



«Анализ транскриптомных данных»

# Лекция #2.

# **Выравнивания и псевдовыравнивания.**

# **Подсчёт экспрессии**

**Серёжа Исаев**

аспирант **ФБМФ МФИ**  
аспирант **MedUni Vienna**

# Содержание курса

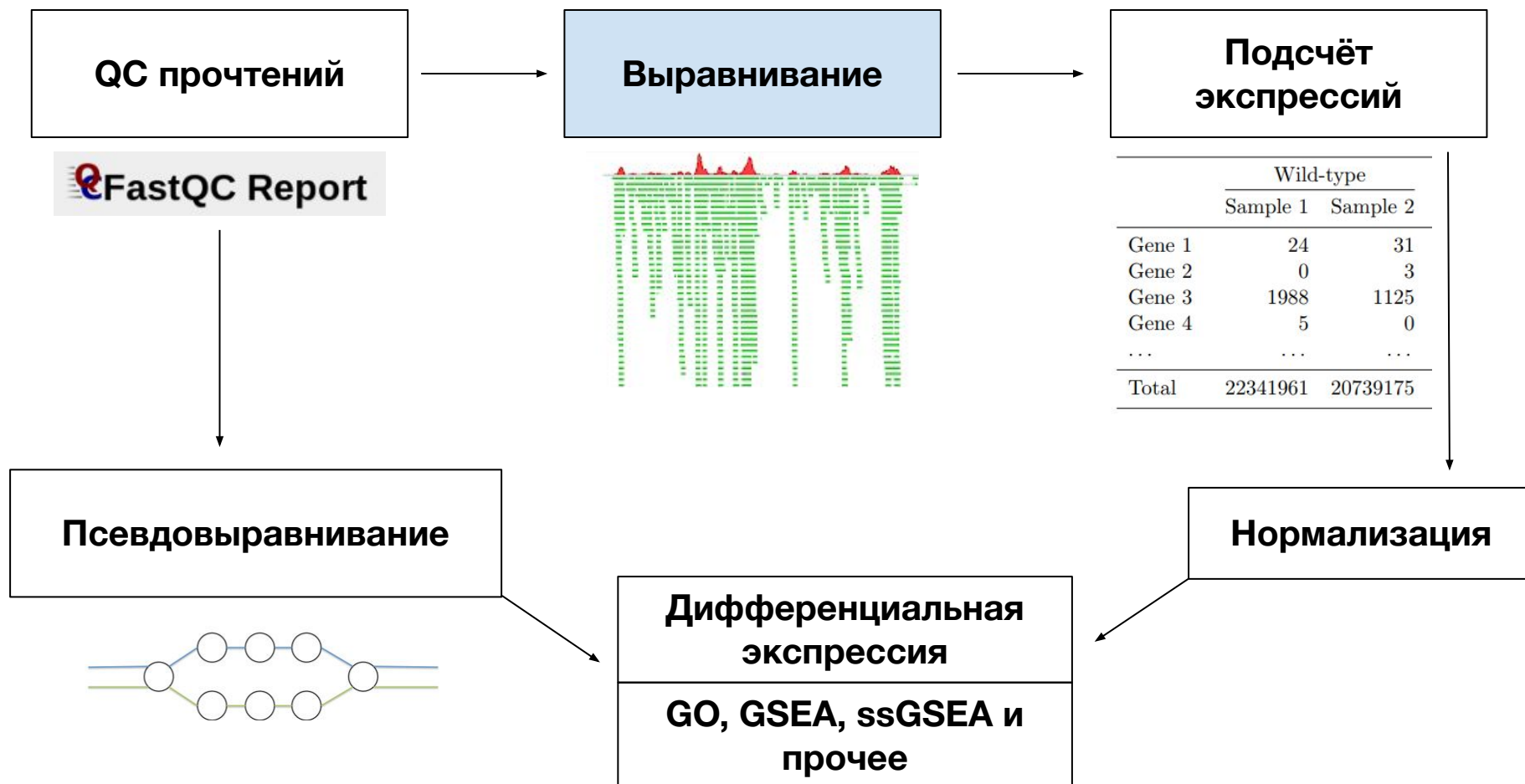
## 1. Bulk RNA-Seq:

- a. экспериментальные подходы,
- b. выравнивания и псевдовыравнивания,**
- c. анализ дифференциальной экспрессии,
- d. функциональный анализ;

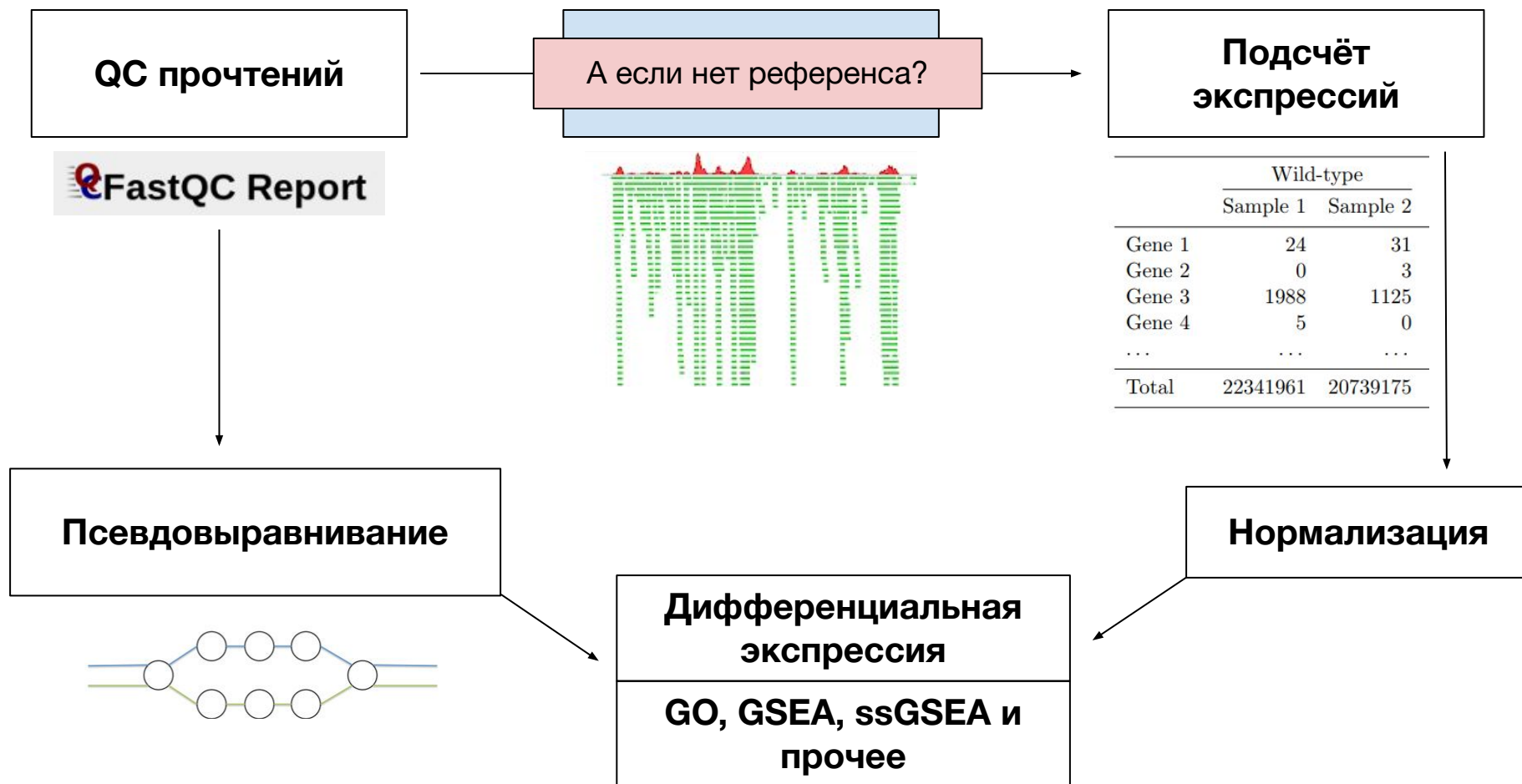
## 2. Single-cell RNA-Seq:

- a. экспериментальные подходы,
- b. отличия от процессинга bulk RNA-Seq,
- c. методы снижения размерности,
- d. кластера и траектории,
- e. мультимодальные омики одиночных клеток.

# Дорожная карта анализа RNA-Seq



# Дорожная карта анализа RNA-Seq



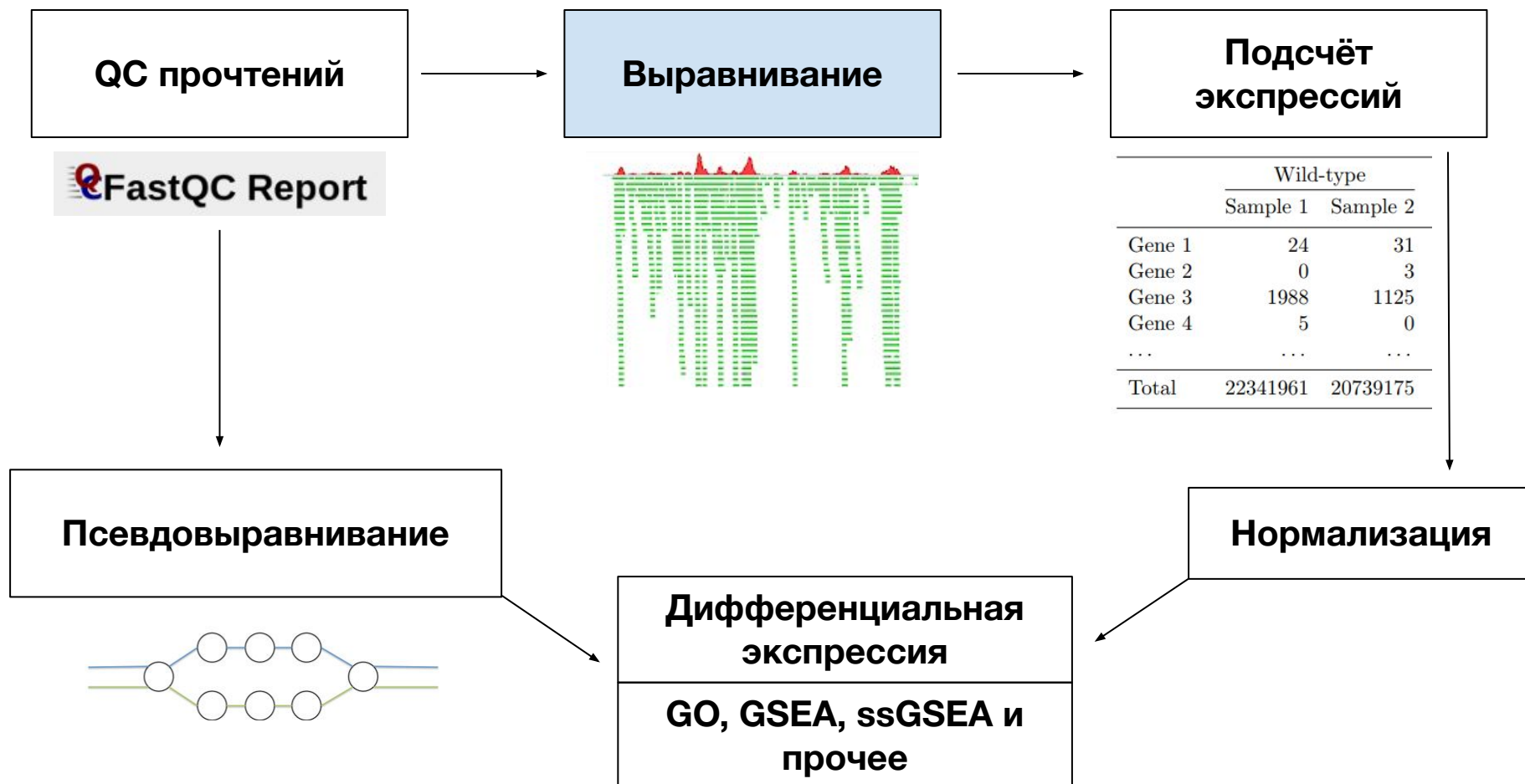
# Сборка транскриптома

Процедура относительно нетривиальная и состоит из нескольких шагов:

1. **Сборка** (SPAdes RNA, Trinity, ...) — строится граф из k-меров прочтений, в котором потом находятся пути, соответствующие транскриптам
2. **Очистка от контаминации** (MCSC, DeconSeq, ...) — различными эвристиками организмы очищаются от последовательностей, которые к ним примешались
3. **Проверка полноты сборки** (BUSCO, ...) — по поиску ортологов определяется, насколько “полно” представлены важнейшие группы генов

В контексте данного курса мы не будем глубоко погружаться в процесс сборки

# Дорожная карта анализа RNA-Seq



# Вообще-то **картирование**, а не выравнивание

Выравнивание — это процесс поиска лучшего (т.е. с наибольшим весом) сопоставления двух последовательностей

Поиск выравнивания — это очень **долгая** процедура (для интереса можете попробовать выравнивать 1000 ридов на геном человека и померять, сколько это займёт по времени)

Картирование — это (с некоторыми оговорками) лучшего **вхождения** одной последовательности в другую. Картировать можно сильно быстрее, чем выравнивать, а потому для NGS-экспериментов используют алгоритмы картирования

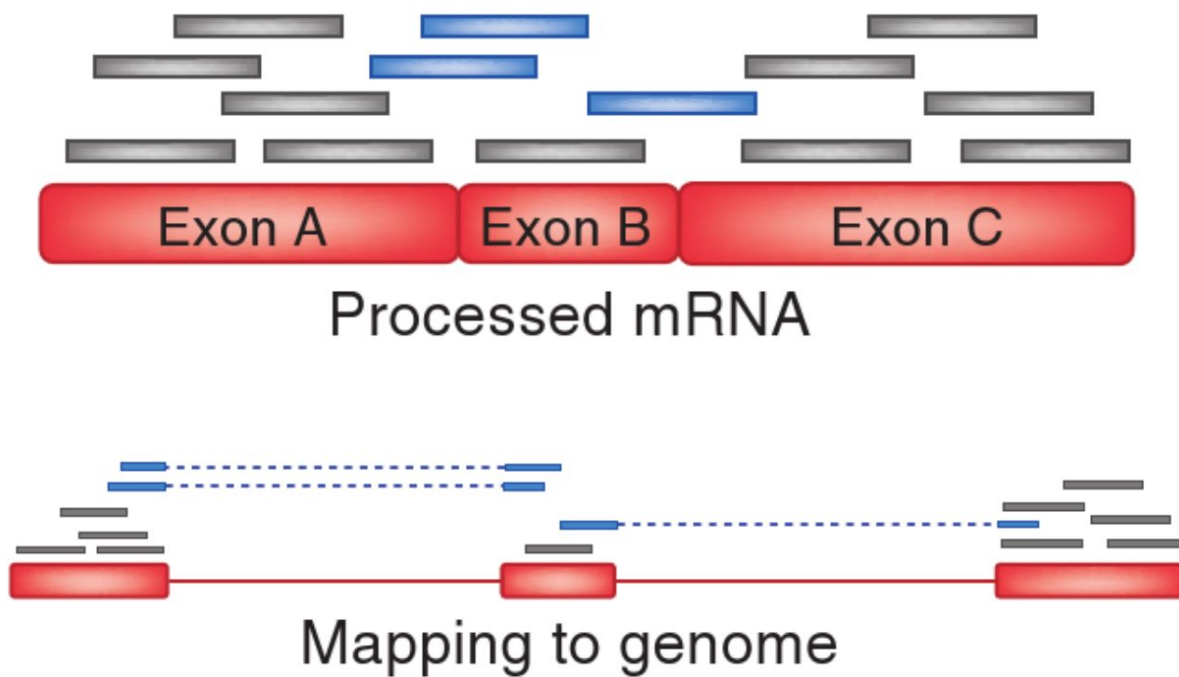
Для простоты будем называть это выравниванием, однако важно понимать, что процедуры разные

# На что выравнивать?

Выравнивать можно как на **референсный геном**, так и на **референсный транскриптом**

В случае выравнивания на транскриптом мы можем не засечь новые события сплайсинга

В случае выравнивания на геном наш алгоритм должен быть устойчив к большим gap'ам





# Главные программы для выравниваний

Для выравнивания прочтений без больших гэпов используют **Bowtie** или **BWA** (это актуально при работе с WES и WGS)

Для выравнивания прочтений с большими гэпами (результат сплайсинга) используют **STAR** (суффиксный массив) и **HISAT2** (bwt)

Name	Version	Mapping	Reference
Bowtie	2.2.6	Unspliced read aligner	<a href="#">[31]</a>
BWA	0.7.12-r1039	Unspliced read aligner	<a href="#">[33]</a>
TopHat	2.10	Spliced read aligner	<a href="#">[18]</a>
STAR	2.5.3	Spliced read aligner	<a href="#">[34]</a>
kallisto	0.43.1	pseudo-alignment	<a href="#">[35]</a>
Salmon	0.8.2	pseudo-alignment	<a href="#">[36]</a>

<https://doi.org/10.1371/journal.pone.0190152.t001>

# STAR

## BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 1 2013, pages 15–21  
doi:10.1093/bioinformatics/bts635

*Sequence analysis*

Advance Access publication October 25, 2012

### **STAR: ultrafast universal RNA-seq aligner**

Alexander Dobin<sup>1,\*</sup>, Carrie A. Davis<sup>1</sup>, Felix Schlesinger<sup>1</sup>, Jorg Drenkow<sup>1</sup>, Chris Zaleski<sup>1</sup>, Sonali Jha<sup>1</sup>, Philippe Batut<sup>1</sup>, Mark Chaisson<sup>2</sup> and Thomas R. Gingeras<sup>1</sup>

<sup>1</sup>Cold Spring Harbor Laboratory, Cold Spring Harbor, NY, USA and <sup>2</sup>Pacific Biosciences, Menlo Park, CA, USA

Associate Editor: Inanc Birol

Рекомендован ENCODE

Хорошо работает даже при больших отличиях от референса и прост в использовании

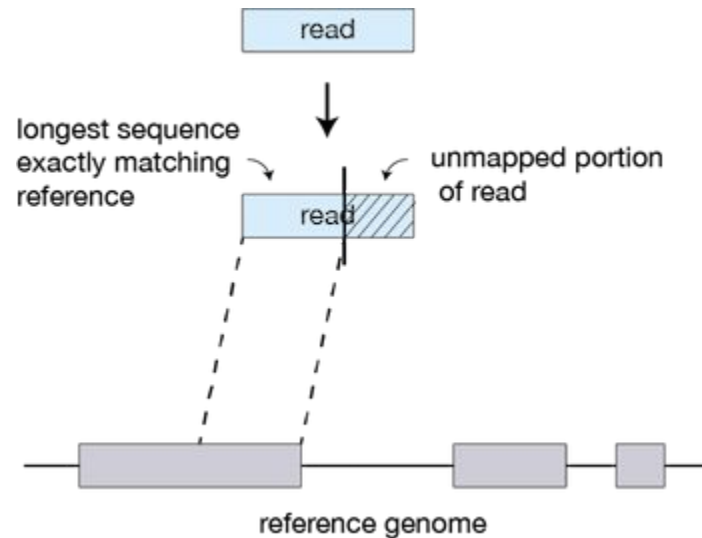
Работает относительно быстро

Требует большое количество RAM (десятки Gb — на ноутбуке не запустишь)

# Алгоритм STAR (1)

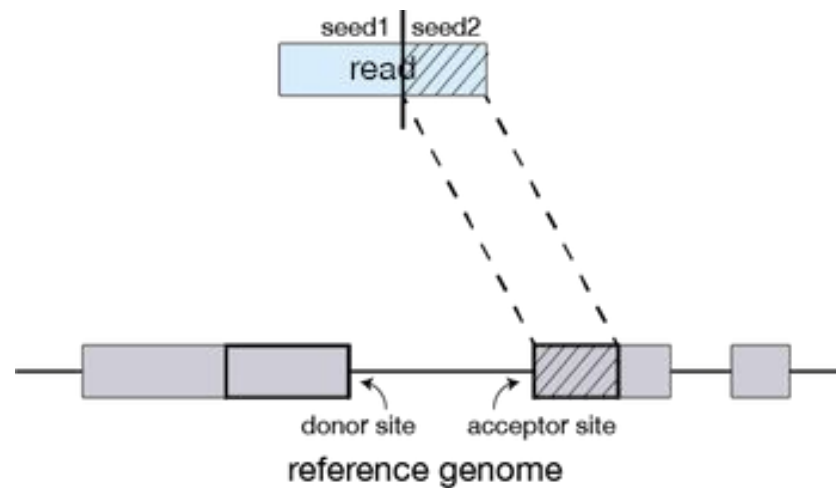
Сначала ищется самое большое подслово, которое совпадает с геномом

Поиск производится путём представления генома как несжатого суффиксного массива (поэтому поиск достаточно быстрый, однако требует достаточно много памяти)



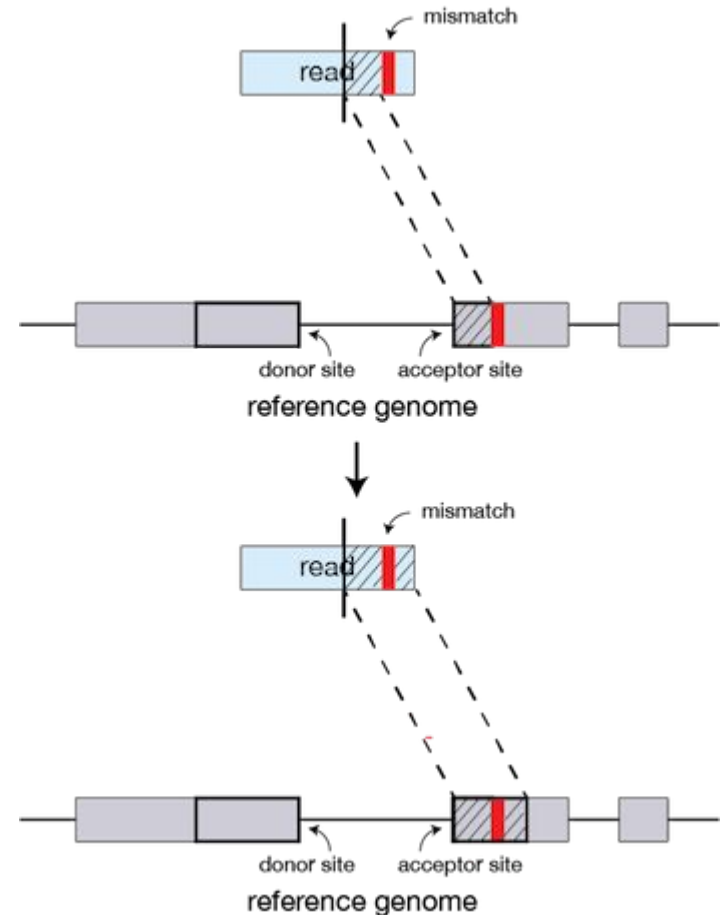
# Алгоритм STAR (2)

Потом STAR проводит аналогичную процедуру для оставшейся части прочтения



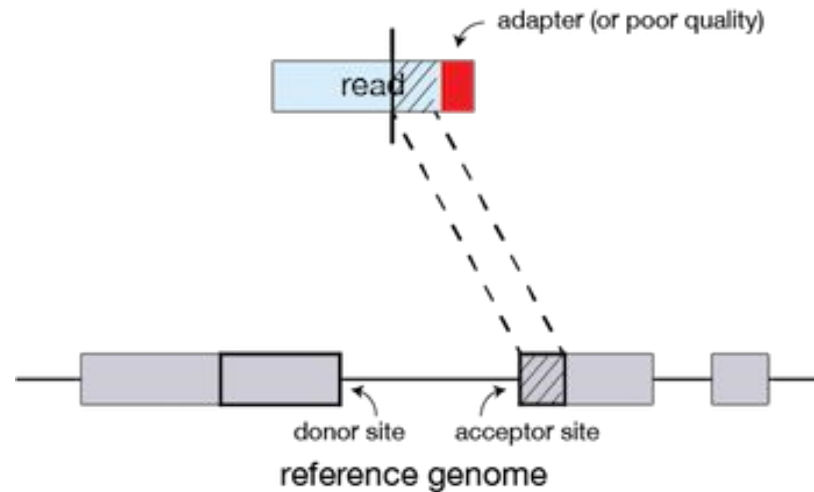
# Алгоритм STAR (3)

Если после второй итерации слово прочтение выравнивалось не полностью, то в таком случае будет допущено наличие SNV или вставки/делеции в продолжении прочтений

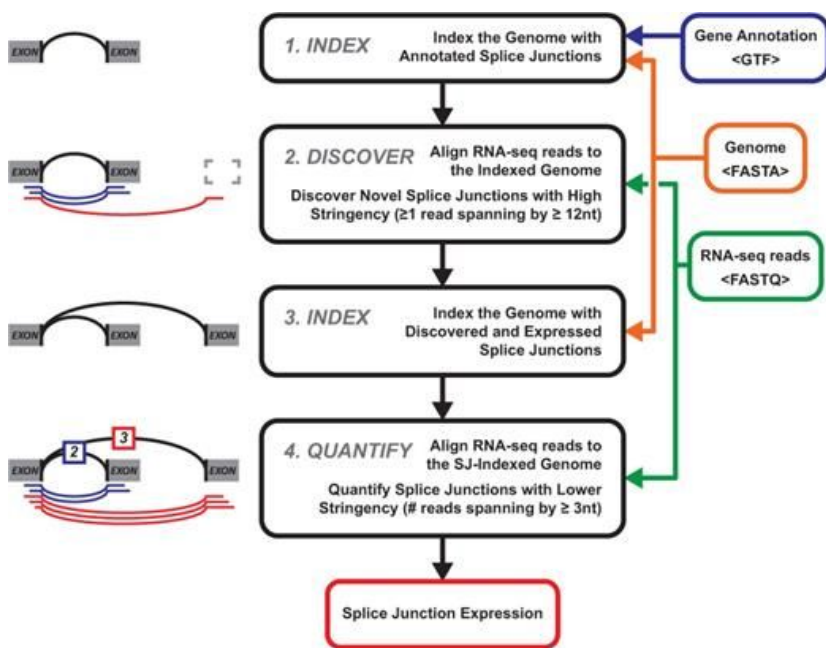


# Алгоритм STAR (4)

Если оставшаяся часть прочтения находится в начале или конце рида, а также не даёт хорошего выравнивания, то она будет считаться остатком адаптера секвенирования и просто отсекается



# Двухшаговое выравнивание при помощи STAR



Veeneman et al., **Bioinf**, 2016

У STAR есть режим `--twopassMode`, в котором выравнивание производится дважды

Логика заключается в том, что при первом выравнивании новые splice junctions могут быть не детектированы при достаточно малом покрытии

Однако если у вас есть несколько транскриптомов, то есть вероятность, что в каком-то они обнаружатся, и тогда можно по ним аннотировать новые события сплайсинга и подсчитать их число в остальных образцах

# Формат BAM / SAM

@HD VN:1.5 S0:coordinate											Header section
@SQ SN:ref LN:45											
r001	99	ref	7	30	8M2I4M1D3M	=	37	39	TTAGATAAAGGATACTG	*	Alignment section
r002	0	ref	9	30	3S6M1P1I4M	*	0	0	AAAAGATAAGGATA	*	
r003	0	ref	9	30	5S6M	*	0	0	GCCTAAGCTAA	* SA:Z:ref,29,-,6H5M,17,0;	
r004	0	ref	16	30	6M14N5M	*	0	0	ATAGCTTCAGC	*	
r003	2064	ref	29	17	6H5M	*	0	0	TAGGC	* SA:Z:ref,9,+,5S6M,30,1;	
r001	147	ref	37	30	9M	=	7	-39	CAGCGGCAT	* NM:i:1	

Optional fields in the format of TAG:TYPE:VALUE

QUAL: read quality; \* meaning such information is not available

SEQ: read sequence

TLEN: the number of bases covered by the reads from the same fragment. Plus/minus means the current read is the leftmost/rightmost read. E.g. compare first and last lines.

PNEXT: Position of the primary alignment of the NEXT read in the template. Set as 0 when the information is unavailable. It corresponds to POS column.

RNEXT: reference sequence name of the primary alignment of the NEXT read. For paired-end sequencing, NEXT read is the paired read, corresponding to the RNAME column.

CIGAR: summary of alignment, e.g. insertion, deletion

MAPQ: mapping quality

POS: 1-based position

RNAME: reference sequence name, e.g. chromosome/transcript id

FLAG: indicates alignment information about the read, e.g. paired, aligned, etc.

QNAME: query template name, aka. read ID



# SNP calling в RNA-Seq

В целом, SNP calling из RNA-Seq делать можно, для этого рекомендуют различные подходы, которые не всегда лучшие для WES (см. <https://doi.org/10.1186/s13059-019-1863-4>), однако **лучше** для того, чтобы определять однонуклеотидные полиморфизмы, **использовать геномные или экзомные секвенирования**

Сложность определения замен заключается в нескольких деталях:

1. присутствуют ошибки секвенирования,
2. присутствует аллель-специфическая экспрессия,
3. покрытие различных позиций отличается очень сильно

# RSeQC

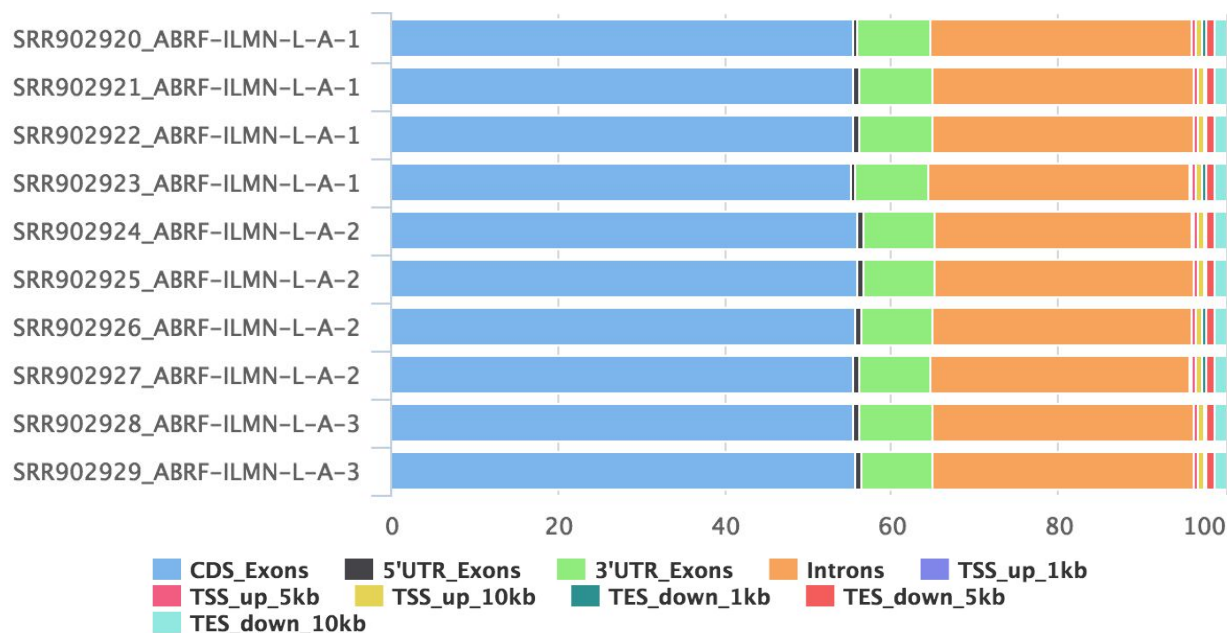
Существует отдельная стадия контроля качества для выравниваний, она обеспечивается при помощи пакета RSeQC

Очень важно обращать внимание на качество выравнивания, потому что оно говорит о качестве образца (а не о качестве секвенирования!) — в результате FastQC может показать хорошие метрики, а сам образец будет очень низкого качества (с деградировавшей РНК и проч.)

Отчёт RSeQC может быть включен в общий QC отчёт MultiQC

# Распределение прочтений по элементам генома

В образцах, которые вы анализируете, должно быть сравнимое распределение ридов, картирующихся на схожие элементы генома (экзоны / интроны / ...)

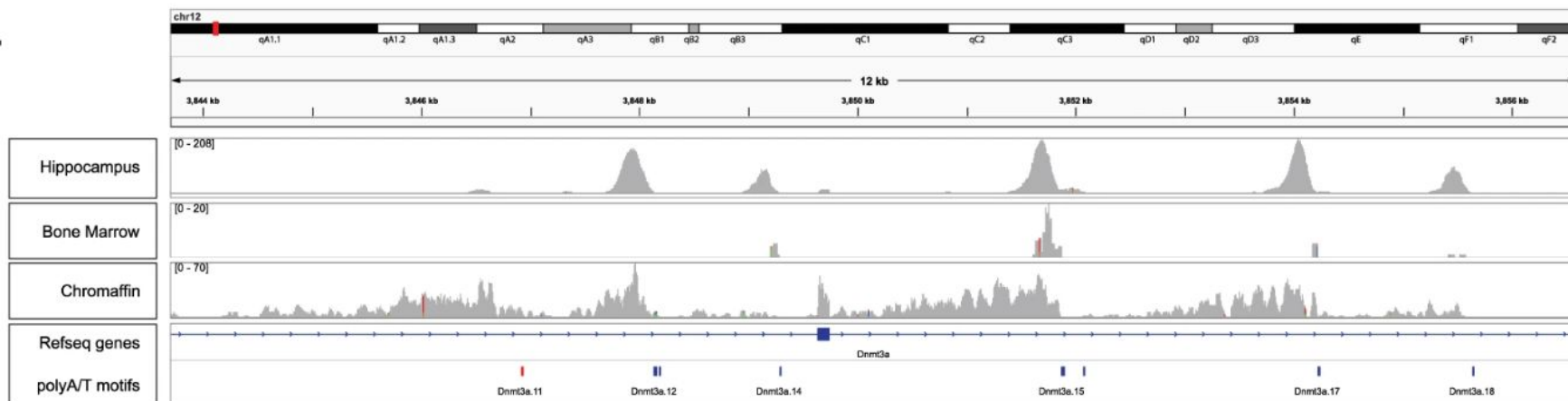


# Почему какие-то прочтения падают на интроны?

По всей видимости, oligo(dT)-праймеры могут отжигаться не только на polyA-хвост мРНК, но и на некоторые polyA-мотивы интронов

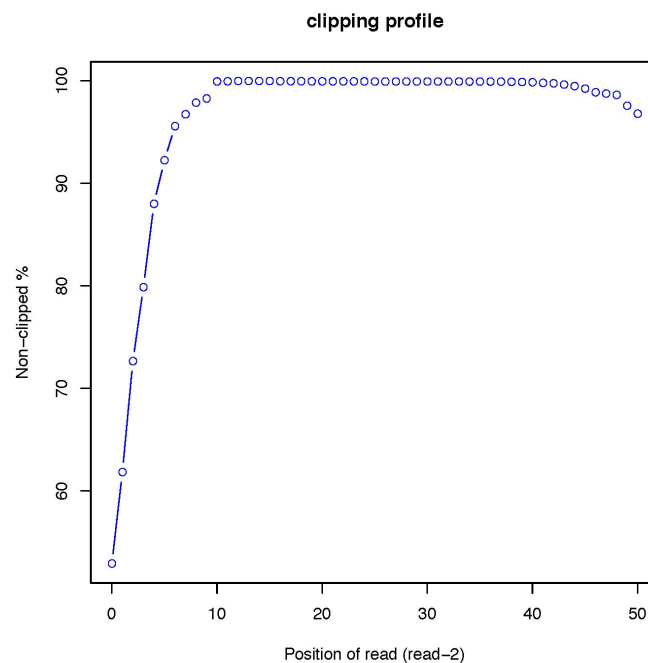
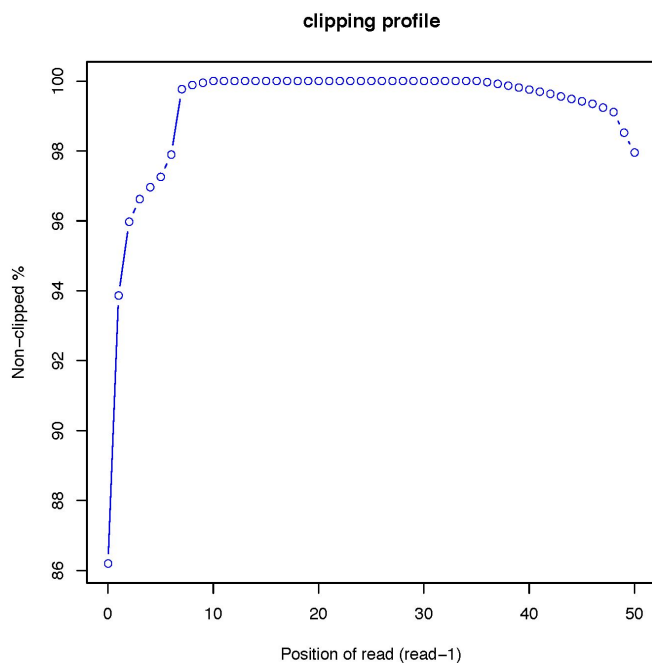
Этим объясняется относительно высокое количество интронных последовательностей в результатах секвенирования

С.

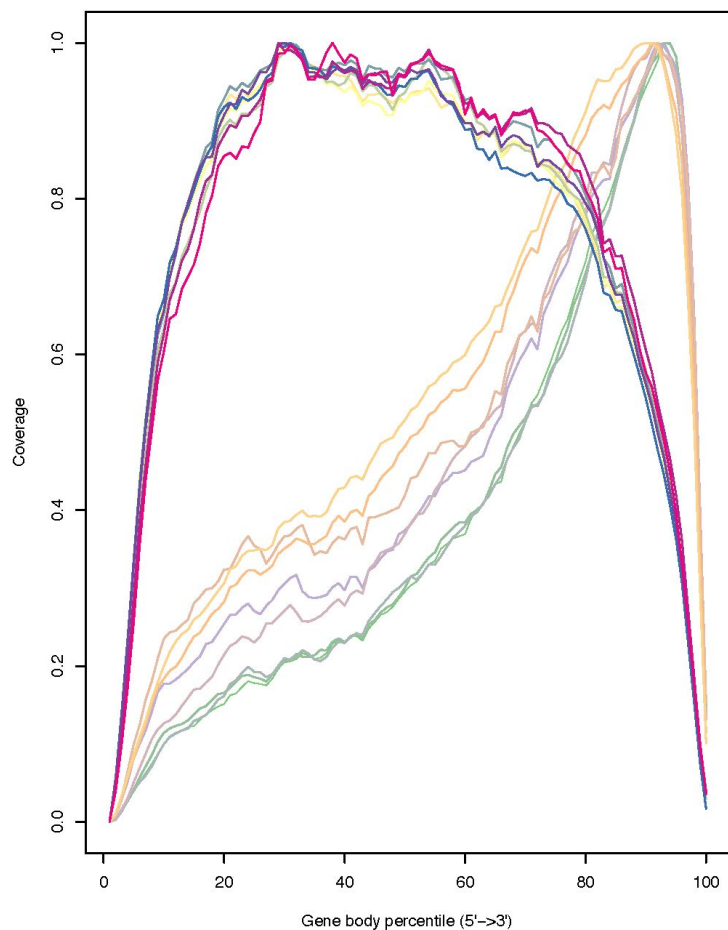


# Clipping profile

Вас также должно смутить большие участки адаптерных последовательностей (возможно, у вас какие-то проблемы с подготовкой библиотеки — например, слишком маленький размер вставки)



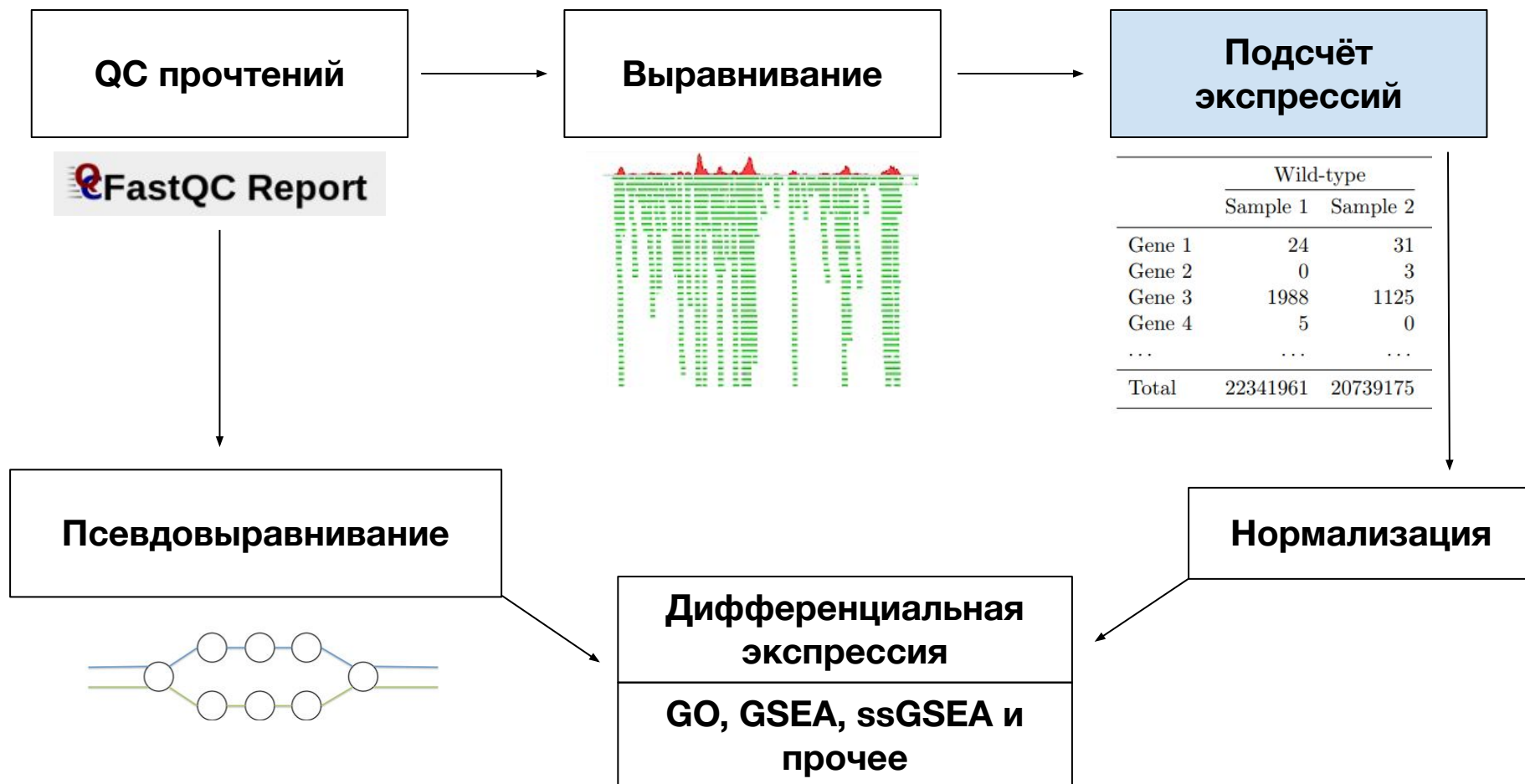
# Gene Body Coverage



Эта метрика уже обсуждалась ранее

На иллюстрации справа распределение покрытия по длине гена. Предположите, какие из образцов FFPE, а какие — FF?

# Дорожная карта анализа RNA-Seq



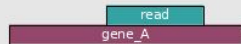
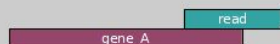

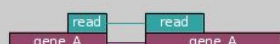



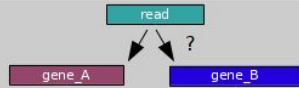
# HTSeq / featureCounts (exon union)

HTSeq и featureCounts — самые простые программы, при помощи которых подсчитывают экспрессию

В основе их работы лежит простая логика: если рид ложится на ген, то мы даём +1 к экспрессии гена

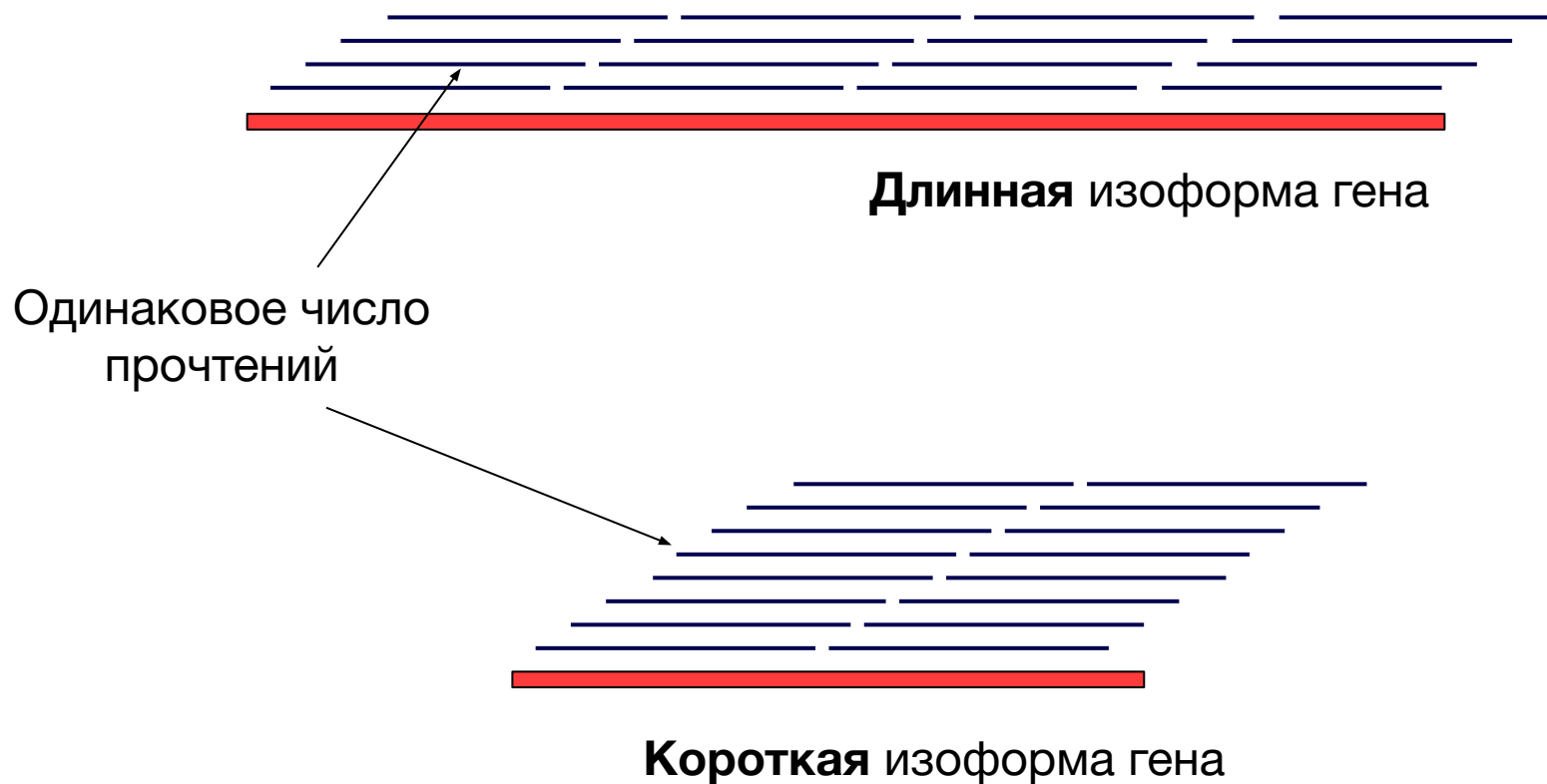
featureCounts по умолчанию вшит в STAR

Liao et al., **Bioinformatics**, 2014 and  
Andres et al., **Bioinformatics**, 2015

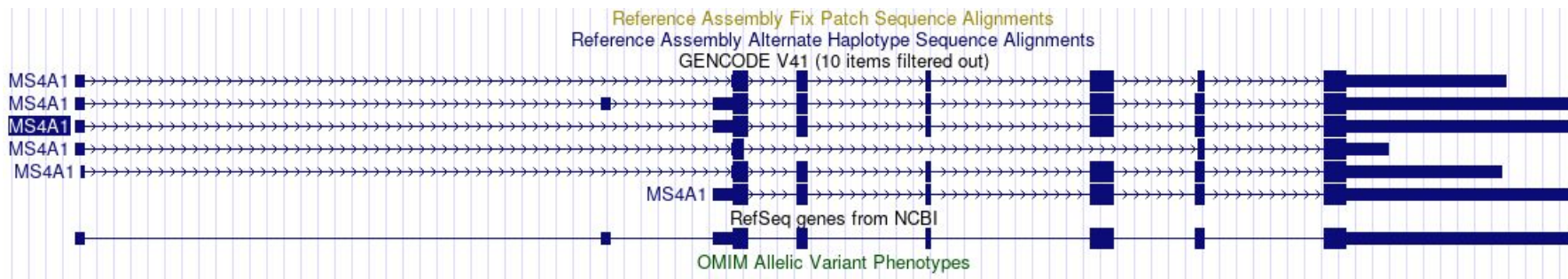
	union	intersection_strict	intersection_nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous (both genes with --nonunique all)	gene_A	gene_A
	ambiguous (both genes with --nonunique all)		
	alignment_not_unique (both genes with --nonunique all)		



# Подсчёт экспрессии различных изоформ



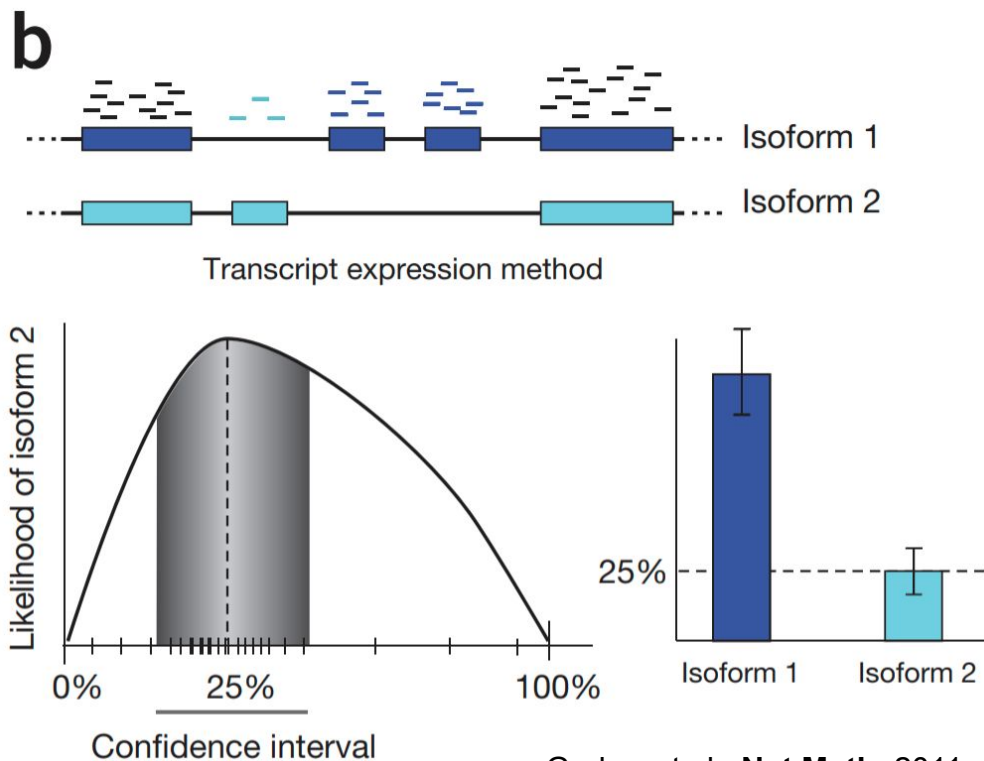
# Разный размер изоформ



На рисунке выше пример гена *MS4A1*, который кодирует белок *CD20*. Как понять, с какой изоформы пришли прочтения?

Можно попробовать найти такое отношение количества изоформ транскриптов, которое бы с наибольшей вероятностью порождало наблюдаемое распределение прочтений по разным участкам транскриптома, то есть **максимально правдоподобное отношение количества изоформ**

# RSEM



Garber et al., **Nat Meth**, 2011

RSEM (RNA-Seq by Expectation Maximization) оптимизирует правдоподобие отношения изоформ, опираясь на покрытие гена — Li and Dewey, **BMC Bioinf**, 2011

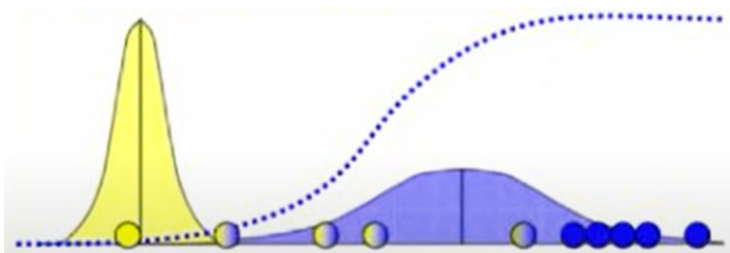
Является стандартом для определения различных изоформ

# ЕМ-алгоритм

## Шаг 1: Expectation



## Шаг 2: Maximization



до сходимости

# Шаг 1: Expectation

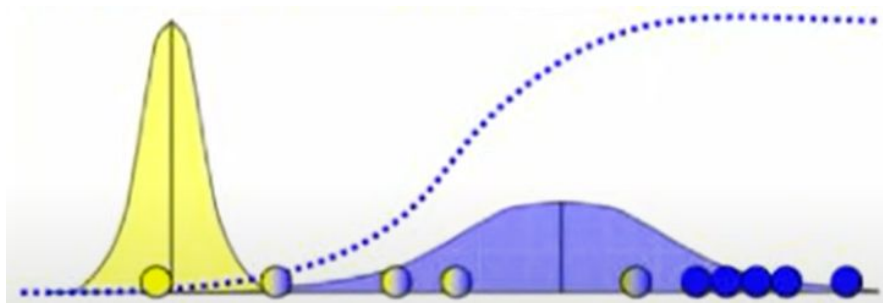


$$P(x_i|b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \exp\left(-\frac{(x_i - \mu_b)^2}{2\sigma_b^2}\right)$$

$$b_i = P(b|x_i) = \frac{P(x_i|b)P(b)}{P(x_i|a)P(a) + P(x_i|b)P(b)}$$

$$a_i = P(a|x_i) = 1 - b_i$$

## War 2: Maximization



$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \cdots + b_n x_n}{b_1 + b_2 + \cdots + b_n}$$

$$\sigma_b^2 = \frac{b_1(x_1 - \mu_b)^2 + \cdots + b_n(x_n - \mu_b)^2}{b_1 + b_2 + \cdots + b_n}$$

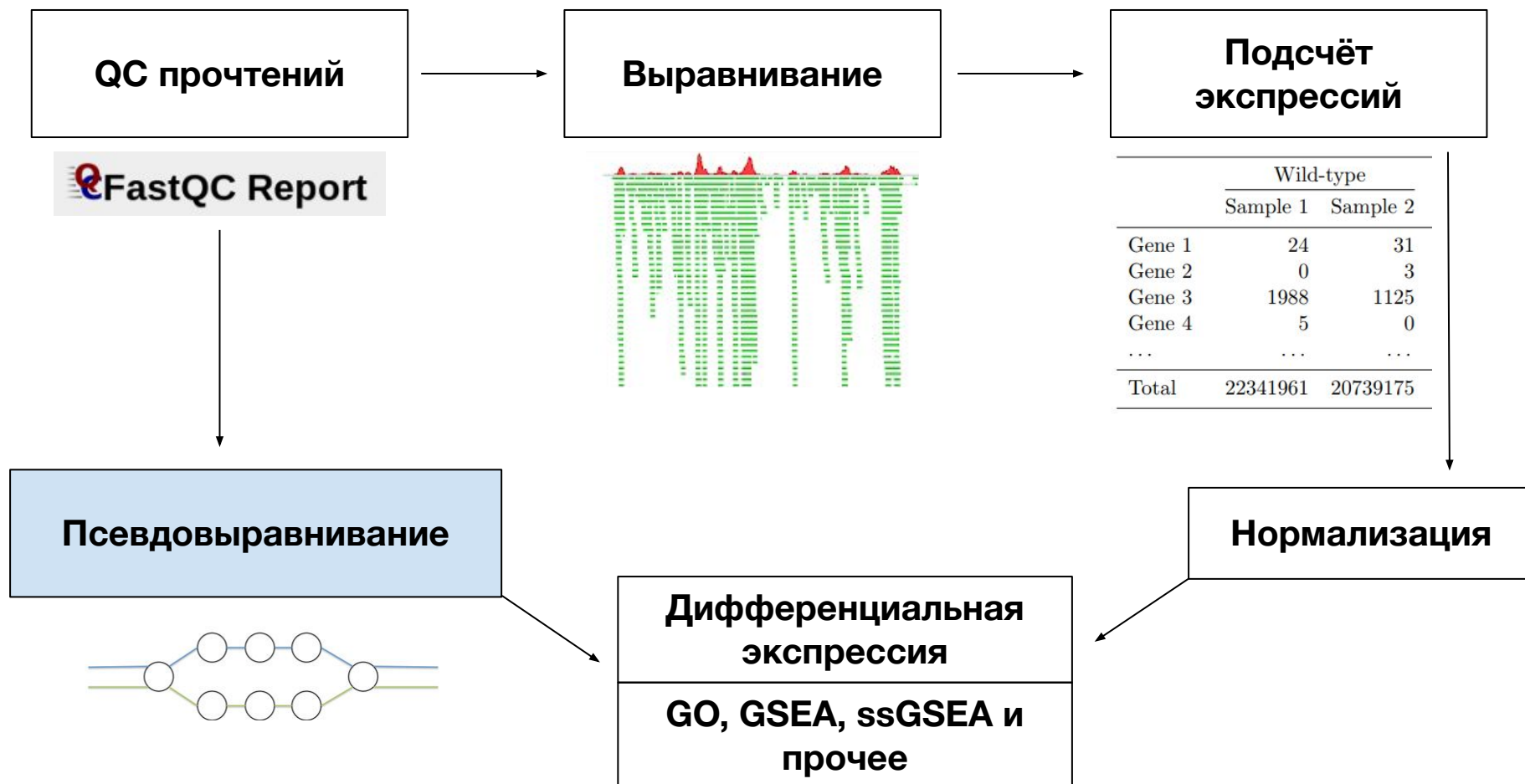
$$P(b) = \frac{b_1 + b_2 + \cdots + b_n}{n}$$

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \cdots + a_n x_n}{a_1 + a_2 + \cdots + a_n}$$

$$\sigma_a^2 = \frac{a_1(x_1 - \mu_a)^2 + \cdots + a_n(x_n - \mu_a)^2}{a_1 + a_2 + \cdots + a_n}$$

$$P(a) = 1 - P(b)$$

# Дорожная карта анализа RNA-Seq

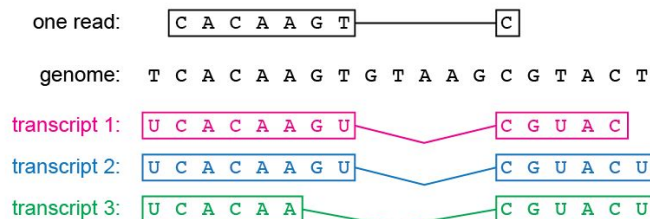


# kallisto

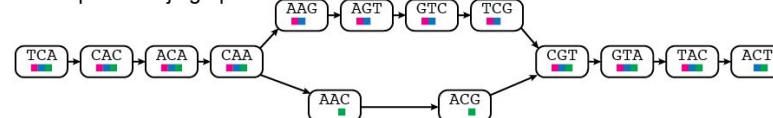
kallisto строит референсный  
окрашенный граф де Брёйна из k-  
меров транскриптома

Экспериментальные прочтения  
разбиваются на те же k-меры

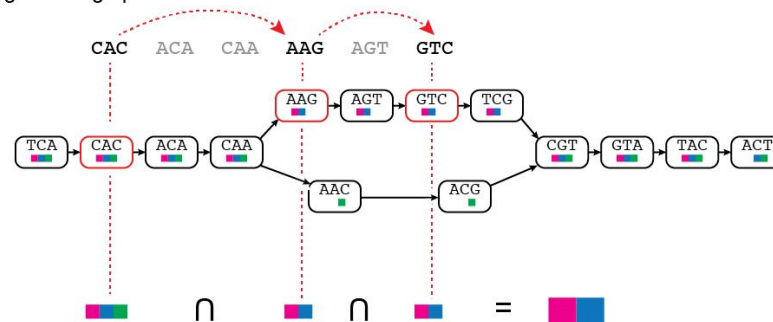
По найденным путям в графе при  
помощи ML находится наиболее  
правдоподобные каунты  
транскриптов



Colored transcript de Bruijn graph:



Matching read to graph:





# Бутстрэп kallisto

Так как kallisto **оценивает** экспрессии, а не напрямую физически высчитывает, мы можем оценить стабильность этой оценки при помощи бутстрэпа

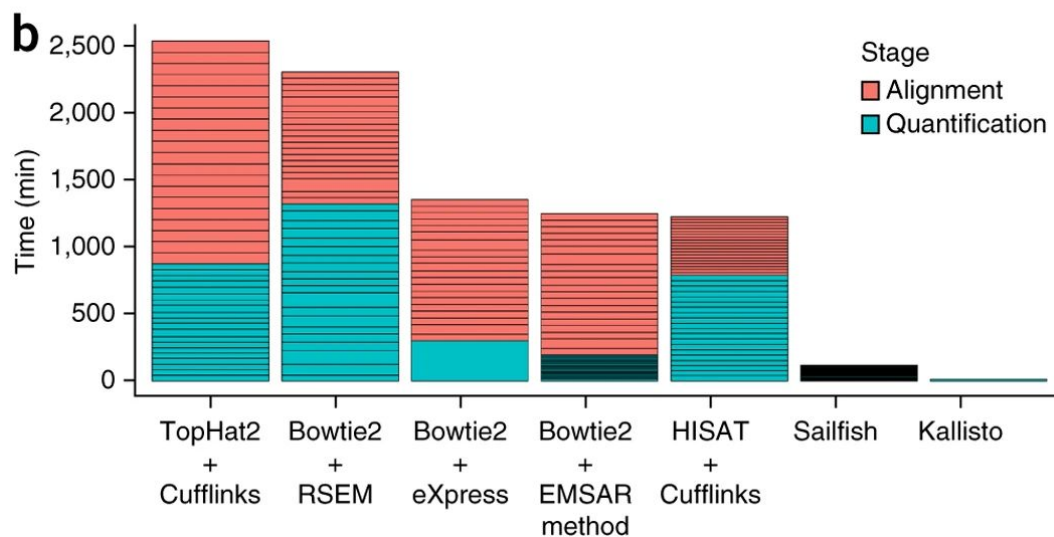
**Бутстрэп** — это процедура, в ходе которой новая выборка создаётся из элементов изначальной выборки с повторениями

Бутстрэп оценивает, насколько стабильным при малом возмущении будет реконструкция параметра (экспрессии генов)

В дальнейшем это можно использовать в специальных пайплайнах (например, в Sleuth), но в основном этим инструментом не пользуются

# Время работы kallisto

kallisto многократно превосходит по скорости большинство других подходов к подсчёту экспрессии



Bray et al., **Nat Biotechnol**, 2016

# Минусы kallisto

Не возвращает .bam-файл с выравниванием, а потому

1. нет возможность производить поиск замен,
2. нет возможности дополнительно до-анализировать датасет,
3. нет возможности искать новые сплайс-изоформы

Для работы с kallisto мы должны быть очень уверены в корректности референса