m}$, где $a(q; x^m) = P(q|x^m)$, и начальное распределение $\pi = \pi(x^m)_{x^m \in S^m}$, где $\pi(x^m) = P(X_{0:m} = x^m)$.

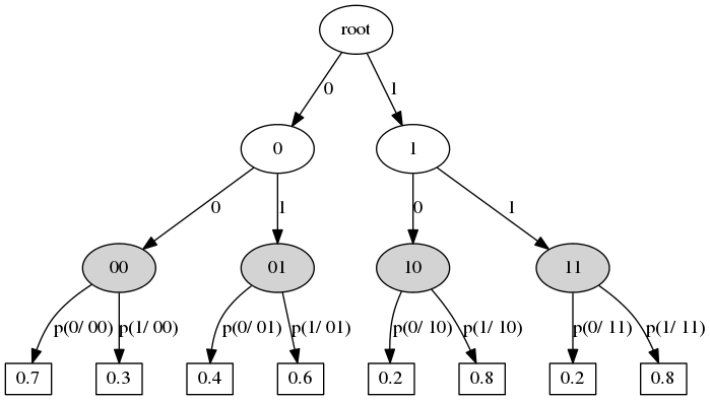
**Определение 3.** *Контекстное дерево* — это дерево, в котором каждая внутренняя вершина имеет $|S|$ ребер, соответствующих состояниям из $S$, и метку, которая является конкатенацией метки на ее родителе и метки ребра от него. Метка в корне — пустая строка.

*Контекстом* состояния $x_t$ будем называть любой префикс строки $x_{t-1}, x_{t-2} \ldots$. Контексты, соответствующие листьям контекстного дерева, будем называть *главными контекстами* (иногда, когда речь будет идти только о листьях, слово «главные» будем опускать).

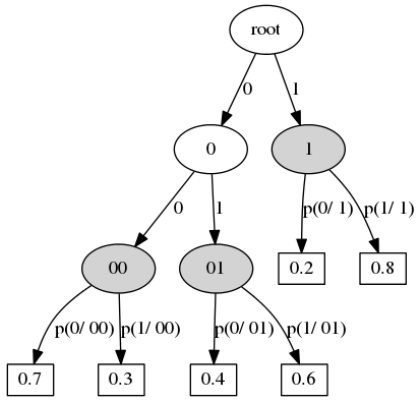
Множество переходов для Марковского процесса порядка $m$ можно определить, как контекстное дерево глубины $m+1$, каждый лист которого содержит распределение $P(. \mid w)$, где $w$ — метка на листе.

Для того, чтобы по дереву определить распределение на следующем состоянии $X_t$, достаточно из корня спуститься по ветке, вершины которой соответствуют контекстам этого состояния, $(x_{t-1})$, $(x_{t-1}x_{t-2}), \ldots$ . Лист на конце ветки и будет задавать распределение $X_t$.

*Замечание* 1. Метки на листьях контекстного дерева полностью определяют дерево.




(а) Контекстное дерево переходов Марковского процесса порядка 2




(б) Подрезанное контекстное дерево

Рис. 2: Эквивалентные контекстные деревья.
Серым подкрашены листья, ниже прямоугольниками обозначены распределения переходов.

На рисунке 2а изображен пример контекстного дерева для Марковского процесса порядка 2. Можно заметить, что в этом примере, имея для некоторого состояния $x_t$ контекст «1» , необходимость уточнять его (т.е. спускаться дальше к листу) отсутствует, т.к. распределение на контекстах «10» и «11» одно и тоже. Таким образом подрезанное дерево с рисунка 2б задает такие же распределения переходов как и дерево с рисунка 2а. Однако второе контекстное дерево меньше (число главных контекстов меньше). Но ни один Марковский процесс фиксированного порядка напрямую его использовать не может.

Определим процесс, который может иметь распределение переходов в виде дерева с рисунка 2б.

Пусть $\tau$ — конечное контекстное дерево. Для $s \in \tau$ будем обозначать через $C(s)$ множество всех потомков, являющихся листами $\tau$. Для $s \notin \tau$, $C(s)$ — лист $\tau$, являющийся префиксом $s$ (можно заметить, что он существует и единственен).

**Определение 4.** *Марковский процесс переменного порядка* (МППП) с максимально-возможным порядком $m$ — это вероятностный процесс, распределения на состояниях которого задаются распределениями на листьях некоторого контекстного дерева $\tau$ глубины не более чем $m + 1$.

**Определение 5.** Марковская модель переменного порядка (ММПП) с максимально-возможным порядком $m$ — вероятностная модель, описывающая соответствующий процесс. Параметрами модели являются множество распределений переходов на листьях некоторого контекстного дерева $\tau$ глубины не более чем $m + 1$ и начальное вероятностное распределение на них (листьях).

*Замечание* 2. Распределения переходов на внутренних вершинах контекстного дерева определяются распределениями на листьях.

$$P(q|s) = \frac{\sum_{c \in C(s)} a(q; c)P(c)}{\sum_{q' \in S} \sum_{c \in C(s)} a(q'; c)P(c)} \tag{2}$$

*Замечание* 3. Вероятности контекстов, соответствующих вершинам контекстного дерева, определяются вероятностным распределением на листьях.

$$P(s) = \sum_{c \in C(s)} P(c) \tag{3}$$

*Замечание* 4. МППП с максимально-возможным порядком $m$ есть обобщение всех Марковских процессов порядка меньше либо равного, чем $m$.

## 2.2. Скрытые Марковские модели

Представим, что состояния — это какой-то скрытый признак/фактор (например, наличие или отсутствие связи белка и ДНК) цепи наблюдений $Y = \{y_t\}_{t \in Z_+}$. Для каж-

дого наблюдения $y_t$ он не известен, однако именно он определяет распределение на $Y_t$. Т.е. цепь $Y$ порождается из Марковской цепи $X = \{x_t\}_{t \in Z_+}$ путем покоординатного определения новой случайной величины $Y_t$ для каждого состояния $x_t$ согласно распределению $P(.|\ x_t)$.

**Определение 6.** Процесс, порождающий цепь по некоторому Марковскому процессу $X = \{x_t\}_{t \in Z_+}$ порядка $m$ и распределению $P(.|\ x_t)$, называется *скрытым Марковским процессом порядка* $m$. $X$ называеются *скрытыми состоянияниями*, $Y$ — *наблюдениями.*

**Определение 7.** *Скрытая Марковская модель* (СММ) порядка $m$ — вероятностная модель, описывающая соответствующий процесс. Параметрами модели является $\Lambda = (A, \pi, B)$, где $A, \pi$ — параметры скрытого процесса $X$ порядка $m$, $B = \{b(y; x)\}_{y \in R^l, x \in S}$ — множество распределений испусканий (где $b(y; x) = P(y|x)$).

**Определение 8.** *Скрытая Марковская модель переменного порядка* (СММПП) — вероятностная модель, описывающая соответствующий процесс. Параметрами модели является $\Lambda = (A, \pi, B)$, где $A, \pi$ — параметры скрытого процесса переменного порядка $X$ , $B = \{b(y; x)\}_{y \in R^l, x \in S}$, где $b(y; x) = P(y|x)$ — множество распределений испусканий.

## 2.3. Обучение модели СММПП

Задачу обучения скрытой Марковской модели переменного порядка можно сформулировать следующим образом: по цепи наблюдений $Y = (y_1, ...y_T)$ найти параметры $\Lambda = (A, B, \pi)$ модели СММПП, которые бы минимизировали количество контекстов не сильно ухудшая правдоподобие модели по сравнению с правдоподобием модели на полном дереве (допустимое отклонение распределений регулирует параметр $\varepsilon_{\text{prune}}$). Правдоподобие модели с параметрами $\Lambda$ на выборке $Y$ — это вероятность породить $Y$ из данной модели, $P(Y|\Lambda)$. При обучении моделей правдоподобие считается на обучающей выборке.

В листинге 1 схематично представлен алгоритм обучения СММПП. Описание основных шагов: инициализация, EM-алгоритм, подрезание дерева — приведено далее.

Конкатенацию строк $a$ и $b$ будем обозначать $ab$. За $\pi(s)$, где $s \in \tau$, будем обозначать продолжение $\pi$ на все дерево (формула 3). Общее распределение на контекстах будем обозначать параметром $\rho(c) := P(c)$.

**Data**:
$Y$, // наблюдения
$m$, $\varepsilon_{EM}$, $\varepsilon_{prune}$
//параметры обучения: максимальная длина контекста,
//порог для остановки EM, порог для обрезания дерева
**Result**: $\Lambda$ // параметры СММПП
$\Lambda$ = *Инициализация(*$Y$, $m$*)* ;
// полное контекстное дерево
**while** *контекстное дерево уменьшается* **do**
  $\Lambda$ = *EM(*$Y$, $\Lambda$, $\varepsilon_{EM}$*)*;
  // максимизируем правдоподобие модели на наблюдениях
  //при фиксированной структуре дерева
  $\Lambda$ = *Подрезание(*$\Lambda$, $\varepsilon_{prune}$*)*;
  // подрезаем дерево, если обученные распределения при этом несильно
  // изменяются
**end**

**Algorithm 1:** Схема обучения СММПП

### 2.3.1. Инициализация

Начальное контекстное дерево является полным $|S|$-нарным деревом глубины $m+1$ (соответствеут СММ порядка $m$).

В качестве начальных распределений переходов и параметров испусканий берутся оценки этих величин на цепи состояний, полученной алгоритмом $k$-means ($k = |S|$) по цепи наблюдений $Y$. [1]

### 2.3.2. EM (Expectation–Maximization algorithm)

Оценки параметров проводятся подобно алгоритму Баума-Велша для СММ [4].

1. E-шаг (Expectation)

   Введем дополнительный параметр $\alpha$

   $$\alpha_t(c) := P(y_0^t, c(x_t) = c|\Lambda)$$

   $\alpha_t(c)$ — вероятность породить первые $t+1$ наблюдений равными $y_0^t$, имея главным контекстом скрытого состояния $x_t$ контекст $c$, из модели СММПП с пара-

[1] В общем случае, EM-алгоритм допускает случайную инициализацию.

метрами $\Lambda$.

$$\alpha_0(c) = \pi(c)b(y_0; c[0])$$
$$\alpha_{t+1}(c) = \sum_{q \in S, c' = C(cq)} \alpha_t(c')a(c[0]; c')b(y_{t+1}; c[0]) \tag{4}$$

Если $c'$ оказался внутренним листом дерева, то величина $a(c[0]; c')$ считается по формуле 2 с распределением на листьях равным $\pi$, распределением переходов равным $A$.

Введем дополнительный параметр $\beta$

$$\beta_t(c) := P(y_{t+1}^T | c(x_t) = c, \Lambda))$$

$\beta_t(c)$ — вероятность того, что последние $T - t$ наблюдений цепи длины $T$, порожденной из модели СММПП с параметрами $\Lambda$, в которой главный контекст скрытого состояния $x_t$ является $c$, совпадают с $y_{t+1}^T$.

$$\beta_T(c) = 1$$
$$\beta_t(c) = \sum_{q \in S, c' = C(qc)} a(q; c)b(y_{t+1}, c'[0])\beta_{t+1}(c') \tag{5}$$

Введем дополнительный параметр $\gamma$

$$\gamma_t(c) := P(x_t = c | Y, \Lambda)$$

$\gamma_t(c)$ — вероятность того, что породив цепь $Y$ моделью СММПП с параметрами $\Lambda$, главный контекст скрытого состояния $x_t$ является $c$.

$$\gamma_t(c) \propto \alpha_t(c)\beta_t(c) \tag{6}$$

2. М-шаг (Maximization)

На этом шаге алгоритм обновляет параметры модели, максимизируя правдоподобие при условии посчитанных величин $\alpha, \beta, \gamma$.

Для пересчета параметра $A$ введем параметр $\xi$

$$\xi_t(q; c) := P(c(x_t) = c, x_{t+1} = q | Y, \Lambda)$$

$\xi_t(q; c)$ — вероятность того, что породив цепь $Y$ моделью СММПП с параметрами $\Lambda$, главный контекст скрытого состояния $x_t$ является $c$ и состояние $x_{t+1}$ совпадает с $q$.

$$\xi_t(q; c) \propto \alpha_t(c)a(q; c)b(y_{t+1}, q)\beta_{t+1}(qc) \tag{7}$$

Обновление $A, \rho, \pi$:

$$a(q; c) \propto \sum_t \xi_t(q, c) \tag{8}$$

$$\rho(c) \propto \sum_t \gamma_t(c) \tag{9}$$

$$\pi(c) \propto \gamma_0(c) \tag{10}$$

Пересчет $B$ зависит от принятого семейства моделей испусканий и производится с помощью $\gamma$ в точности также, как и в алгоритме Баума-Велша. В случае распределения Пуассона $b(. \mid q) \sim Poisson(\lambda_q)$ пересчет параметров происходит следующим образом:

$$\lambda_q = \frac{\sum_c \sum_t \gamma_t(c) I[c[0] = q] y_t}{\sum_c \sum_t \gamma_t(c) I[c[0] = q]} \tag{11}$$

EM-алгоритм запускает поочередно E-шаг и M-шаг, пока правдоподобие с предыдущей итерации отличается от правдоподобия с текущей итерации более, чем на $\varepsilon_{EM}$, т.е. пока итерация дает значимый прирост правдоподобия

$$P(Y|\Lambda) = \sum_{c \in C} \alpha_T(c) \tag{12}$$

*Замечание* 5. При пересчете вероятности могут очень близко подходить к нулю, что отрицательно влияет на точность расчета. Для избежания этой проблемы все расчеты проводятся не с вероятностями, а с их логарифмами.

*Замечание* 6. EM-алгоритм следует запускать несколько раз, т.к. он может «застревать» в локальных максимумах функции правдоподобия.

### 2.3.3. Подрезание дерева

Если существует внутренний лист контекстного дерева $s$ такой, что

$$\forall q \in S \;\; P(sq) KL(sq, s) < \varepsilon_{prune} \tag{13}$$

(дети не уточняют родителя), то $s$ становится листом, а все его потомки обрезаются, где

$$KL(u, w) = \sum_{q' \in S} P(q'|u) \log \frac{P(q'|u)}{P(q'|w)} \tag{14}$$

$$\tag{15}$$

расстояния Кульбака-Лейблера для апостериорных распределений.

Если таких листьев не существует, алгоритм заканчивает работу.

$P(s), P(q|s)$, где $s \in \tau$ (и, как частный случай, новые $\rho$, $\pi$, $A$) считаются по формулам 3, 2, соответственно (по еще неподрезанному $\tau$).

## 2.4. Обучение на нескольких выборках

В случае пропусков в наблюдениях (связанных, например, с отсутствием данных), обучение модели может проходить на множестве из нескольких цельных кусков наблюдений. Т.е. на вход алгоритма будет подаваться не одна выборка $Y$, а $N$ выборок $\{Y^1 \dots Y^N\}$, подчиненных единому скрытому Марковском процессу переменного порядка.

Для применения вышеописанного алгоритма для обучения СММПП на нескольких выборках были внесены изменения в Е-шаг и М-шаг.

1. Е-шаг

   Дополнительные параметры $\alpha^d, \beta^d, \gamma^d, \xi^d$ пересчитываются отдельно на каждой выборке $d \in \{1, \dots, N\}$ по формулам 4, 5, 6, 7, соответственно.
   Общая $\gamma$ — конкатенация $\gamma^d$.

$$\gamma = [\gamma^1, \dots, \gamma^N] \tag{16}$$

2. М-шаг

$$a(q; c) \propto \sum_d \sum_t \xi_t^d(q; c) \tag{17}$$

$$P(\{Y^1 \dots Y^N\}|\Lambda) = \prod_d P(Y^d|\Lambda) \tag{18}$$

## 2.5. Сравнение

### Критерий Акаике

Чем больше параметров у модели, тем лучше она подстраивается под данные, и тем проще переобучается. Поэтому, при сравнении моделей, обученных на одних и тех же данных, со схожим правдоподобием, предпочтительней будет та, которая проще. Конкретную величину, которую следует сравнивать для моделей, обученных на одинаковых данных, предлагает критерий Акаике (AIC)

$$AIC = 2k - 2 \log L \tag{19}$$

где $k$ — число степеней свободы или число параметров модели, $L$ — максимальное правдоподобие модели на заданной выборке. Чем $AIC$ меньше, тем модель лучше.

Число параметров для СММПП с $n$ скрытыми состояниями, $l$ контекстами, и Пуассоновскими испусканиями

$$\begin{aligned} k &= [\text{количество степеней свободы } A] + [\text{количество степеней свободы } B] \\ &+ [\text{количество степеней свободы } \pi] \\ &= l(n-1) \ + \ n \ + \ (l-1) \\ &= nl + n - 1 \end{aligned} \tag{20}$$

При $n = 2, k = 2l + 1$

Для СММ порядка $m$, $l = 2^m$, поэтому $k = 2^{m+1} + 1$.

**FDR, FNR**

Пусть гипотеза *H0* соответствует состоянию «0» (отсутствие взаимосвязи с белком), *H1* — отвержение *H0*. Тогда ошибки первого, второго рода характеризуются величинами $FDR$ — математическое ожидание доли ложных единиц среди всех предсказанных единиц (false discovery rate), $FNR$ — математическое ожидание доли ложных нулей среди всех предсказанных нулей (false non-discovery rate) [6].

$$FDR = \mathbb{E}\frac{FP}{P}, \quad FNR = \mathbb{E}\frac{FN}{N} \tag{21}$$

где $FP$ (False Positive) — количество неправильно идентифицированных единиц (количество состояний 0, предсказанных как 1), $P$ (Positive) — количество предсказанных единиц, $FN$ (False Negative) — количество неправильно идентифицированных нулей, $N$ (Negative) — количество предсказанных нулей.

Сами величины $FDR$, $FNR$ напрямую оценить трудно, поэтому далее будем рассматривать их верхние оценки $mFDR$ (marginal false discovery rate), $mFNR$ (marginal false non-discovery rate).

$$mFDR = \frac{\mathbb{E}FP}{\mathbb{E}P}, \quad mFNR = \frac{\mathbb{E}FN}{\mathbb{E}N} \tag{22}$$

Оценки $m\hat{F}DR$, $m\hat{F}NR$:

$$m\hat{F}DR = \frac{\sum_t I[x_t]P(x_t = 0)}{\sum_t I[x_t]} \tag{23}$$

$$m\hat{F}NDR = \frac{\sum_t I[x_t = 0]P(x_t = 1)}{\sum_t I[x_t = 0]} \tag{24}$$

Контроль $mFDR$ будем осуществлять с помощью статистики $LIS$, как это делают в работе [6]. Для контроля $mFDR$ величиной $\alpha$, состояние «1» присваивается позициям, соответствующим первым $k$ позициям в отсортированном по убыванию списке

величин $LIS_t$, где $LIS_t = P(x_t = 0|\text{взаимосвязь есть}) = I[x_t = 0]P(x_t = 1)$ — (local index of significance),

$$k = max\{i : \frac{1}{i}\sum_{t=0}^{i} LIS_t < \alpha\} \tag{25}$$

# 3. Реализация

Общий алгоритм обучения скрытой Марковской модели переменного порядка был реализован на языке программирования Python версии 3.4.

Python является выразительным, но местами медленным языком программирования, поэтому критические по производительности места (E-шаг) были перенесены на Cython. Язык Cython — расширение языка Python, транслирующее код в язык Си. Cython поддерживат опциональную типизацию и имеет эффективный интерфейс для работы с массивами NumPy.

С использованием библиотеки Joblib было распараллелено выполнение E-шага на нескольких выборках по потокам. Для эффективной работы с матрицами были использованы библиотеки NumPy, SciPy. Для отрисовки деревьев использовалась библиотека Pygraphviz. Все графики строились с помощью библиотеки Matplotlib.

Проект доступен по следующей ссылке: https://github.com/atanna/hmm

# 4. Применение

## 4.1. Применение к симулированным данным

Проверка работы алгоритма обучения происходила в несколько этапов:

1. генерация параметров $\Lambda$ начальной модели СММПП;
2. порождение нескольких выборок $Y$ из заданной модели;
3. нахождение новых параметров $\hat{\Lambda}$ путем обучения модели на порожденных выборках;
4. сравнение полученных параметров $\hat{\Lambda}$ с реальными $\Lambda$, подсчет абсолютного среднего отклонения параметров реальной модели от предсказанных параметров.

Ниже проиллюстрирована работа алгоритма на простых моделях: Пуассоновская смесь (СММ нулевого порядка) и СММ (СММ первого порядка). Также проиллюстрирован более интересный случай: СММПП, не являющейся СММ фиксированного порядка.

Для каждого теста было сгенерировано по 100 выборок длиной 5000. Значения параметров алгоритма обучения были выбраны следующими: максимально-возможный порядок $m = 4$, порог для обрезания $\varepsilon_{prune} = 0.007$, порог для остановки EM $\varepsilon_{EM} = 0.01$

Были посчитаны абсолютное среднее отклонение (MAE — Mean Absolute Error) полученных распределений переходов от реальных распределений, отклонение параметров Пуассоновского испускания $\lambda$ от реальных, оценки $m\hat{F}DR$ , $m\hat{F}NR$ (формулы 23, 24, соответственно).

Средние значения этих величин, полученных обучением модели на разных выборках при фиксированной начальной модели, показаны в соответствующих таблицах (4.1.1, 4.1.2, 4.1.3).

### 4.1.1. Пуассоновская смесь

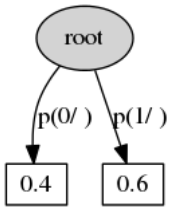
Модель смеси была выбрана с параметрами переходов, изображенных контекстным деревом на рисунке 3а и параметрами испусканий

$$B = \{Poisson(\lambda = 2), Poisson(\lambda = 10)\}$$

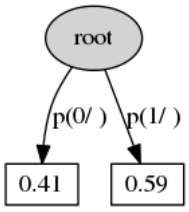
.

Сравнение полученных в ходе обучения параметров с реальными, $m\hat{F}DR$, $m\hat{F}NR$ показаны в таблице 4.1.1. Пример предсказанного дерева изображен на рисунке 3б.

| | $MAE(A, \hat{A})$ | $MAE(\lambda, \hat{\lambda})$ | $m\hat{F}DR$ | $m\hat{F}NR$ |
|---|---|---|---|---|
| *Среднее по тестам* | 0.07 | 0.21 | 0.01 | 0.02 |




(а) Реальное дерево




(б) Предсказанное дерево

Рис. 3: Пуассоновская смесь

### 4.1.2. CMM

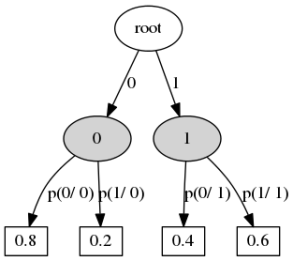
Модель CMM была выбрана с параметрами переходов, изображенных контекстным деревом на рисунке 4а и параметрами испусканий

$$B = \{Poisson(\lambda = 1), Poisson(\lambda = 8)\}$$

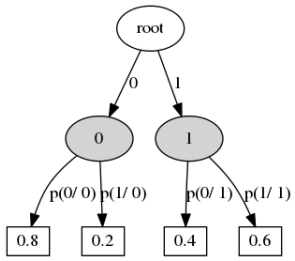
.

Сравнение полученных в ходе обучения параметров с реальными, $m\hat{F}DR$, $m\hat{F}NR$ показаны в таблице 4.1.2. Пример предсказанного дерева изображен на рисунке 4б.

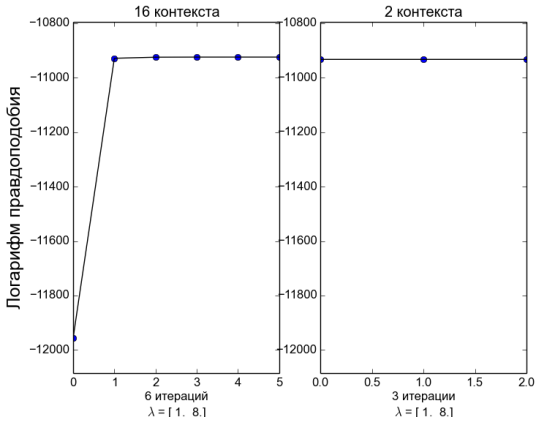
| | $MAE(A, \hat{A})$ | $MAE(\lambda, \hat{\lambda})$ | $m\hat{F}DR$ | $m\hat{F}NR$ |
|---|---|---|---|---|
| *Среднее по тестам* | 0.09 | 0.21 | 0.02 | 0.02 |




(а) Реальное дерево




(б) Предсказанное дерево




(в) График обучения

Рис. 4: CMM

На рисунке 4в изображен график обучения. Каждая EM-часть выделена контуром, сверху которого написано число контекстов на момент обучения, снизу количество итераций в этой части и параметры распределения испусканий, полученных на последней итерации, внутри — график логарифма правдоподобия по итерациям EM. На такой схеме видно, как сначала алгоритм 6 итераций EM обучался на 16 контекстах, после чего дерево подрезалось до 2 контекстов. Следующему EM не удалось значимо увеличить правдоподобие модели, поэтому на третей итерации он закончил работу. Далее дерево не удалось подрезать, поэтому весь алгоритм закончил свою работу (это видно из отсутствия следующего контура под график EM).

### 4.1.3. СММПП, не являющаяся СММ фиксированного порядка

Модель СММПП была выбрана с параметрами переходов, изображенных контекстным деревом на рисунке 5а и параметрами испусканий

$$B = \{Poisson(\lambda = 3), Poisson(\lambda = 15)\}$$

.

Сравнение полученных в ходе обучения параметров с реальными,$m\hat{F}DR$, $m\hat{F}NR$ показаны в таблице 4.1.3. Пример предсказанного дерева изображен на рисунке 5б. На рисунке 5в представлен график обучения модели на одном из тестов.

| | $MAE(A, \hat{A})$ | $MAE(\lambda, \hat{\lambda})$ | $m\hat{F}DR$ | $m\hat{F}NR$ |
|---|---|---|---|---|
| *Среднее по тестам* | 0.11 | 0.20 | 0.02 | 0.02 |

## 4.2. Применение к реальным данным

Для оценки модели на реальных данных использовались данные из проекта ENCODE (ENCyclopedia of DNA Elements). В качестве изучаемого белка рассматривался гистон H3 с ацетилированным лизином в 27-й позиции. Исследуемые клетки — эмбриональные стволовые клетки человека [1]. Размер окна был выбран равным 200 п.н.

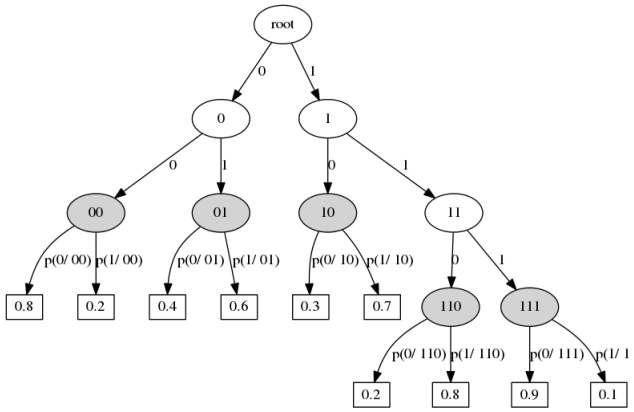
В качестве выборок брались ненулевые участки массива, полученного после деления результата эксперимента ChIP-seq на окна.

Ниже приведены результаты обучения на данных четвертой хромосомы (сумма длин обучающих выборок $\sim 10^5$).

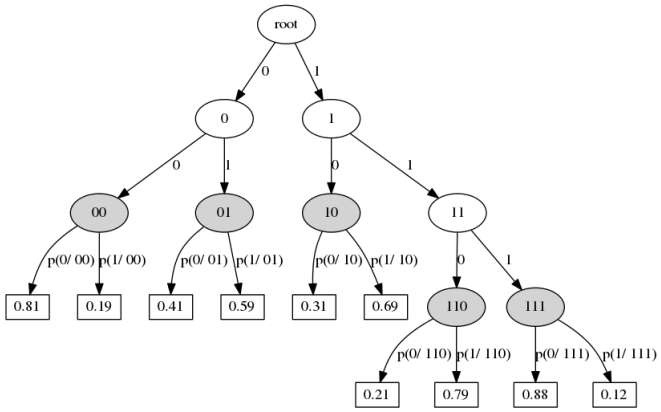
Параметры обучения стояли следующими: $m = 5$, $\varepsilon_{prune} = 0.04$, $\varepsilon_{em} = 0.05$

Из графика обучения (рис. 6а) видно, как сначала алгоритм 12 итераций EM обучался на 32 контекстах, потом подрезал дерево до 5 контекстов. После чего ни обучение, ни подрезание не дало результатов, поэтому, алгоритм закончил работу.

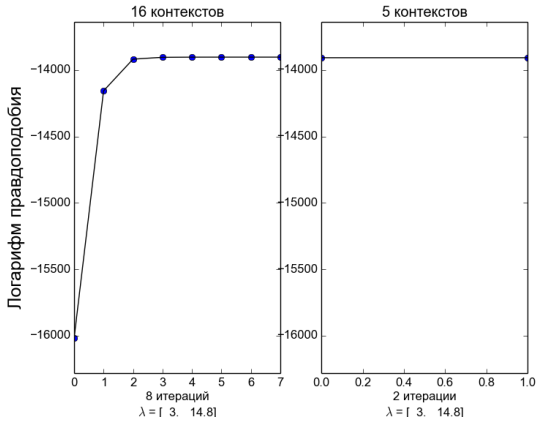
Полученное контекстное дерево переходов для скрытого слоя состояний, отвечающих за ДНК-белковую связь проиллюстрировано на рисунке 6б. Из него видно каким




(а) Реальное дерево



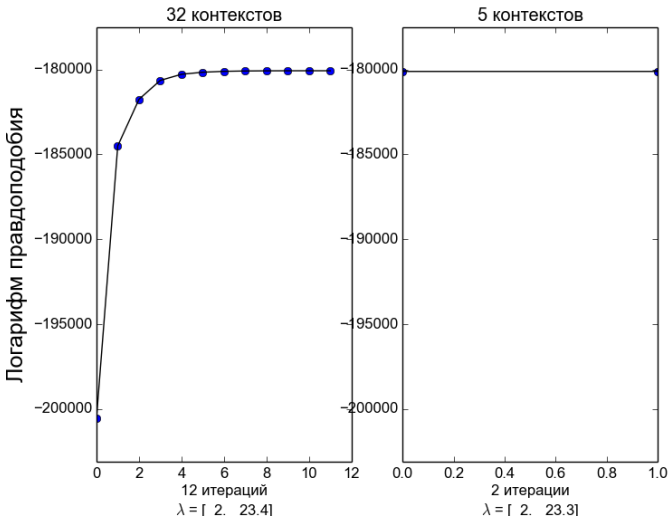

(б) Предсказанное дерево



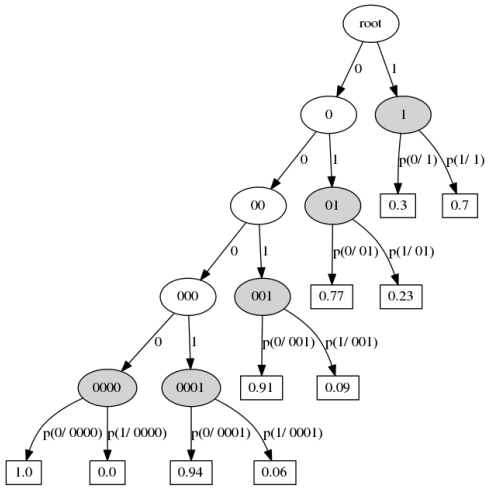

(в) График обучения

Рис. 5: СММПП, не являющаяся СММ фиксированного порядка

образом наличие взаимодействия с белком в фиксированном окне генома определяется взаимодействиями в предшествующих окнах.

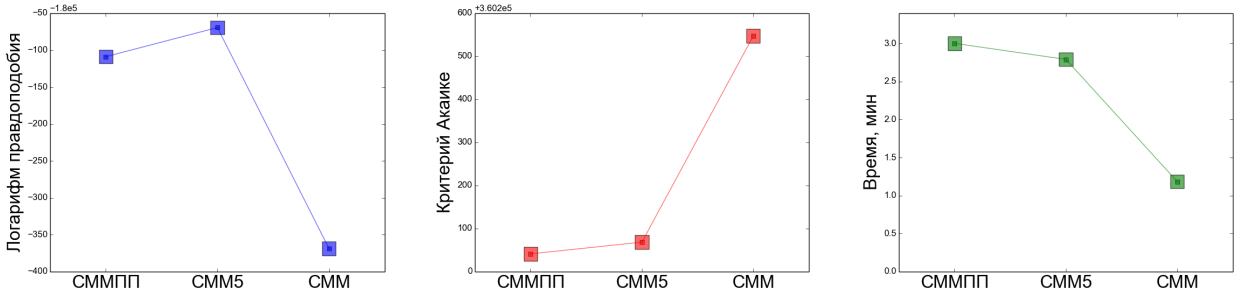



(а) График обучения




(б) Контекстное дерево

Рис. 6: СММПП на реальных данных




(а) Сравнение логарифма правдоподобия

(б) Сравнение критерия Акаике

(в) Сравнение времени обучения

Рис. 7: Сравнение моделей СММПП, СММ5 (СММ 5-го порядка, соответствует дереву, с которого начиналось обучение СММПП), СММ (СММ 1-го порядка, именно его чаще всего используют для анализа данных ChIP-seq)

Ниже представлено сравнение логарифма правдоподобия (рис. 7а), критерия Акаике (рис. 7б) и времени обучения (рис. 7в) для моделей СММПП, СММ5 и СММ.

По критерию Акаике (рис. 7б) выигрывает СММПП (напомним, что данный критерий, чем меньше, тем лучше).

СММ5 имеет лучшее среди этих трех моделей правдоподобие (рис. 7а), однако ее губит большое количество параметров. СММ имеет меньшее среди данных моделей количество параметров, однако ее правдоподобие совсем невелико.

Время обучения СММПП (рис. 6б) дает схожий результат с СММ5, немного ей уступая. СММ, в силу простоты структуры, обучается быстрее всех.

# Заключение

В ходе работы были решены поставленные задачи:

1. реализована СММПП, подходящая под данные ChIP-seq;
2. проведен анализ эффективности работы СММПП на синтетических данных;
3. проведено сравнение СММПП с СММ порядка 1 и СММ порядка 5 на данных проекта ENCODE, согласно критерию Акаике СММПП является более подходящей моделью.

# Список литературы


[1] Broad Bradley Bernstein. Experiment summary for encsr000anp, 2011.

[2] P Bühlmann and AJ Wyner. Variable length Markov chains. *The Annals of Statistics*, 27(2):480–513, April 1999.

[3] Thierry Dumont. Context tree estimation in variable length hidden Markov models. *IEEE Transactions on Information Theory*, 60:3196–3208, 2014.

[4] Lawrence Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[5] Lynch AG Tavare S Spyrou C, Stark R. BayesPeak: Bayesian analysis of ChIP-seq data. 2009.

[6] Wenguang Sun and T. Tony Cai. Large-scale multiple testing under dependence. *J. R. Statist. Soc. B*, pages 393–424, 2009.

[7] Y Wang, Lizhu Zhou, and Jianhua Feng. Mining complex time-series data by learning Markovian models. In *International Conference on Data Mining*, 2006.

[8] Jérôme Eeckhoute David S Johnson Bradley E Bernstein Chad Nusbaum Richard M Myers Myles Brown Wei Li Yong Zhang, Tao Liu Clifford A Meyer and X Shirley Liu. Model-based Analysis of ChIP-Seq (MACS). 2008.