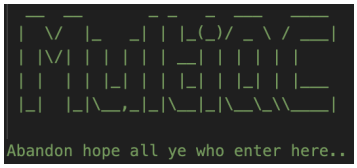
Сборка и анализ геномов

Демидов Иван, ФББ, практикумы 11-13, 15



Резюме: В ходе работы над этими практикумами были найдены и аннотированы варианты одного человека по данным экзомного секвенирования на примере пятой хромосомы, а также построен экспрессионный профиль на основании данных секвенирования РНК.

Часть I (11 практикум)

Задача практикума: подготовить необходимые файлы, изучить качество предложенных чтений и проиндексировать референс

1.1 Подготовка референса

Так как многие программы перед работой с большими файлами требуют предварительной индексации, первым делом была проиндексирована последовательность референса (пятой хромосомы человека).

Индексация для программы **hisat2** (картирование, см. часть 2):

```
hisat2-build chr5.fa chr5_indexed
```

`chr5.fa` – файл с последовательностью пятой хромосомы;
выходные данные - 8 файлов с расширением .ht2

Индексация с помощью **samtools**:

```
samtools faidx chr5.fa
```

На выходе получается текстовый файл `chr5.fa.fai`, состоящий из одной строки (так как во входном файле одна последовательность) со следующими столбцами:

`NAME`	**`5`**	Название последовательности
`LENGTH`	**`181538259`**	Длина последовательности, в нуклеотидах
`OFFSET`	`50`	“Координата” первого нуклеотида последовательности в файле в байтах начиная с 0
`LINEBASES`	`60`	Количество нуклеотидов в строке
`LINEWIDTH`	`61`	Количество байтов в строке (включая символ переноса строки)

1.2 Работа с чтениями ДНК

1) Мне достались чтения **SRR10720404**. Ниже приведена некоторая информация, полученная из NCBI (ссылка):

Таблица 1. Описание образца

Прибор для секвенирования	Illumina Genome Analyzer IIx
Организм	*Homo sapiens*
Стратегия секвенирования	Экзомное
Тип чтений	Парноконцевые
Ожидаемое количество чтений (spots)	38,518,929

2) Следующим этапом стала проверка качества чтений с помощью программы **fastqc.** Команда:

```
fastqc SRR10720404_*
```

SRR10720404_1.fastq.gz, SRR10720404_2.fastq.gz - файлы в формате fastq прямых и обратных чтений соответственно

В выдаче получаем два html-файла со сводными отчетами о качестве чтений. Всего 38518929 пар чтений (количество прямых и обратных чтений совпадает (логично)). Далее были проанализированы некоторые графики из отчета (Рис 1,2,3; комментарии в описании):

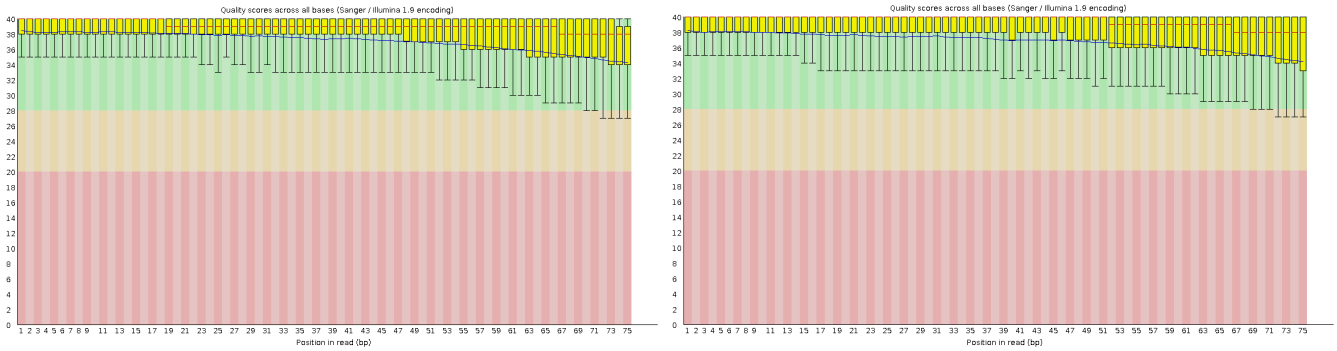


Рис. 1. Per base sequence quality (слева - для прямых чтений, справа - для обратных). Очень хорошие чтения, все позиции с качеством >26 (не считая выбросов), не считая последних четырех позиций и в прямых, и в обратных чтениях все в “зеленой зоне”. В целом, не вызывают беспокойства

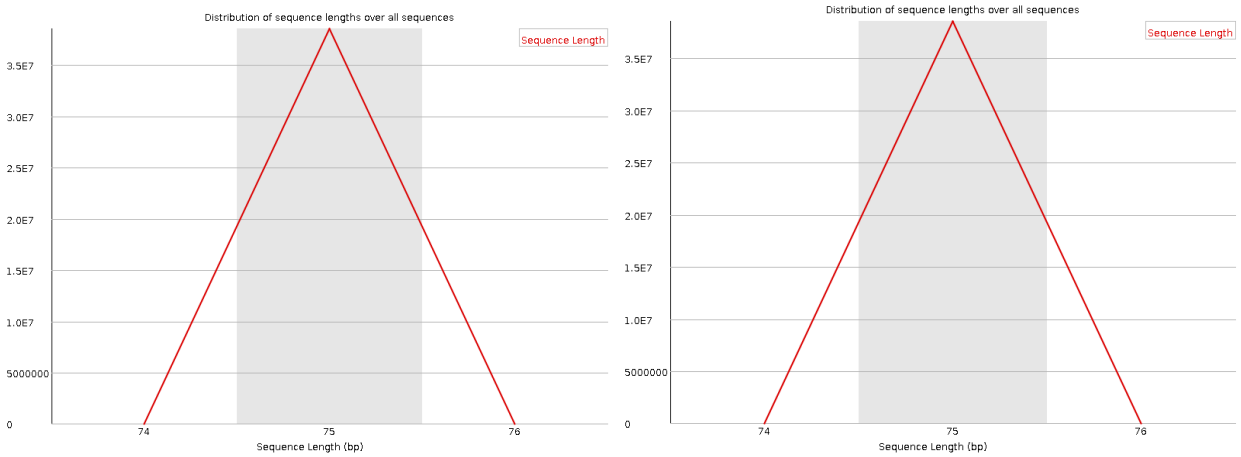


Рис. 2. Sequence Length Distribution (слева - для прямых чтений, справа - для обратных). Все чтения длиной 75 нуклеотидов

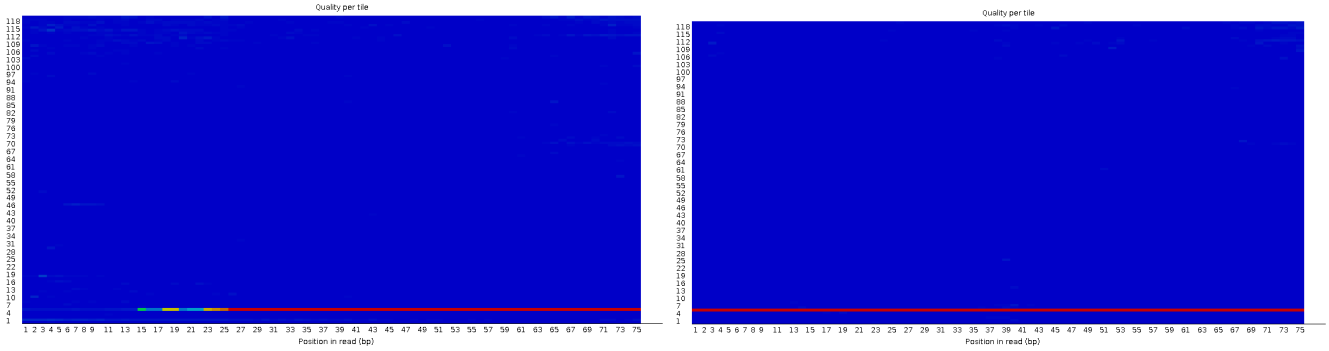


Рис. 3. Per tile sequence quality (слева - для прямых чтений, справа - для обратных). А вот тут уже интересно. Тепловая карта зависимости качества чтений от ячейки подложки, в которой они амплифицировались. Очевидно, что с одной из ячеек что-то не так (это может быть пузырь/контаминация или что-то еще). Частично эти плохие чтения удалились с помощью trimmomatic (см. пункт 5)

3) Далее была проведена фильтрация чтений с помощью программы **trimmomatic:**

```
TrimmomaticPE –phred33 SRR10720404_1.fastq.gz SRR10720404_2.fastq.gz
1Paired.fastq.gz 1Unpaired.fastq.gz 2Paired.fastq.gz 2Unpaired.fastq.gz
TRAILING:20 MINLEN:50
```

SRR10720404_1.fastq.gz, SRR10720404_2.fastq.gz - файлы в формате fastq прямых и обратных чтений соответственно. TRAILING:20 - удаляет с конца нуклеотиды с качеством ниже 20; MINLEN:50 - оставляет только чтения с длиной не меньше 50.

На выходе получаем 4 файла в формате fastq: 1Paired.fastq.gz 1Unpaired.fastq.gz 2Paired.fastq.gz 2Unpaired.fastq.gz, где 1,2 - прямые и обратные чтения соответственно; файлы Paried содержат “пережившие” обработку чтения, “партнерские” (то есть полученные с одного фрагмента, но с разных концов) чтения которых тоже “пережили” обработку, а Unpaired содержит чтения, для которых “партнерские” были удалены (Рис. 4).

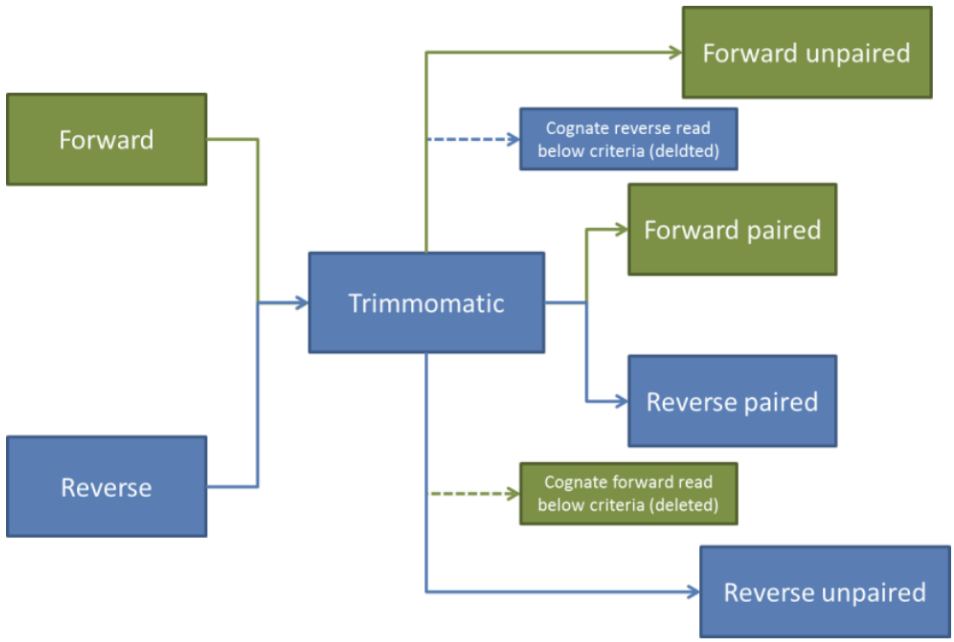


Рис. 4. Поток чтений в Trimmomatic Paired End mode

4) Качество тримированных чтений было проверено с помощью программы **fastqc** (команда - см. пункт 2)**.** Полученные результаты (Таблица 2; Рис 5,6):

Таблица 2. Результаты работы trimmomatic

Количество оставшихся пар чтений (paired)	37,136,668
Процент от исходного количества	96,4%

Рис. 5. Per base sequence quality (слева - для paired, справа - для unpaired; сверху - прямые, снизу - обратные). Качество непарных чтений гораздо хуже, чем парных (что логично, ведь если чтение было удалено, то, наверное, его партнер тоже не очень хорошего качества). Можно заметить, что последние нуклеотиды Unpaired все же имеют качество >20, что следует из параметров работы trimmomatic (см. пункт 3).

Если сравнивать парные чтения до и после триммирования (см Рис. 1 и Рис. 5, левый столбец), то можно заметить, что качество стало еще лучше - все box plotы в зеленой области. Таким образом, снижение качества к концу рида теперь менее выражено.

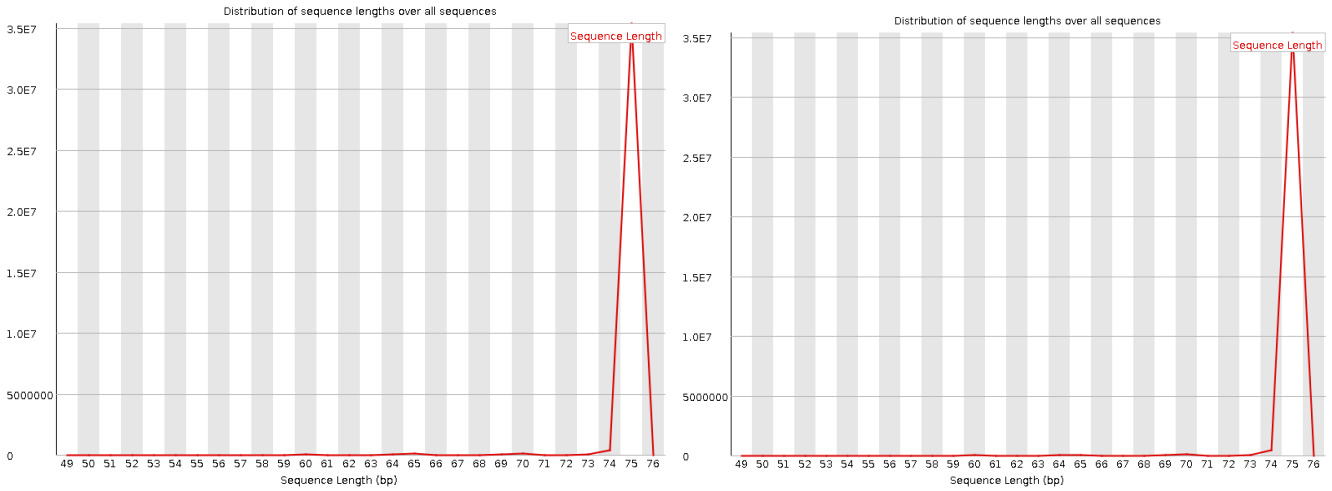


Рис. 6. Sequence Length Distribution (paired, слева - для прямых чтений, справа - для обратных). Длина чтений стала менее однородной (см Рис. 3), но, все равно, подавляющее большинство чтений имеют длину 75.

5) Сводный отчет о качестве чтений

Отчеты о качестве чтений до и после улучшения были систематизированы с помощью программы **multiqc**. Команда:

```
multiqc .
```

В директории, из которой выполняется команда, лежат все полученные ранее fastqc.html–файлы. Получаем html–файл со сводным отчетом (ссылка на сводный отчет). Самый наглядный и обобщающий “график”:

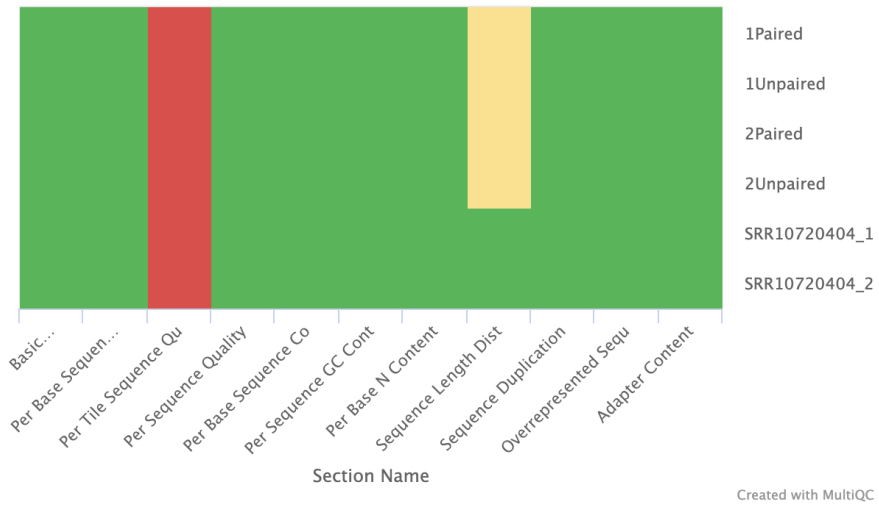


Рис. 7. Status Checks

Heatmap - Состояние каждого раздела FastQC, показывает, являются ли результаты полностью нормальными (зеленый), слегка ненормальными (оранжевый) или очень необычными (красный)

Очевидно, что Per tile sequence quality внушает опасения (до фильтрации - Рис. 3). Поэтому я решил дополнительно посмотреть на эту характеристику после фильтрации (Рис. 8).

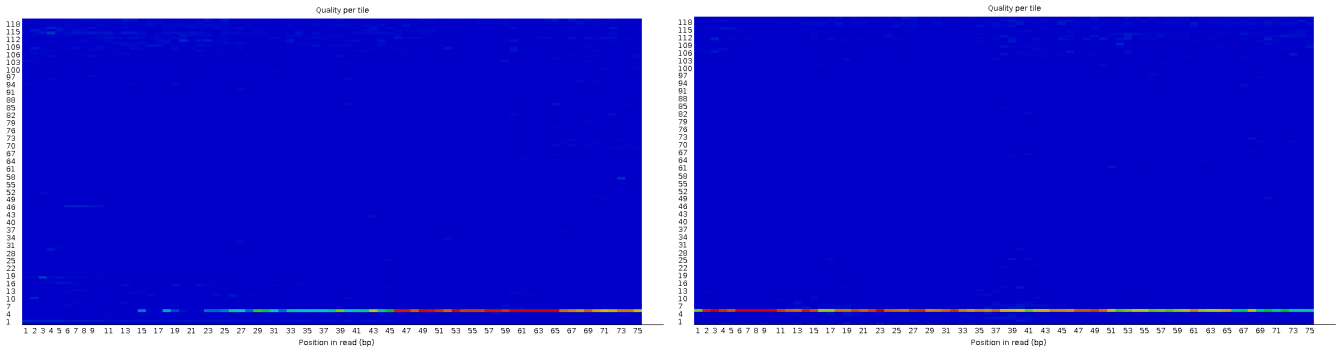


Рис. 8. Per tile sequence quality (слева - для прямых чтений, справа - для обратных). Стало немного лучше, чем было до фильтрации, но все равно довольно плохо. Грустно, что trimmomatic пропустил эти чтения, но это объясняется параметрами его запуска (см. пункт 3). Если trimmomatic в ходе “откусывания” с конца последовательности встречает позицию с качеством ≥20, то чтение остается (если длина ≥50), даже если следующие нуклеотиды очень плохие. Удалить такие чтения можно было с помощью прохода окном, но нам так делать нельзя, поэтому придется смириться с их существованием(

P.S. My favorite thing about multiqc - это шапка html–файла со сводным отчетом (см. картиночку в заглавии).

Часть II (12 практикум)

Задача практикума: картировать чтения хорошего качества на референсный геном и отобрать только такие чтения, которые удалось картировать в корректных парах

2.1 Картирование чтений на референсный геном

Хорошие чтения (paired после фильтрации) были картированы на референсный геном (последовательность пятой хромосомы) с помощью программы **hisat2.**

Картирование программой **hisat2**:

```
hisat2 -x chr5_indexed -1 1Paired.fastq.gz -2 2Paired.fastq.gz -p 8
--no-spliced-alignment >cart.sam 2>hisat2_logs.txt
```

`-x chr5_indexed` – префикс файлов с проиндексированным референсным геномом (см пункт 1.1)
`-1 1Paired.fastq.gz` – прямые чтения
`-2 2Paired.fastq.gz` – обратные чтения
`-p` – данная опция указывает hisat2 запускать указанное количество параллельных потоков поиска
`--no-spliced-alignment` – параметр, запрещающий возможность сплайсинга (то есть запрещаем картирование с разрывами)
`>cart.sam` – записывать вывод в файл `cart.sam`
`2>logs.txt` – направляет stderr в файл `logs.txt`

Получаем файл с расширением .sam (Рис. 9)

```
@HD     VN:1.0  SO:unsorted
@SQ     SN:5    LN:181538259
@PG     ID:hisat2       PN:hisat2       VN:2.2.1        CL:"/usr/bin/hisat2-align-s --wrapper basic-0 -x chr5_indexed -p 8 --no-spliced-alignment -1 /tmp/213143.inpipe1 -2 /tmp/213143.inpipe2"
SRR10720404.805 77      *       0       0       *       *       0       0       CTCGGCAGCTGTATCTTCTGGGATGGATTCTTATAGCACCTATTTGGCAACTGAC D7B@;DD>D:>)>1B2;B><B<:3<;>78?*<B9D7DD>D:B4<375B;D>0&'? YT:Z:UP
SRR10720404.805 141     *       0       0       *       *       0       0       GCGAGATTCAACCTCATAAATGTCTTCATCTTGCTCTTCAACCACGACCCACTGACTTTGAC EBGD?D47@?=>D=8BB=8EB>BBF66C6B:?=40,8<,AAGAC7=C-7C69,A?4<@93B@ YT:Z:UP
SRR10720404.808 77      *       0       0       *       *       0       0       CCTAAAAATTCTGCAAAGCATATGACTTACCCATAAAAAGGCATAGTACCAAGGCATTCAACAAGTGCAA  GGEGGGGGGDGDGGGGGCGGDGD<GGDDGGGGGBGGGDDABGG@91=B?EE=:EC&FC?BB?>-=EB?  YT:Z:UP
SRR10720404.808 141     *       0       0       *       *       0       0       AATTCTGATACTCTTTCCTTATAGGTGTTTCAGCTCAACTCACCAGTACACGTCTGCGGAGAAAC       GGGGGGCEC3GGCGA>GGGDDGG@DA>?>?AGGCGAGGDDGDIIIIHEGIIFC9F@:B967>ECCE       YT:Z:UP
SRR10720404.809 77      *       0       0       *       *       0       0       AGTCAGGATTCTCCCCTGACGACTCTCCTGTGGTCCCTGCACAACCTGGGGAGACTCGGCGCTGCGGGTGCAGA       FC@FFCAEAAGCGGGGDBGG@CF9F=FF=<9;=6AFFA=EGGGEGG@BB>:?(;?=C9=A9?.9?,,6/4@>;>
SRR10720404.809 141     *       0       0       *       *       0       0       GTCTCGTCTTCCCTGGTCTGTGTCCTCTTCTCCTAGGGGCCCAGCCTCTGTGGCCCTGTGACCTGCNG     FA;@;D<=BB?=EBB:EFEF<EFA?AEF3E<><BDA=>?ABE:CCFBB=E:+?DD?7=;7F<F<BB#?    YT:Z:UP
SRR10720404.812 77      *       0       0       *       *       0       0       TACAGACCTCTGAAAATATCTATTGAAAATGGGCAACTTATGATTGTATCATATATAGTCAGCCTTCAGATTCCT      F@B@GGGBE>GBF@BGGFGGHHHFHB>DB?FEEFC=CFFEBGGGGGGGGGGGGGDEGDDGGFEECBGGGGGGGG@
SRR10720404.812 141     *       0       0       *       *       0       0       TGTCCTCCAACAGGGGTAAAACAAAGGAATCATAATCGTTATCCCAGGAATCTGAAGGCTGACTATATA   EFF6AG8@GEDBEGGGIEGIIIDF;GDGGBGGGGGGGGGDGGGGGGG<DG@GGEDDBBDDGBG>GDD@D    YT:Z:UP
SRR10720404.806 77      *       0       0       *       *       0       0       AGATAAAATATTCTTACTTGACTGCAGCAAATACATTATCTCCATTTGGAACAATTATACAATTTTTC     =BECFED@@@@?FEEDD?@?BFEBFGBGGGGGGGE(304A=CBA?92>BBBB;;1AF@FF4;'8474@     YT:Z:UP
SRR10720404.806 141     *       0       0       *       *       0       0       ATGGGCAGCTAAATTATAAATGTCTTTTTCTCAAACAGGGAATCTCCTGATGAACTTCCTGTATA       B8-B7=B5?:DBEBE+*)=<877:>?8;;;17BBDA=2-,1+667:=53;+1;==<?FFB@?+?B      YT:Z:UP
SRR10720404.810 77      *       0       0       *       *       0       0       TGGAGTGGCACAGTTGGTCATTGTGGTTTCTTCTTTTTTAACCTGTCAGCATGGATTCATTAATTATTAAAGCAA     GGGGBEGGGGGGEAGGGEGGGG@GEGG@DGGHHHHHHHHHHGGGDGEDGGGGEGGB3GGDG@GD>GDEGDHHH>B
SRR10720404.810 141     *       0       0       *       *       0       0       TTAATTTCTCCCTCTTTCCTTTTCTTTCCCAAAATTCAATACGTGCAGAATGTCTTCCAAATACT       GG@@GGDDGDDGGGG8<BE:?DB>=GGDEBBEEG@DGGGGDGGDGEGGG@DFECEGGGG?EBDBD       YT:Z:UP
SRR10720404.816 77      *       0       0       *       *       0       0       ATATAAACTGACTGCTTACCTGGTGGGTGTGGTGGTCAGGGTGGCAAACCACAAAGTGCACCC BGGGGGGDGBDHHHHHHGGHHHHFGGDDGA?@>BD@GGDA?2@=>EFBGGBGGD@>@>C>??B YT:Z:UP
SRR10720404.816 141     *       0       0       *       *       0       0       CCATGCAGATAGGGGCGGCTGCTCATAGCCTTCTTCTGTTTGTCGTAGCAGAGGTGAAAACCCCC       IIEIIIIDHHB??>+C;);8=B=B?ADCGE52@6*;::6:(<;?<7B8?><><<@BDGGAGGBBF       YT:Z:UP
SRR10720404.815 77      *       0       0       *       *       0       0       ATGTTTAATCTTGATAGAATTGCTTTTGTACCCTTCTCAAAGATAAGATGTCCATATATATAAGGTTCTATTTCT     IIIIIIIIIIIIIHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIHHHIIIIIHFIIIIIIIIIHFIIDEGIDGEEGGF
SRR10720404.815 141     *       0       0       *       *       0       0       TCTTGTCCCTTTTTTTTGAGAAATTGACAACTTGATTCTAAAATTCACAGGGAAGTGTGAAGACCNA     IIIIIIIIIIGBHHIIIIIBGIGHIGEHIIHIIIIHFIFIFIIGIIIIIBIHBIEGFIIIFDI:#?     YT:Z:UP
SRR10720404.817 77      *       0       0       *       *       0       0       TCTTTCCCTTATGACAGGTAAACCTAAAGAATGTCATCTTCATTGATGATTAAATTTTATTTCTCTTTTTGTCCC     IHIGIHHIIIIIIIIGGGBGGGGGGIHHIIFHGFIIIII>GGGGGFHDFIFHIFIFEIEA>DDDGFIIHFDFHII
SRR10720404.817 141     *       0       0       *       *       0       0       ATCACTCAATTTATTAAAATTATTAGAAGACCACAGAGCTGGGATCTTTCCTCAAAAATTCTTACC      IHIIIIIHDIGBGG:GDGGGHIH@IGGGDGDGGEGIEGGEDIDHEDGIGIIIIHIGDGEEF+8FE@      YT:Z:UP
SRR10720404.819 77      *       0       0       *       *       0       0       ACTCCTTACACTTATATGGCTTCTCTCCGGTGTGGAGTCTCTGGTGGGAGCTGAGGCCCGCATGCTGACTGAAGC     HIIIIIIIIIIIGIGHHDIIHHIFHAEFCE=C@CCDDBDBGDDD=EECCCHBEEBBFGF@@?:?@=@@D==?A>A
SRR10720404.819 141     *       0       0       *       *       0       0       CTCATTCGTCACTATCGGATCCACACTGGGGAGAAGCCTTATCAGTGTAACGAATGTGG     IIIIIGGG<GGGDGGGFDEEGGDGGIHIHHEIFFD>@EBCDBDDD@EBFEEEEEBE:BE     YT:Z:UP
SRR10720404.818 77      *       0       0       *       *       0       0       TTCACCTTTAAATAGTACACCAGAAGCTCATTCAGAGCTGGGTCAGCATTCAGATTCACAAGGAAGCATTTATCA     HH@HHHHHHHHHGHFGEEGFAEGGGGGGGGGBGGGBHHFGGGDGGGGGGAG@GG>GEDDGGDGGGGGG>FCGAGE
SRR10720404.818 141     *       0       0       *       *       0       0       CAACACTATTGAATAATTGTAACATTGTAATGTAAACGCTTGGATATCTCACTGCCTCAATAGCT     IIIGGIFIBGGDDDGAA?A;GGGDGGDBGB?F@DCEB?C@8@<?=@@D@GEGA3GECECA@DBGB     YT:Z:UP
SRR10720404.823 77      *       0       0       *       *       0       0       GCTCGCTCACCAGCGTCTGCTTTTAGACCAGCAGGGGGAGGGGGCAGGGGG     IGIIFHEHHIIGIIGFIEIIIFIDF>IIDE<EBBF@@BCDD=DDDEBDB?     YT:Z:UP
SRR10720404.823 141     *       0       0       *       *       0       0       ACCCTAGGGTCCAGCCTCCCCCAAAGCCTCCCCCCTCGCCCCAGGCTGGCTC     IIIIGDGGE?GGGEGIFIIFIIIGH2CEA?F=@@=3AACC?8;1<8;89=9@     YT:Z:UP
SRR10720404.827 77      *       0       0       *       *       0       0       CTCTTAGTGATTGATTGAAAGCACACTGTTACTAATTGACTCGGTATGAAGTGCTTTTTTTTCTTCCCTTTCAAG     EEEEBBD?D@;==;=BDDDDGGGG@GBEEEEGGG@GBGB<GEGEDD@>?C?A?8?B*:888635=;>AAAAADDBDC
SRR10720404.827 141     *       0       0       *       *       0       0       GATATACAGATACATATGTGTGTGTAGATATATATATATATAGTTTTTTTTTTTTTTAA     GG4GGEFD+DFFE=47>5>8?DBDBFBGDGIIE>GGDE@E1:159FFD@>=@@@@=*?B     YT:Z:UP
SRR10720404.821 77      *       0       0       *       *       0       0       GGTACGTGGCCATCTGCCACCCTCTCCGATATTTCATCATCATGACCTGAAAGTCTGCATCACTCTGGCCATCA     GGEGGIIIIGIIIHIIIIIDIIIEBGGBGEEEEGGGDGIBIIHHHHHEH@BDDB@EBEEEAHEEBFC>FFGG;IE
SRR10720404.821 141     *       0       0       *       *       0       0       CACGAGGCCCACAAAAGGGCAGTCTTAGGATGAGGCTCACATGGACCATAGCCAGGAGGGAGCCACA     IIIIGHEIHIIIHGFDIGBG<B8E2AEBEEEFG<FE-CC<E@ECEHHIEFHIEEFB>CD=DBB-B??     YT:Z:UP
SRR10720404.820 77      *       0       0       *       *       0       0       AAAAATTCTGTTATACAGCAGAGTCTGCATCAATAGCTATCAGAAAAAGCTTTCTGTCTACCTGCCCAGGTGTAA     IIIIIIIIIIIIIIIIIIIIIIIIIHHFHIIIIIHIIIIIGHIIIIIFIIFHICIIIIHHIFIFGBEGFGDEE?FDFF
```

Рис.9. Первые несколько строк полученного sam-файла (извините, что мелко)

Sam-файл состоит из “шапки” и основного содержания (выравниваний)
Шапка:

@HD - header line; VN - версия формата; SO - способ сортировки выравниваний (в данном случае несортированы)
@SQ - информация о референсных последовательностях (здесь одна); SN - имя последовательности (5); LN - длина последовательности (181538259)
@PG - информация о программе, с помощью которой выполнено картирование

Основное содержание (по столбцам):
QNAME: имя (ID) рида (ex: SRR10720404.805)
FLAG: кодировка некоторой информации о картировании этого рида (например, flag 77 в данном случае означает, что этот рид прямой и он не был картирован)
и тд (см. Рис. 10)

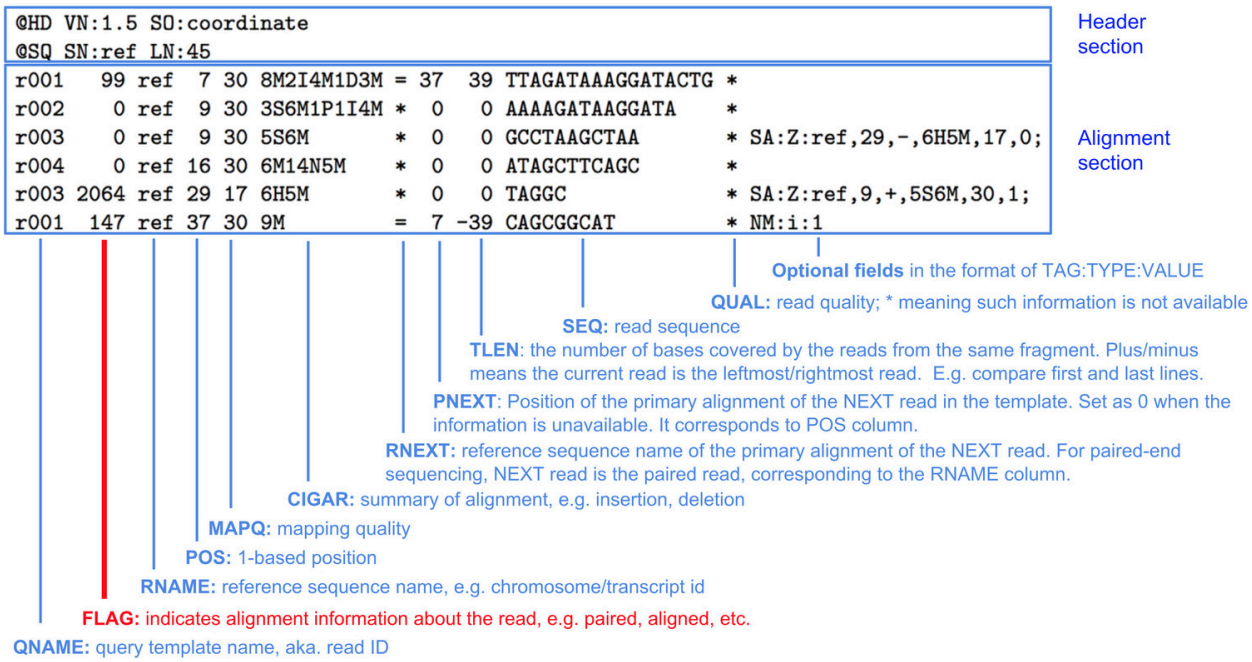


Рис. 10. Анатомия sam файла

2.2 Конвертация sam в bam

sam файл весит >15 Гб, поэтому он был конвертирован (и сортирован) в бинарный аналог - bam файл с помощью следующей команды:

```
samtools sort -o cart.bam cart.sam
```

-o задает вывод в файл
полученный bam файл весит меньше - 4Гб

Далее полученный bam файл был проиндексирован с помощью **samtools index**:

```
samtools index cart.bam
```

2.3 Анализ bam файла

Заглянуть в bam файл просто так не получится, он бинарный, поэтому он был проанализирован с помощью возможностей программы samtools, а именно **samtools flagstat,** которая выводит количества ридов, разбитым по категориям на основании, в основном, FLAGs.

```
samtools flagstat cart.bam > flagstat.txt
```

Получили следующий текстовый файл:

```
75116214 + 0 in total (QC-passed reads + QC-failed reads)
74273336 + 0 primary
842878 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
5263673 + 0 mapped (7.01% : N/A)
4420795 + 0 primary mapped (5.95% : N/A)
74273336 + 0 paired in sequencing
37136668 + 0 read1
37136668 + 0 read2
3871328 + 0 properly paired (5.21% : N/A)
3967302 + 0 with itself and mate mapped
453493 + 0 singletons (0.61% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

Каждая категория (строчка) разбита на риды, прошедшие QC- Quality Check и не прошедшие (#PASS + #FAIL)

Итак,

1) *Что значит число в поле «in total»?* Общее число полученных выравниваний ридов с референсом (очевидно, оно может быть не одно для каждого рида, например, из-за повторов) (см. Рис. 11)

2) *Сколько чтений (не пар!) поступило на картирование?* `74273336` – можно понять по строке `primary` – если для рида нашлось несколько выравниваний, то выбирается лучшее, ну а если одно, то оно и есть лучшее). А значит количество primary выравниваний будет равно количеству ридов, поступивших на картирование

Multiple mapping The correct placement of a read may be ambiguous, e.g., due to repeats. In this case, there may be multiple read alignments for the same read. One of these alignments is considered primary. All the other alignments have the secondary alignment flag set in the SAM records that represent them. All the SAM records have the same QNAME and the same values for 0x40 and 0x80 flags. Typically the alignment designated primary is the best alignment, but the decision may be arbitrary.[3]

Рис. 11. Выдержка из SAM file format specification

3) *Сколько чтений картировано на референс в корректных парах в штуках?* 3,871,328 (строка `propely paired`)

4) *Сколько чтений картировано на референс в корректных парах в процентах относительно потупивших на картирование?* 5,21% (указано в скобках в строке `propely paired`, проверено на калькуляторе)

Процент ожидаемо низкий, так как изначально у нас чтения для всего экзома, а референс - только одна из хромосом (пятая). +Производится дополнительная фильтрация, так как корректно картированными парами признаются только пары ридов, картированные:
1) по направлению друг к другу
2) недалеко друг от друга
Таким образом, только 99, 147 и 163, 83 пары ридов будут `propely paired`

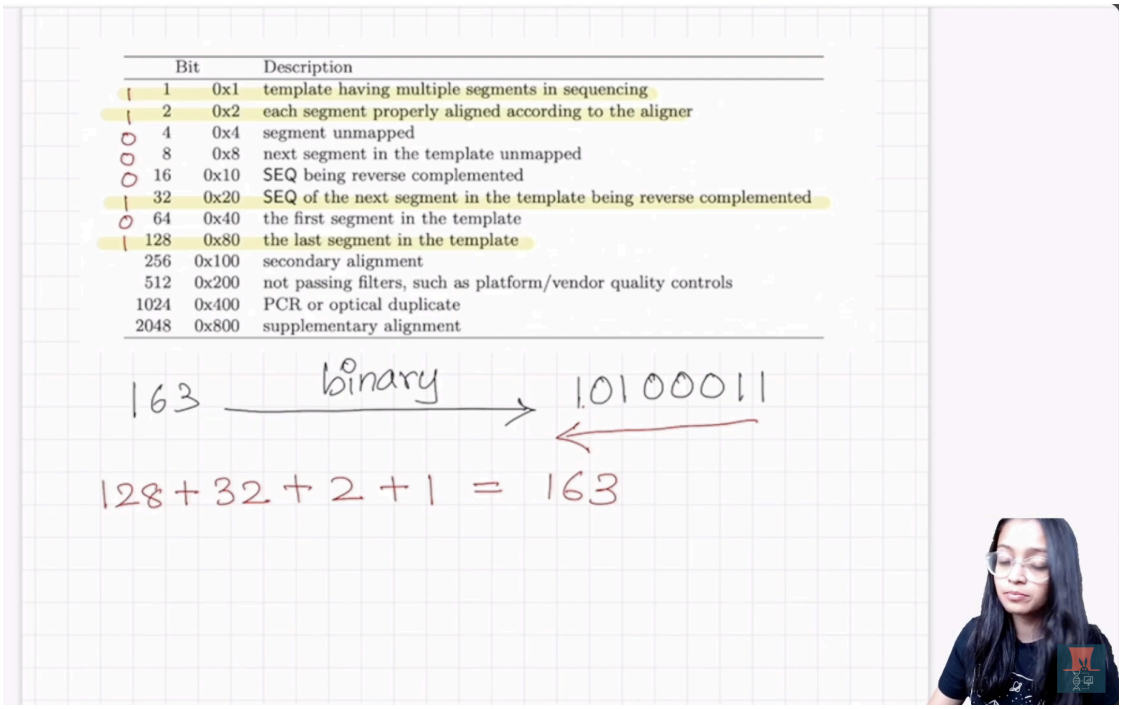


Рис. 12. Мой любимый youtube-канал на данный момент

2.4 Получение чтений, картированных на пятую хромосому

Как было сказано выше, изначально у нас чтения для всего экзома, а референс - только одна из хромосом (пятая). Поэтому необходимо отобрать только чтения, которые на нее картировались. Это было сделано с помощью следующей команды:

```
samtools view -h -bS cart.bam 5 > cart.chr5.bam
```

5 - Regions (определяем регион, риды, картированные на который, будут выводиться, в данном случае **5 - название хромосомы**, определено в пункте 1.1)
-h - выводить вместе с header (“шапкой”)
-b - выводить в формате bam
-S - автоматически определить формат ввода
Получили файл в формате bam с чтениями, картированными на пятую хромосому

Далее к полученому файлу была применена команда flagstat (см. предыдущий пункт)

```
samtools flagstat cart.chr5.bam > flagstat.chr5.txt
```

Получили текстовый файл:

```
5717166 + 0 in total (QC-passed reads + QC-failed reads)
4874288 + 0 primary
842878 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
5263673 + 0 mapped (92.07% : N/A)
4420795 + 0 primary mapped (90.70% : N/A)
4874288 + 0 paired in sequencing
2437144 + 0 read1
2437144 + 0 read2
3871328 + 0 properly paired (79.42% : N/A)
3967302 + 0 with itself and mate mapped
453493 + 0 singletons (9.30% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

По сравнению с аналогичным файлом из предыдущего пункта, тут уже гораздо меньше рассматриваемых чтений (4,874,288 по сравнению с 74,273,336) и больше процент properly mapped (хотя мне не очень понятно, почему вообще остались некартированные чтения - возможно, не проходит по mapping quality).

2.5. Получение только правильно картированных пар чтений

Как обсуждалось в пункте 2.3, не все пары чтений properly paired, поэтому надо отобрать только правильно спаренные пары чтений. Это было сделано с помощью следующей команды:

```
samtools view -f 2 -bS cart.chr5.bam > proper.chr5.bam
```

-f 2 - выводить только чтения, обладающие FLAG=2 (Read mapped in proper pair)

Далее к полученому файлу была применена команда flagstat:

```
samtools flagstat proper.chr5.bam > flagstat.proper.txt
```

Получили текстовый файл:

```
4382096 + 0 in total (QC-passed reads + QC-failed reads)
3871328 + 0 primary
510768 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
4382096 + 0 mapped (100.00% : N/A)
3871328 + 0 primary mapped (100.00% : N/A)
3871328 + 0 paired in sequencing
1935664 + 0 read1
1935664 + 0 read2
3871328 + 0 properly paired (100.00% : N/A)
3871328 + 0 with itself and mate mapped
0 + 0 singletons (0.00% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

Ура, жизнь прекрасна (нет), все пары чтений в этом bam файле правильно спарены и картированы. Всего их 3,871,328 (собственно, столько же было в строках properly paired в предыдущих файлах).

bam файл с правильно спаренными и картированными чтениями был проиндексирован:

```
samtools index proper.chr5.bam
```

Дальнейшая работа (поиск вариантов) осуществлялась только с этим bam файлом и его индексами.

2.6. Получение чтений, картированных только в границы экзома

Как уже было сказано, у нас есть чтения экзомного секвенирования. Поэтому чтения, которые картировались за пределы экзома, нам не нужны. Для этого была использована программа **bedtools intersect:**

```
bedtools intersect –abam proper.chr5.bam –b
/mnt/scratch/NGS/DATA/genes/seqcap_hg38.bed > proper.exom.bam
```

-abam - входные выравнивая в формате bam
-b - сравниванием с features в файле формата bed

Получаем bam файл с чтениями, картированными в пределах экзома.

Далее к полученому файлу была применена команда flagstat:

```
samtools flagstat proper.exom.bam > flagstat.exom.txt
```

Получили текстовый файл:

```
2556586 + 0 in total (QC-passed reads + QC-failed reads)
2290402 + 0 primary
266184 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
2556586 + 0 mapped (100.00% : N/A)
2290402 + 0 primary mapped (100.00% : N/A)
2290402 + 0 paired in sequencing
1143424 + 0 read1
1146978 + 0 read2
2290402 + 0 properly paired (100.00% : N/A)
2290402 + 0 with itself and mate mapped
0 + 0 singletons (0.00% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

По сравнению с прошлым пунктом количество ридов значительно уменьшилось, а значит много ридов картировалось за пределами экзома (на самом деле, я думал, будет меньше таких, ведь у нас данные экзомного секвенирования).

2.7. Получение чтений, картированных в границы расширенного экзома

Далее были проведены те же процедуры, но в качестве границ экзома был взять расширенный файл. Команды:

```
$ bedtools intersect -abam proper.chr5.bam -b
/mnt/scratch/NGS/DATA/genes/seqcap_hg38_50.bed > wide_exom.bam
$ samtools flagstat wide_exom.bam > flagstat.wide_exom.txt
```

Получили следующий текстовый файл:

```
2773035 + 0 in total (QC-passed reads + QC-failed reads)
2479610 + 0 primary
293425 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
0 + 0 primary duplicates
2773035 + 0 mapped (100.00% : N/A)
2479610 + 0 primary mapped (100.00% : N/A)
2479610 + 0 paired in sequencing
1239017 + 0 read1
1240593 + 0 read2
2479610 + 0 properly paired (100.00% : N/A)
2479610 + 0 with itself and mate mapped
0 + 0 singletons (0.00% : N/A)
0 + 0 with mate mapped to a different chr
0 + 0 with mate mapped to a different chr (mapQ>=5)
```

По сравнению с прошлым пунктом, количество отобранных ридов (почти всех категорий) оказалось несколько больше (всего `2,479,610` против `2,290,402`). Ожидаемый результат, так здесь мы берем расширенную версию разметки экзома).

Часть III (13 практикум)

Задача практикума: получить список вариантов на основании полученного ранее bam файла и аннотировать их средствами VEP

3.1 Получение вариантов

С помощью инструментария **bcftools** был получен vcf файл с вариантами. Команда:

```
bcftools mpileup -f chr5.fa proper.chr5.bam | bcftools call -mv -o variants.vcf
```

-f – указание на последовательность референса (и на ее индексированную версию)
-v – выводить только вариабельные сайты
-o FILE – output file

Вот такой вот vcf файл получился (Рис. 13):

```
##fileformat=VCFv4.2h=181538259>
##FILTER=<ID=PASS,Description="All filters passed"> than observed.">
##bcftoolsVersion=1.16+htslib-1.16g,Description="Indicates that the variant is an INDEL.">
##bcftoolsCommand=mpileup -f ../samtools/chr5.fa ../hisat2/proper.chr5.bamads supporting an indel">
##reference=file://../samtools/chr5.facription="Maximum fraction of raw reads supporting an indel">
##contig=<ID=5,length=181538259>ger,Description="Raw read depth">
##ALT=<ID=*,Description="Represents allele(s) other than observed.">s for filtering splice-site artefacts in RNA-seq da>
##INFO=<ID=INDEL,Number=0,Type=Flag,Description="Indicates that the variant is an INDEL.">as (closer to 0 is better)">
##INFO=<ID=IDV,Number=1,Type=Integer,Description="Maximum number of raw reads supporting an indel">ser to 0 is better)">
##INFO=<ID=IMF,Number=1,Type=Float,Description="Maximum fraction of raw reads supporting an indel"> to 0 is better)">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Raw read depth">z test of Mapping Quality vs Strand Bias (closer to 0 >
##INFO=<ID=VDB,Number=1,Type=Float,Description="Variant Distance Bias for filtering splice-site artefacts in RNA-seq data (bigger is better)",Version="3">
##INFO=<ID=RPBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Read Position Bias (closer to 0 is better)">>
##INFO=<ID=MQBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Mapping Quality Bias (closer to 0 is better)">
##INFO=<ID=BQBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Base Quality Bias (closer to 0 is better)">
##INFO=<ID=MQSBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Mapping Quality vs Strand Bias (closer to 0 is better)">
##INFO=<ID=NMBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Number of Mismatches within supporting reads (closer to 0 is better)">
##INFO=<ID=SCBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Soft-Clip Length Bias (closer to 0 is better)">
##INFO=<ID=FS,Number=1,Type=Float,Description="Phred-scaled p-value using Fisher's exact test to detect strand bias">li>
##INFO=<ID=SGB,Number=1,Type=Float,Description="Segregation based metric."> called genotypes">
##INFO=<ID=MQ0F,Number=1,Type=Float,Description="Fraction of MQ0 reads (smaller is better)">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="List of Phred-scaled genotype likelihoods">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes for each ALT allele, in the same order as listed">
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotypes">
##INFO=<ID=DP4,Number=4,Type=Integer,Description="Number of high-quality ref-forward , ref-reverse, alt-forward and alt-reverse bases">
##INFO=<ID=MQ,Number=1,Type=Integer,Description="Average mapping quality">
##bcftools_callVersion=1.16+htslib-1.16
##bcftools_callCommand=call -mv -o variants.vcf; Date=Tue Dec 17 04:44:03 2024
#CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  ../hisat2/proper.chr5.bam
5       11580   .       A       T       30.4183 .       DP=2;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,1,0;MQ=60      GT:PL   1/1:60,3,0
5       13018   .       C       A       19.4636 .       DP=2;VDB=0.84;SGB=-0.453602;MQSBZ=0;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,1,1;MQ=60     GT:PL   1/1:49,6,0
5       13114   .       G       C       9.88514 .       DP=1;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:39,3,0
5       13125   .       G       T       4.38466 .       DP=1;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:32,3,0
5       15784   .       A       G       3.77146 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;BQBZ=1;NMBZ=1;SCBZ=1;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,0,1;MQ=60  GT:PL   0/1:34,0,32
5       15851   .       C       A       43.4147 .       DP=2;VDB=0.18;SGB=-0.453602;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,2;MQ=60     GT:PL   1/1:73,6,0
5       16645   .       A       G       3.21931 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;BQBZ=-1;NMBZ=0;SCBZ=0;FS=0;MQ0F=0;AC=1;AN=2;DP4=1,0,1,0;MQ=60 GT:PL   0/1:33,0,34
5       16685   .       G       A       30.4183 .       DP=2;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:60,3,0
5       16709   .       C       T       3.21931 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;MQSBZ=0;BQBZ=-1;NMBZ=0;SCBZ=0;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,1,0;MQ=60 GT:PL   0/1:33,0,34
5       16823   .       T       C       81.415  .       DP=3;VDB=0.0481132;SGB=-0.511536;MQSBZ=0;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,2,1;MQ=60        GT:PL   1/1:111,9,0
5       16836   .       G       A       83.415  .       DP=3;VDB=0.0481132;SGB=-0.511536;MQSBZ=0;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,2,1;MQ=60        GT:PL   1/1:113,9,0
5       16904   .       T       C       24.0788 .       DP=3;VDB=0.52;SGB=-0.453602;RPBZ=-1.22474;MQBZ=0;MQSBZ=0;BQBZ=-0.707107;NMBZ=1.22474;SCBZ=0.707107;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,2,0;MQ=60      GT:PL   0/1:57,0,31
5       16914   .       C       G       88.6983 .       DP=5;VDB=0.497461;SGB=-0.556411;RPBZ=-1.45095;MQBZ=0;MQSBZ=0;BQBZ=-1.45095;NMBZ=1.58114;SCBZ=0.5;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,2,2;MQ=60       GT:PL   0/1:123,0,25
5       16982   .       C       T       10.7923 .       DP=1;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:40,3,0
5       17108   .       A       T       3.21802 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;MQSBZ=0;BQBZ=1;NMBZ=-1;SCBZ=0;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,1,0;MQ=60 GT:PL   0/1:33,0,29
5       17121   .       T       A       3.72425 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;MQSBZ=0;BQBZ=1;NMBZ=-1;SCBZ=0;FS=0;MQ0F=0;AC=1;AN=2;DP4=0,1,1,0;MQ=60 GT:PL   0/1:34,0,20
5       17214   .       G       A       7.30814 .       DP=1;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:36,3,0
5       17230   .       A       T       5.04598 .       DP=1;SGB=-0.379885;FS=0;MQ0F=0;AC=2;AN=2;DP4=0,0,0,1;MQ=60      GT:PL   1/1:33,3,0
5       17963   .       T       A       3.77146 .       DP=2;SGB=-0.379885;RPBZ=-1;MQBZ=0;BQBZ=1;NMBZ=1;SCBZ=1;FS=0;MQ0F=0;AC=1;AN=2;DP4=1,0,1,0;MQ=60  GT:PL   0/1:34,0,32
```

Рис. 13. Первые несколько строк полученного vcf файла (думаю, у этого рисунка неспроста тринадцатый номер)

Структура vcf файла

1) Шапка (строки начинаются с ##)
2) Строка (одна) с заголовками столбцов (начинается с #)
3) Основное содержание (информация о вариантах)

Подробнее о столбцах:

`CHROM`	Название хромосомы (5)
`POS`	Позиция варианта

`ID`	Может быть любая информация о варианте (в данном случае пустой - везде “.”
`REF`	Референсная аллель (видно только SNP, но вообще могут быть и отличия в несколько нуклеотидов)
`ALT`	Альтернативная аллель
`QUAL`	Качество варианта (Phred-scaled)
`FILTER`	Везде “.”, так как дополнительная фильтрация не проводилась
`INFO`	Различные характеристики варианта
`FORMAT`	Список параметров варианта для конкретного образца
`ID образца`	Значения параметров из `FORMAT`

Для большинства вариантов приведены параметры GT:PL, поэтому поясню их подробнее на примере:
`GT:PL  1/1:60,3,0 (последние два столбца)`
`GT` – наиболее вероятный генотип образца, то есть человека, экзом которого секвенировали (в данном случае, 1/1 - альтернативная гомозигота по этой позиции - Рис. 14)
`PL` – нормализованные «вероятности» возможных генотипов (по шкале Phred). Поле содержит 3 числа, что соответствует генотипам 0/0, 0/1, 1/1. PL наиболее вероятного генотипа = 0 (для этой позиции, насколько я понимаю, вероятнось того, что образец референсная гомозигота, в миллион раз меньше, чем что он альтернативная гомозигота).

genotype	description
`0/0`	the sample is a homozygous reference
`0/1`	the sample is heterozygous (carries both reference and alternate alleles)
`1/1`	the sample is a homozygous alternate
`./.`	No genotype called or missing genotype

Рис. 14. Значения GT

Проанализируем полученный vcf файл с помощью команды **bcftools stat:**

```
bcftools stat variants.vcf > stat.txt
```

Получаем большой текстовый файл со всякими статистками (например, распределение по quality, depth и еще много чего - Рис. 16)
Прочитав описание, обратим внимание на эти строчки (Рис. 15):

```
# SN    [2]id   [3]key  [4]value
SN      0       number of samples:      1
SN      0       number of records:      86101
SN      0       number of no-ALTs:      0
SN      0       number of SNPs: 84890
SN      0       number of MNPs: 0
SN      0       number of indels:       1211
SN      0       number of others:       0
SN      0       number of multiallelic sites:   65
SN      0       number of multiallelic SNP sites:       41
```

Рис. 15. Поле SN (Summary numbers)

a) *Сколько получилось вариантов?* 86101 (number of records)
b) *Сколько из полученных вариантов являются однонуклеотидными заменами?* 84890 (number of SNPs)
c) *Сколько получилось коротких вставок и делеций?* 1211 (number of indels)

```
# TSTV  [2]id   [3]ts   [4]tv   [5]ts/tv
TSTV    0       52195   32736   1.59
```

Рис. 16. Поле TSTV – сводная статистика по транзициям/трансверсиям

d) Посмотрим подробнее на вывод команды **bcftools mpileup:**

```
bcftools mpileup -f chr5.fa proper.chr5.bam > all_variants.vcf
```

-f – указание на последовательность референса (и на ее индексированную версию)

Насколько я понимаю, получается gvcf файл, содержащий информацию по всем позициям [на которые картировались риды, наверное], а не только вариабельные сайты (но и те где нет отличий между референсом и образцом). Соответственно, вывод только позиций с отличающимися вариантами происходит с помощью команды `bcftools call -v`.

3.2 Фильтрация вариантов

Варианты были отфильтрованы с помощью команды **bcftools filter**:

```
bcftools filter –i'QUAL>30 && DP>50' variants.vcf –o variants_filt.vcf
```

-i'expression' – отобрать сайты, для которых верно expression, в данном случае чтения с качеством больше 30 и "глубиной" (общее количество чтений, прошедших фильтрацию и поддерживающих каждую из представленных аллелей) больше 50
-o – output в файл

Проанализируем полученный vcf файл с помощью команды **bcftools stat**:

```
bcftools stats variants_filt.vcf > filt_stat.txt
```

Получаем текстовый файл со статистикой (Рис. 17).

```
# SN    [2]id   [3]key  [4]value
SN      0       number of samples:      1
SN      0       number of records:      1720
SN      0       number of no-ALTs:      0
SN      0       number of SNPs: 1651
SN      0       number of MNPs: 0
SN      0       number of indels:       69
SN      0       number of others:       0
SN      0       number of multiallelic sites:   0
SN      0       number of multiallelic SNP sites:       0
```

Рис. 17. Поле SN (Summary numbers) для отфильтрованных вариантов

a) ***Сколько осталось вариантов?*** 1720 (1,99%)
b) ***Сколько осталось однонуклеотидных замен?*** 1651 (1,94%)
c) ***Сколько осталось коротких вставок и делеций?*** 69 (5,70%)

3.3 Аннотация вариантов

Профильтрованные варианты были проаннотированы с помощью сервиса VEP. Для начала опишу информацию из раздела Summary statistics (Таблица 3, Рис. 19,20).

Таблица 3. Basic Statistics

Category	Count
Variants processed	1720
Variants filtered out	0
Novel / existing variants	398 (23.1) / 1322 (76.9)
Overlapped genes	824
Overlapped transcripts	5362
Overlapped regulatory features	22

1720 вариантов обработано (столько же, сколько было входном файле - см. пункт 3.2);
0 варинтов отфильтровано (уже сами до этого отфильтровали - см. пункт 3.2)
Новых вариантов (то есть тех, которых нет в Ensembl Variation database) - 398 (23,1%)
Описанных - 1322 (76,9%)
Я не сразу понял, что такое overlapped, но, видимо, это количество генов/транскриптов/регуляторных фич, на которые попали наши варианты (так и есть).

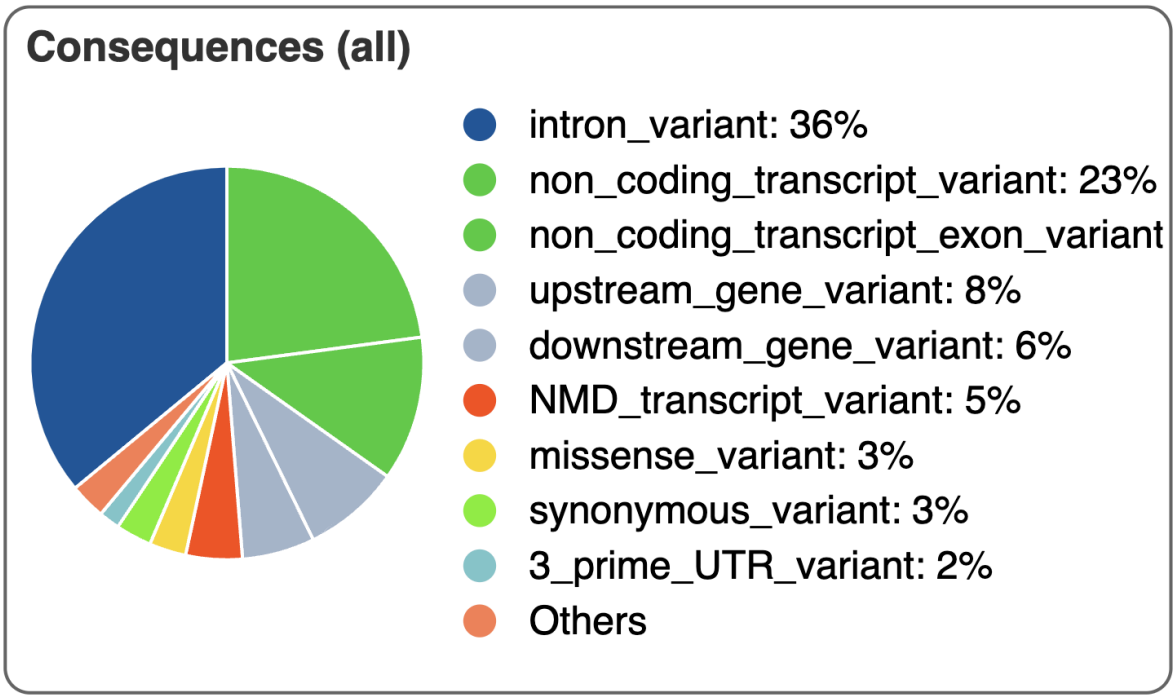


Рис. 19. Consequenses (all). Распределение эффектов вариантов (условно куда они попали)

Большинство, ожидаемо, попали на некодирующие последовательности - интроны и проч. Сначала меня сильно удивило такое относительно большое количество NMD_transcript_variant, так как я подумал, что это те варианты, появление которых будет приводить к тому, что транскрипт будет подвержен моему любимому Nonsense-Mediated Decay). А такие варианты – это по сути тоже самое, что stop_gained, которых гораздо меньше. Но потом я понял, что не так понял. Это варианты, которые попадают в транскрипты, которые и так (“референсно”) подвержены нонсенс-опосредованному распаду. Вот так вот.

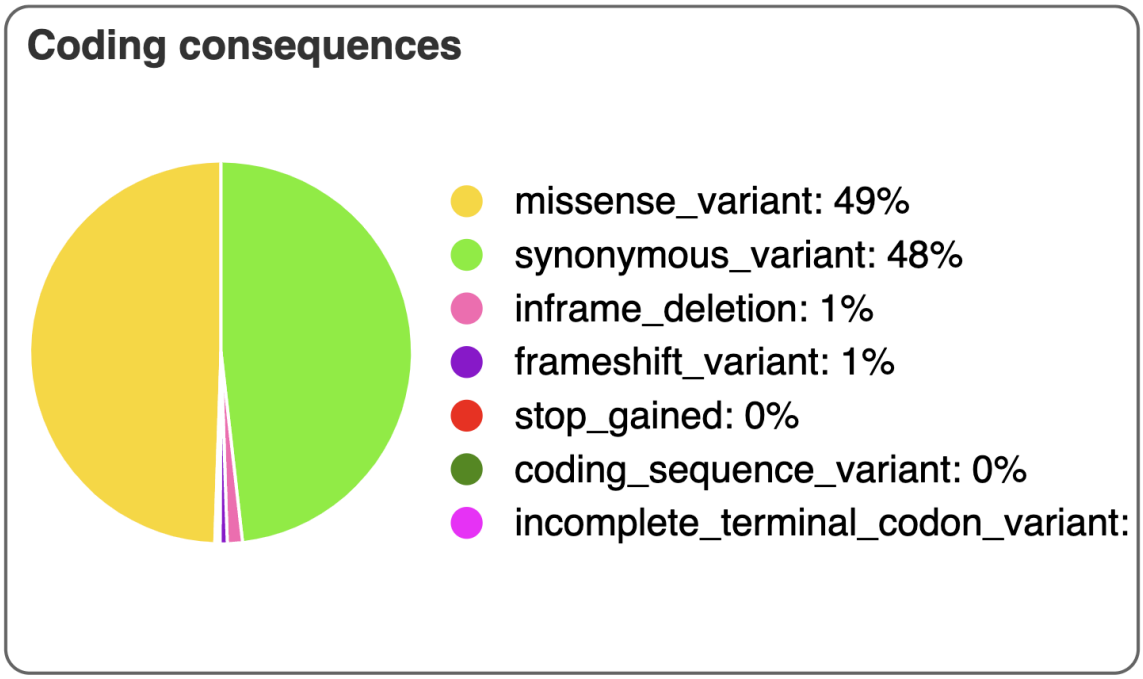


Рис. 20. Coding consequenses. Распределение эффектов вариантов среди попавших на кодирующие последовательности

Все тоже более менее ожидаемо, большинство вариантов соотвествуют синонимичным или миссенс-мутациям. Так и ожидалость исходя из свойств генетического кода.

25 вариантов с импактом HIGH (1,5%). Из них:

1 приходится на псевдоген, остальные на гены, из них:

- 4 попали на NMD_transcript_variant (то есть вряд ли сильно влияют, так как транскрипт деградирует на выходе из ядра)
- 1 на интрон (retained_intron)
- 2 на protein_coding_CDS_not_defined
- 17 на нормальные кодирующие транскрипты (то есть на экзоны)

Из этих 17:

- 10 frameshift_variant (сдвигов рамки считывания) в 5 генах
- 4 splice_acceptor_variant в 2 генах
- 3 stop_gained в 2 генах

Всего генов, которые затронуты вариантами с импактом HIGH в экзонах: 8

//Это PARP8, CLINT1, MROH2B, PGGT1B, SDHA, PCDHGA8, TIGD6, OR4F3.

Ссылка на сценарий для командной строки на основании команд, использованных для выполнения практикумов 11-13: вот она (ой, русские буквы не понимает)

Часть IV (15 практикум)

Задача практикума: построить экспрессионный профиль на основании данных секвенирования РНК.

4.1 Описание образца

Мне достались чтения **ENCFF641WPY (ID)**. Ниже приведена некоторая информация, полученная из ENCODE (ссылка):

Таблица 4. Описание образца

Организм	*Mus musculus*
Ткань	heart tissue
Стратегия секвенирования	Тотальная РНК
Тип чтений	Одноконцевые
Цепь-специфичность	Цепь-специфичные (обратная)

4.2 Проверка качества чтений

Качество исходных чтений было проверено с помощью программы fastqc (Рис. 21,22,23). Количество чтений – 48,692,928.

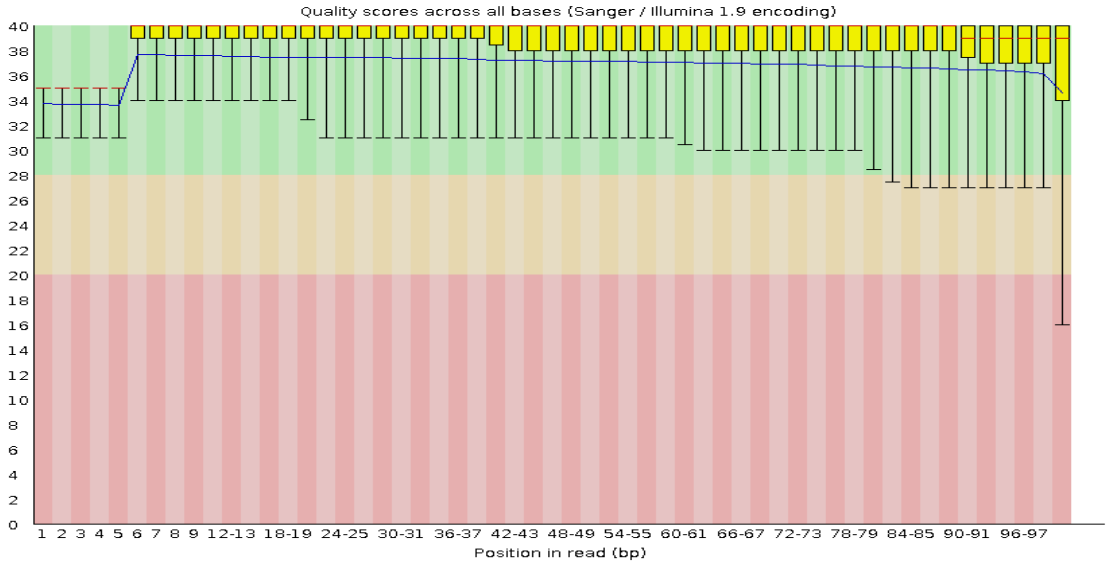


Рис. 21. Per base sequence quality

В целом довольно хорошие чтения, все боксплоты, кроме последнего в зеленой зоне. В начале только 5 позиций сильно выделяются (может димеры адаптеров в каких-то чтениях??)

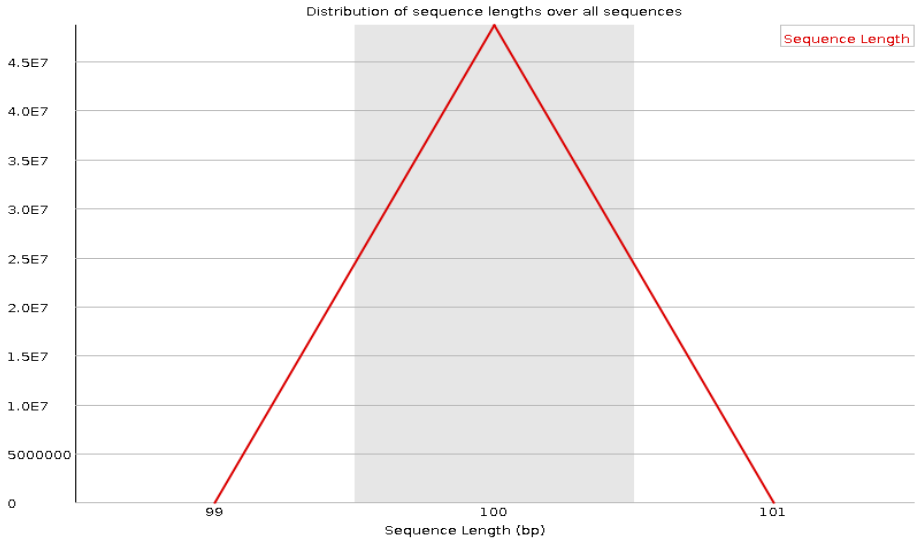


Рис. 21. Sequence Length Distribution. Все чтения длиной 100

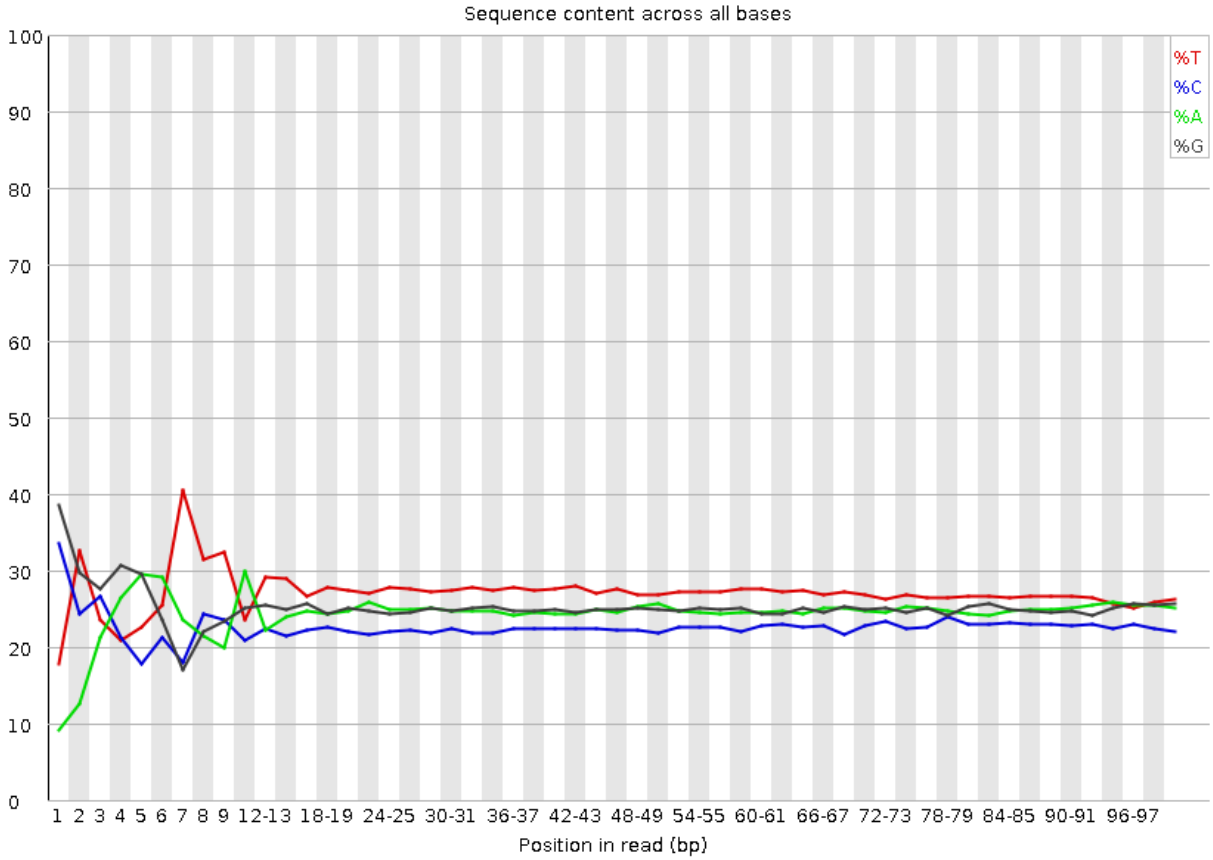


Рис. 22. Per base sequence content. Все-таки первые 12-13 позиций вызывают опасения и их хорошо бы триммировать немножно, но нельзя, поэтому просто примем как факт

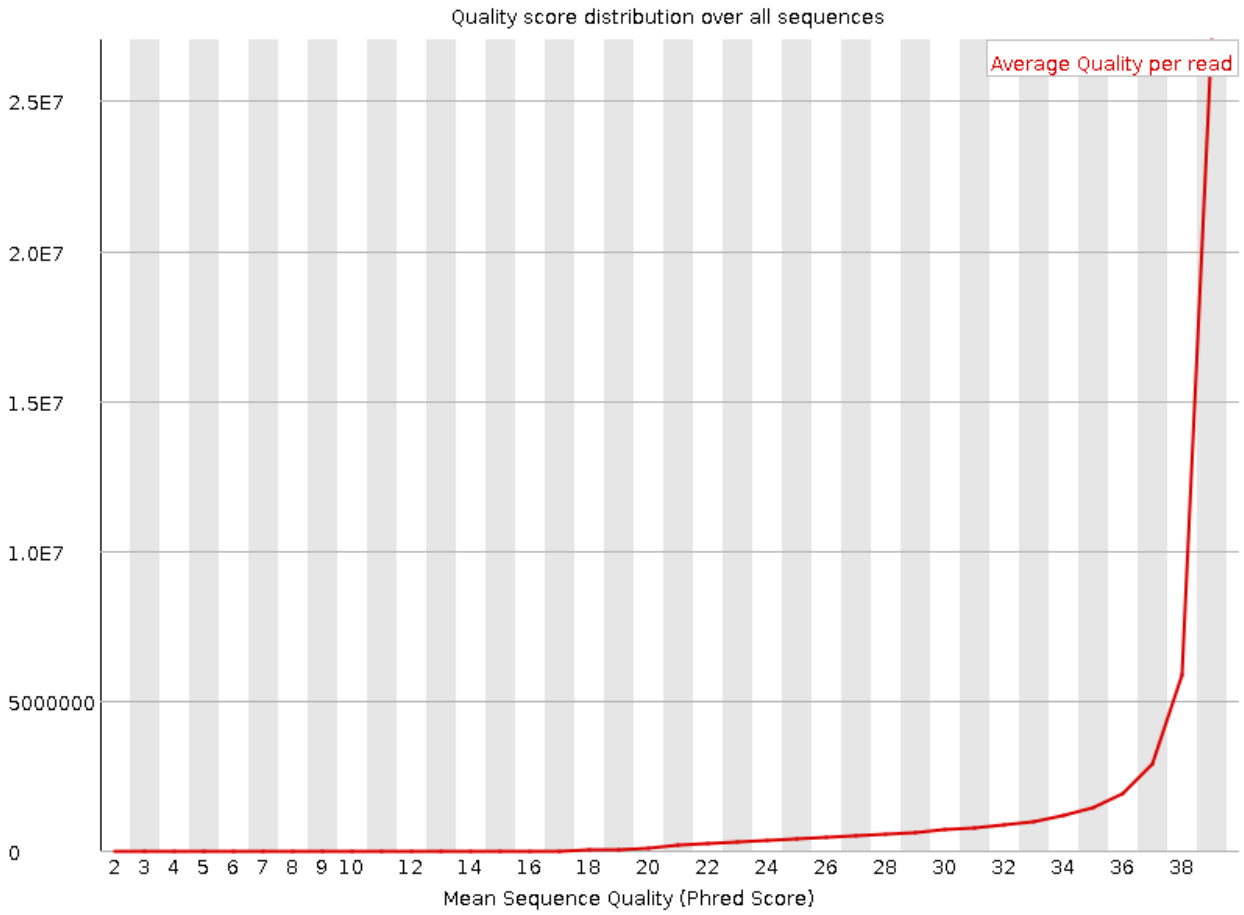


Рис. 23. Per sequence quality scores. Overall чтения очень хорошие, поэтому идем дальше

4.3 Картирование чтений на референс

С помощью программы **hisat2** чтения были картированы на референс (пятую хромосому). Команда:

```
hisat2 -x ../hisat2/$2'_indexed' -U $1.fastq.gz -k 3 -p 8 >rna_cart.sam
2>rna_logs.txt
```

-x – указали путь до индексированного референса

-U $1.fastq.gz – подали на вход непарные чтения

-k 3 – искать не более 3 первичных выравниваний для каждого рида (см. пункт 2.3), причем не гарантируется, что лучших

`>rna_cart.sam` – записывать вывод в файл `rna_cart.sam`

`2>rna_logs.txt` – направляет stderr в файл `rna_logs.txt`

Посмотрим на файл с логами:

```
48692928 reads; of these:
  48692928 (100.00%) were unpaired; of these:
    48661403 (99.94%) aligned 0 times
    31164 (0.06%) aligned exactly 1 time
    361 (0.00%) aligned >1 times
0.06% overall alignment rate
```

Сначала я подумал, что это полный бред и не может быть такого низкого процента картированных чтений, переделал несколько раз – получалось то же самое. На потом я вспомнил, что у нас чтений тотальной РНК, причем из сердечной ткани МЫШИ, а картируем мы на человеческую хромосому (конечно, организмы довольно близкие, но все же). Поэтому, наверное, такой процент вполне адекватный.

Сколько чтений закартировалось на вашу хромосому? 31525 (0,06%)

Далее был запущен следующий программный конвеер (описания команд см. часть 2):

```
#конвертация sam в bam
samtools sort -o rna_cart.bam rna_cart.sam
#индексация
samtools index rna_cart.bam
#анализ bam файлов
mkdir flagstat
samtools flagstat rna_cart.bam > flagstat/rna_cart.txt
#отбор чтений, картированных на референс
samtools view -h -bS rna_cart.bam 5 > rna_cart.chr5.bam
samtools flagstat rna_cart.chr5.bam > flagstat/chr5.txt
```

В итоге получили bam файл с картированными на 5 хромосому чтениями (осталось столько же - 31525)

4.4 Поиск экспрессирующихся генов
Посмотрим на файл генной разметки

Шапка:

`#!genome-build GRCh38.p14`	версия разметки
`#!genome-version GRCh3814`	версия генома
`#!genome-date 2013-1238`	дата публикации
`#!genome-build-accession GCA_000001405.29`	AC разметки
`#!genebuild-last-updated 2023-03001405.29`	последнее обновление

Тело файла состоит из множества строк, соответствующих какому-то объекту - гену/транскрипту/экзону/CDS/5'-нетранслируемой области и тд. Каждая строка состоит из следующих столбцов:

seqname	название хромосомы или скаффолда
source	источник аннотации (ensembl - автоматическая или havana - ручная)
feature	что за фича (ген/транскрипт/экзон/CDS/5'-нетранслируемой область...)
start	координата начала
end	координата конца
score	не понял, что это (наверное что-то о достоверности, но в файле везде точка стоит)
strand	цепь (+/-)
frame	рамка считывания (./1/2)
attribute	дополнительная информация

Сколько на вашей хромосоме аннотировано генов?
Узнал я это с помощью следующей команды:

```
grep '^5'  Homo_sapiens.GRCh38.110.chr.gtf | cut -f3 | grep -c 'gene'
```

Получилось **3074. НО**, наример, псевдогены в третьем столбце аннотированы тоже как gene, хотя вроде бы обычно эти понятия разделяются, поэтому, мне кажется, это число несколько завышено.

```
htseq-count -f bam -s yes -t gene -m union -o pergene.sam rna_cart.bam
Homo_sapiens.GRCh38.110.chr.gtf 1>htseq.txt 2>htseq.logs.txt
```

-f bam – входной файл в формате bam
-s yes – чтения цепь-специфичные
-t – для какой фичи (третий столбец) считать (ну вроде для генов надо)
-m union – подсчитывать простое объединение перекрываемых ридом фич
-o – записать все получившиеся

Посмотрим на получившийся файл `htseq.txt`. В нем каждому гену сопоставлено количество ридов, которые на него попадают. В конце есть сводная статистика:

```
ENSG00000292371 0
ENSG00000292372 0
ENSG00000292373 0
__no_feature    24205          не попали в границы feature (gene)
__ambiguous     359            попали в несколько feature
__too_low_aQual 0              были пропущены (не задавали параметров)
   __not_aligned  0            не выровнены
__alignment_not_unique  361    чтения с больше, чем одним выравниванием
```

Сколько чтений попало в границы генов? 7320 (31525 - 24205)
Сколько чтений попало мимо границ генов? 24205

Как-то очень мало получилось(

4.5 Аннотация высоко экспрессируемых генов

Файл с экспрессионным профилем был отсортирован по убыванию каунтов с помощью следующей команды:

```
sort -k2 -nr htseq.log1.txt >sort.txt
```

топ 10 самых высоко экспрессируемых генов:

```
ENSG00000250974 2068
ENSG00000175471 837
ENSG00000122012 496
ENSG00000245864 339
ENSG00000175309 332
ENSG00000177932 327
ENSG00000272742 270
ENSG00000266751 202
ENSG00000247572 171
ENSG00000171530 143
```

Первый ген оказался не белок-кодирующим))) Поэтому я взял второй - ENSG00000175471. Это оказался Multiple C2 And Transmembrane Domain Containing 1 (MCTP1).

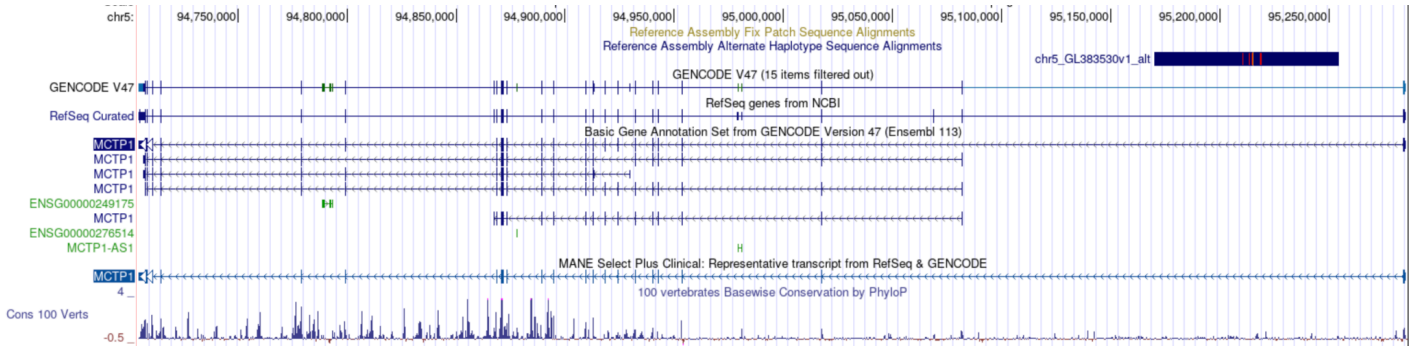


Рис. 24. Скрин из геномного браузера. Снизу - трек консервативности. Есть довольно консервативные участки в первой половине гена (экзоны), что важно, так как иначе чтения из мыши хуже бы картировались на человеческий ген. Кстати, ген на обратной цепи (и чтения тоже).

По информации из GeneCards, белковый продукт этого гена – это кальциевый сенсор, который необходим для стабилизации нормального базового высвобождения нейромедиатора, а также для индукции и долгосрочного поддержания пресинаптической гомеостатической пластичности. И хотя основной акцент делается на его участие в синаптической передаче, этот белок вовлечен в принципе в кальций-опосредованный сигналинг, поэтому неудивительно, что он активно экспрессируется в сердечной ткани. Плюс место его основной локализации - ЭПР, очень хорошо развито в сердечномышечной ткани (саркоплазматический ретикулюм) и выполняет как раз функцию депонирования кальция и обеспечение кальциевого сигналинга (ну там высвобождение ионов кальция, необходимых для сокращения). Вот такие пироги.

OVERALL: В первых частях практикума были найдены и аннотированы варианты одного человека по данным экзомного секвенирования, вроде все нормально получилось. В последней части была предпринята попытка построения экспрессионнного профиля на основе данных РНК-секвенировния. И здесь меня очень смущает такое маленькое количество оставшихся после всех фильтраций чтений (картировавшихся в пределах генов на пятой хромосоме). Но, возможно, учитывая, что это чтения из мышки, так и должно было получится. По крайней мере, белок вполне логичный нашелся. КОНЕЦ.