Все задания практикумов 11-13 выполнялись в папке /mnt/scratch/NGS/yaz008/pr11-13
Все задания практикума 14 выполнялись в папке /mnt/scratch/NGS/yaz008/pr14

Практикум 11

Получение референса

Первым делом я скопировал заданную мне хромосому (11) к себе в папку командой

cp /mnt/scratch/NGS/DATA/hg38/Homo_sapiens.GRCh38.dna.chromosome.11.fa chr11.fa

Индексация для hisat2

Далее я проиндексировал хромосому для hisat2 командой

hisat2-build chr11.fa indexed/chr11

Эта программа принимает файл с референсным геномом и путь по которому будут созданы несколько output-файлов (chr11.N.ht2 for N in 1..8)

Индексация для samtools

Я проиндексировал хромосому для samtools командой

samtools faidx chr11.fa

Эта программа принимает файл name.fa и создаёт файл name.fa.fai, содержащий строку

11 135086622 58 60 61

(номер хромосомы, длина хромосомы, номер байта начала последовательности, количество нуклеотидов в строке, количество байтов в строке)

Описание образца

a. ID: SRR10720402

b. Ссылка: https://www.ncbi.nlm.nih.gov/sra/SRR10720402

c. Прибор: Illumina Genome Analyzer IIx

d. Организм: Homo sapiens

e. Стратегия: whole-exome sequencing (экзомное)

f. Парно-концевые риды

g. Ожидаемое кол-во ридов (spots): 38 530 707

Проверка качества исходных ридов

Риды я скопировал для дальнейшей работы с ними:

cp /mnt/scratch/NGS/DATA/dna_reads/SRR10720402_1.fastq.gz SRR10720402_1f.fq.gz

cp /mnt/scratch/NGS/DATA/dna_reads/SRR10720402_2.fastq.gz SRR10720402_2r.fq.gz

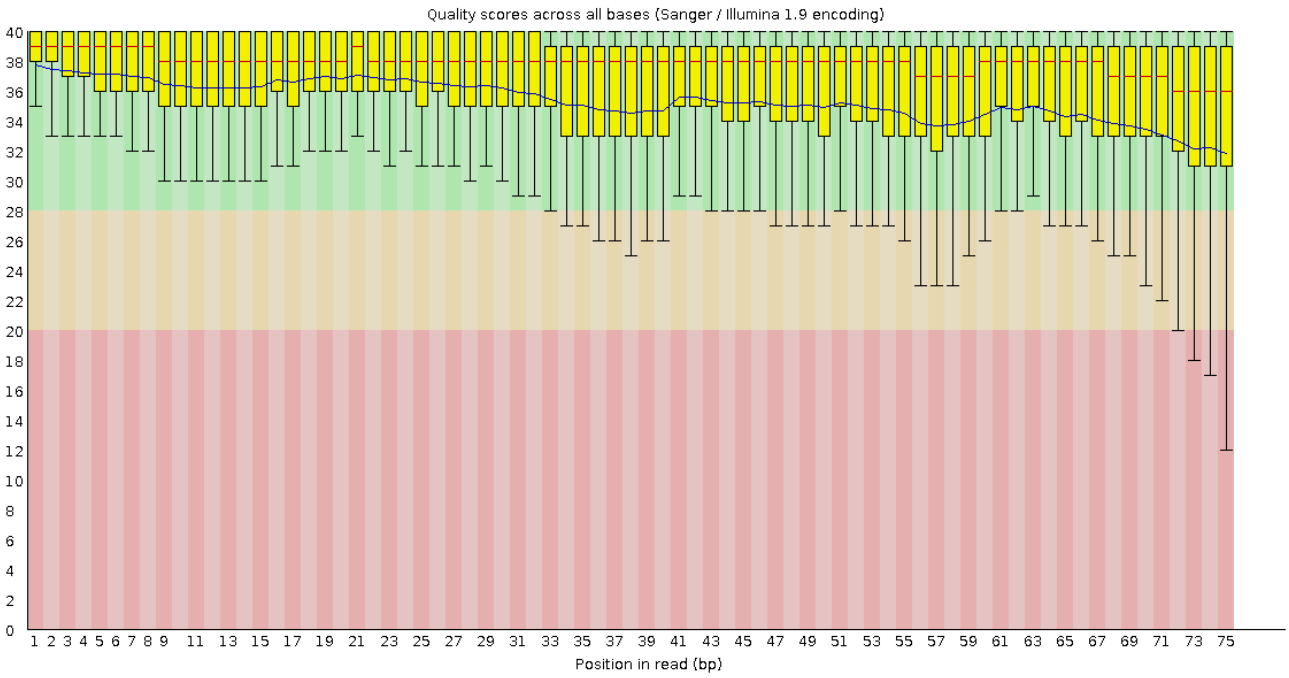
Дальше получил файлы для анализа ридов:

fastqc SRR10720402_1f.fq.gz SRR10720402_2r.fq.gz

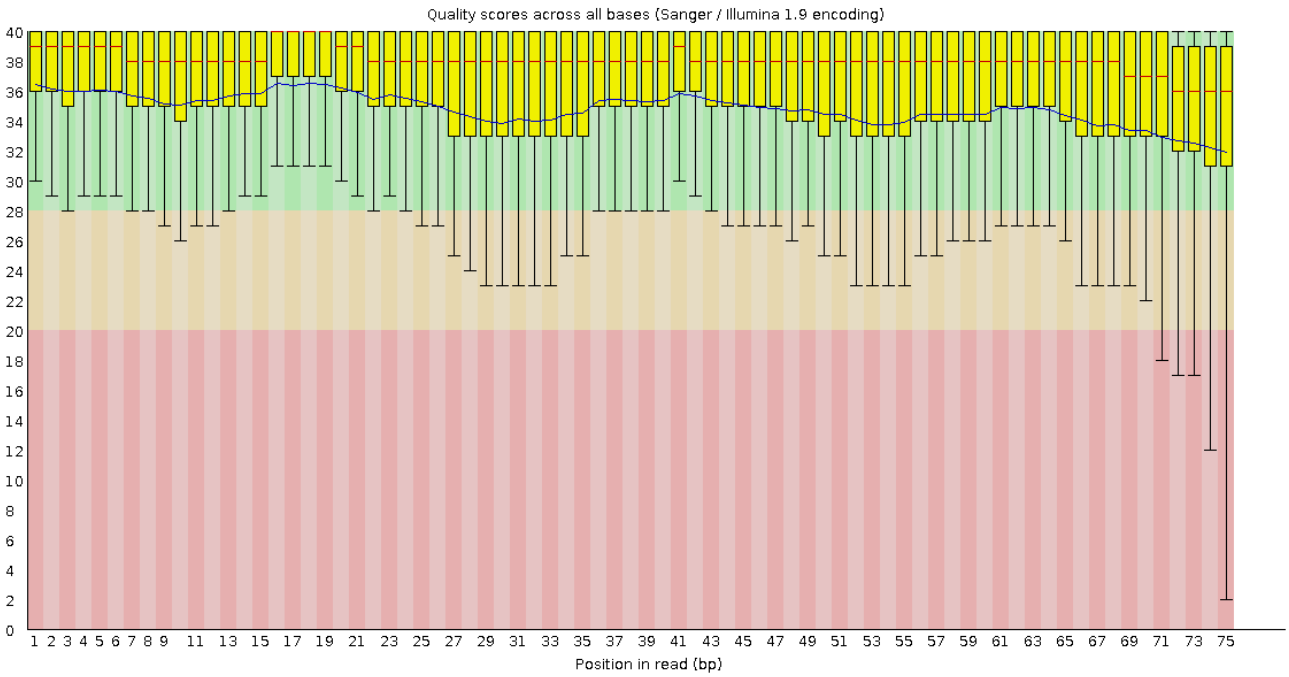
Программа fastqc принимает файлы с ридами name.fq.gz и создаёт файлы вида name_fastqc.zip и name_fastqc.html

a. Риды: 38 530 707
b. Количества прямых и обратных ридов совпадают
c. На изображениях ниже показаны качества прямых и обратных ридов соответственно:

Качество прямых ридов



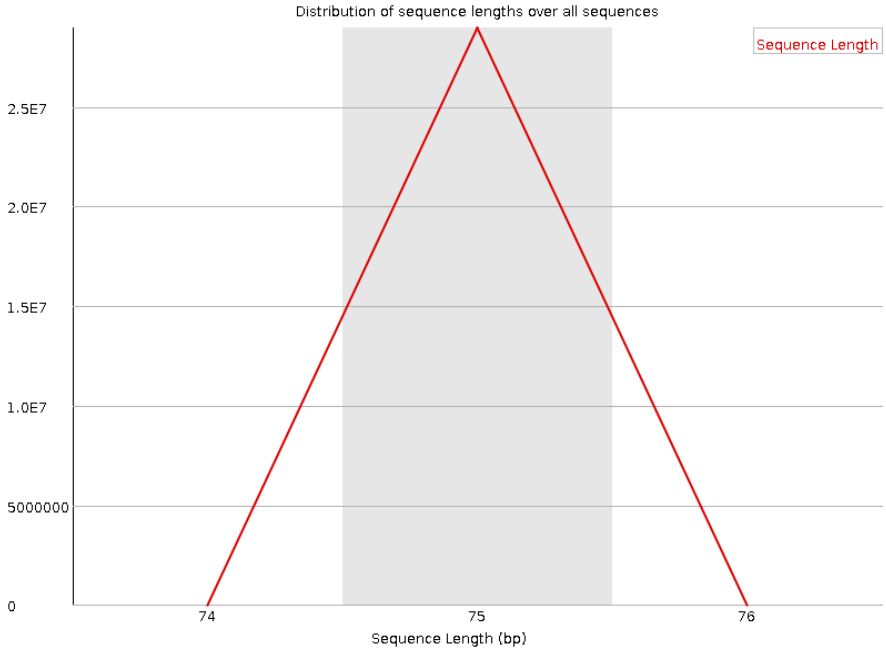
Качество обратных ридов



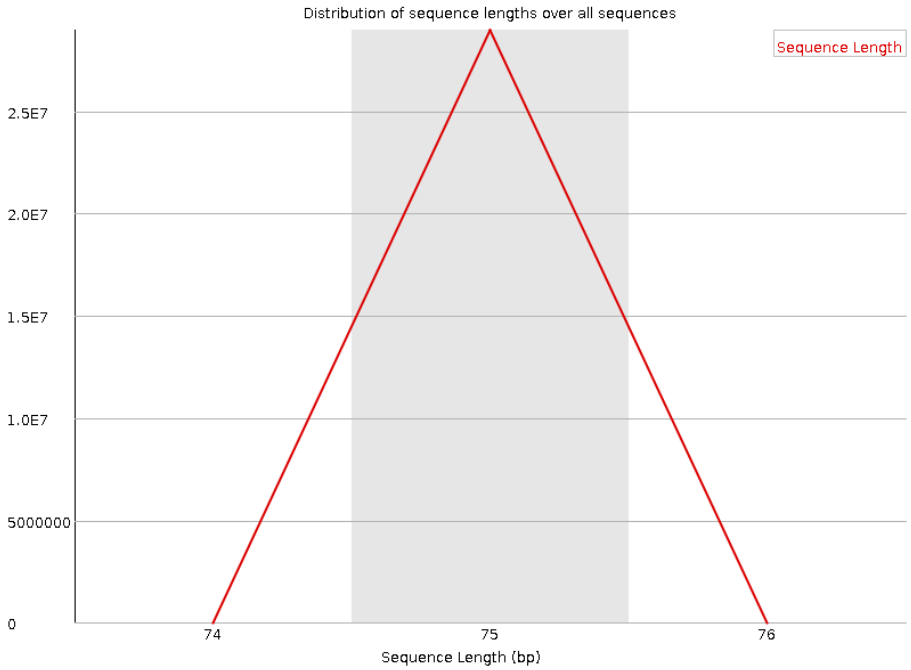
Качество ридов достаточно неплохое, однако чуть-чуть снижается ближе к концу

d. Распределение длин ридов:

Распределение длин прямых ридов



Распределение длин обратных ридов



Длина всех ридов – 75

Фильтрация ридов

Я отфильтровал парно-концевые риды программой TrimmomaticPE командой

TrimmomaticPE -phred33 -trimlog trimlog.txt SRR10720402_1f.fq.gz SRR10720402_2r.fq.gz trimmed_1f_paired.fq.gz trimmed_1f_unpaired.fq.gz trimmed_2r_paired.fq.gz trimmed_2r_unpaired.fq.gz TRAILING:20 MINLEN:50

Эта программа принимает файлы с прямыми и обратными ридами. Параметр TRAILING: 20 отвечает за удаление ридов с качеством ниже 20, MINLEN: 50 за удаление ридов длиной меньше 50.

В результате работы TrimmomaticPE получается 4 файла. Поскольку для получения парно-концевых ридов нужно сохранить оба рида в паре, я буду в дальнейшем использовать только файлы trimmed_1f_paired.fq.gz и trimmed_2r_paired.fq.gz, а trimmed_1f_unpaired.fq.gz и trimmed_2r_unpaired.fq.gz могут содержать риды, пара которых была удалена.

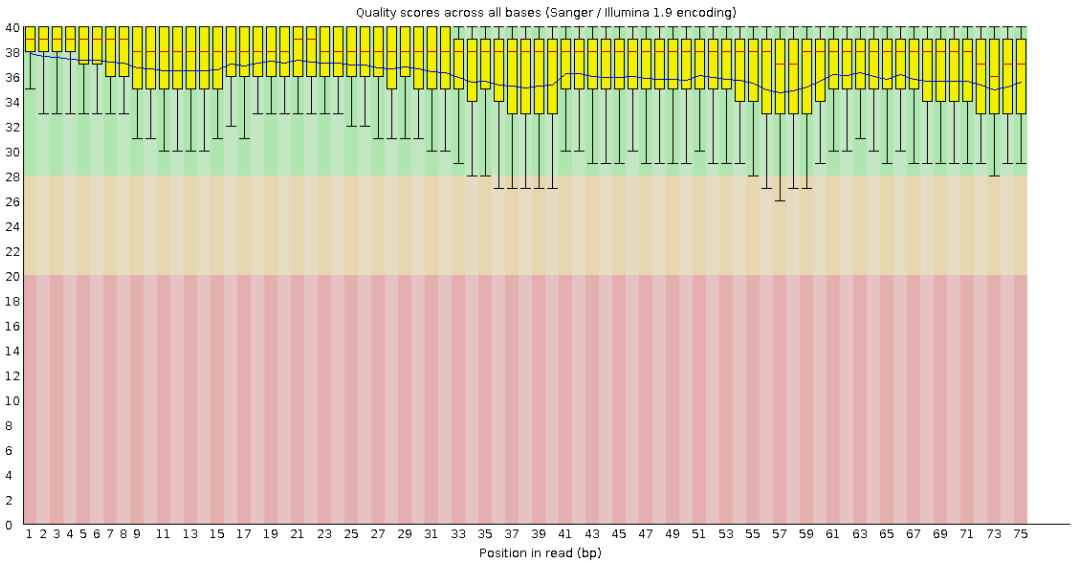
Качество триммированых ридов

Для проверки качества ридов я запустил fastqc trimmed *

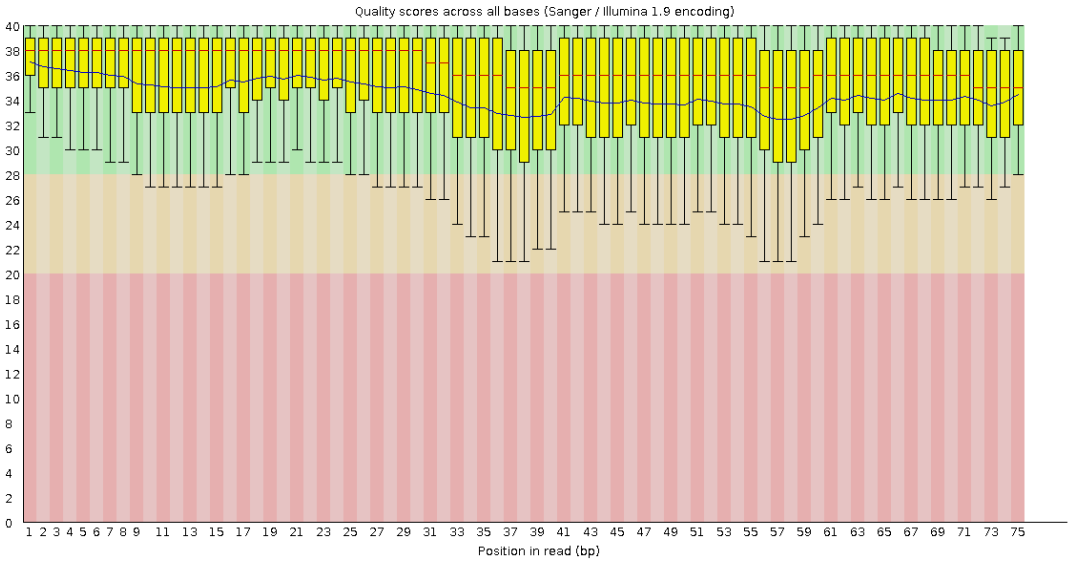
Результаты:

a. Количество ридов: 27172718
b. Осталось % ридов: 93.81
c. На изображениях ниже показаны графики качества пар прямых парных, прямых непарных, обратных парных и обратных непарных ридов соответственно:

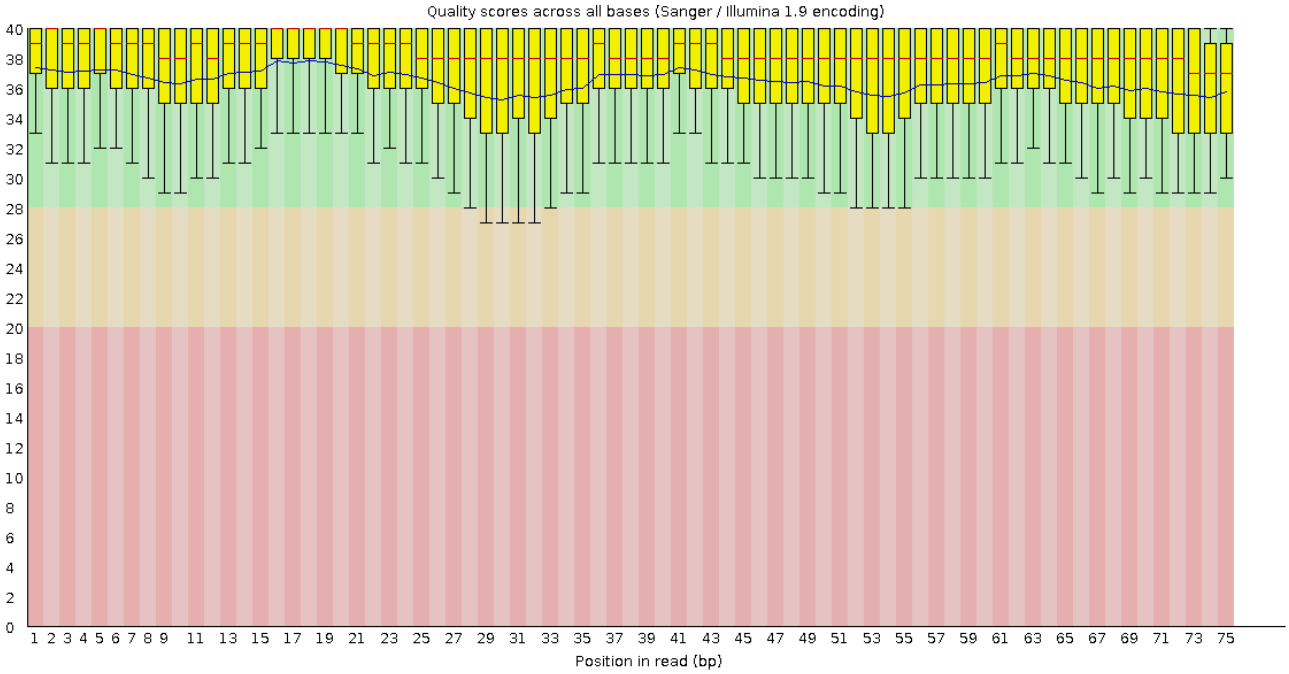
Качество прямых парных ридов



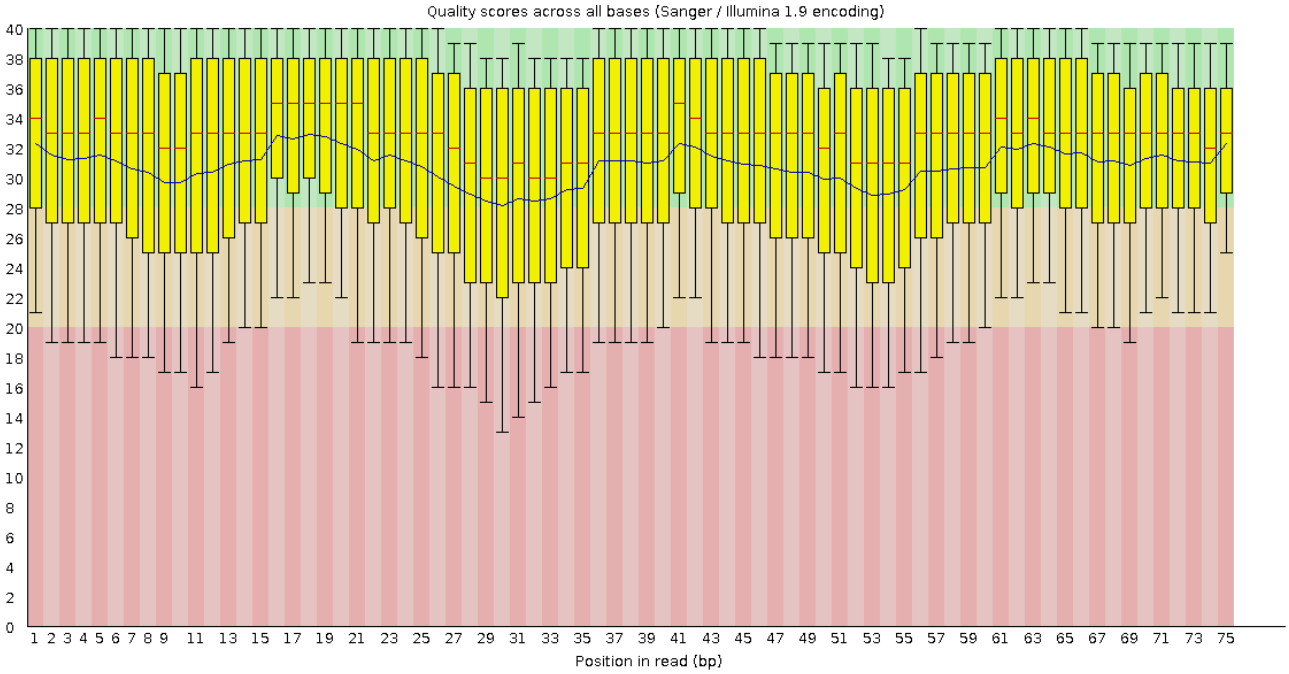
Качество прямых непарных ридов



Качество обратных парных ридов



Качество обратных непарных ридов

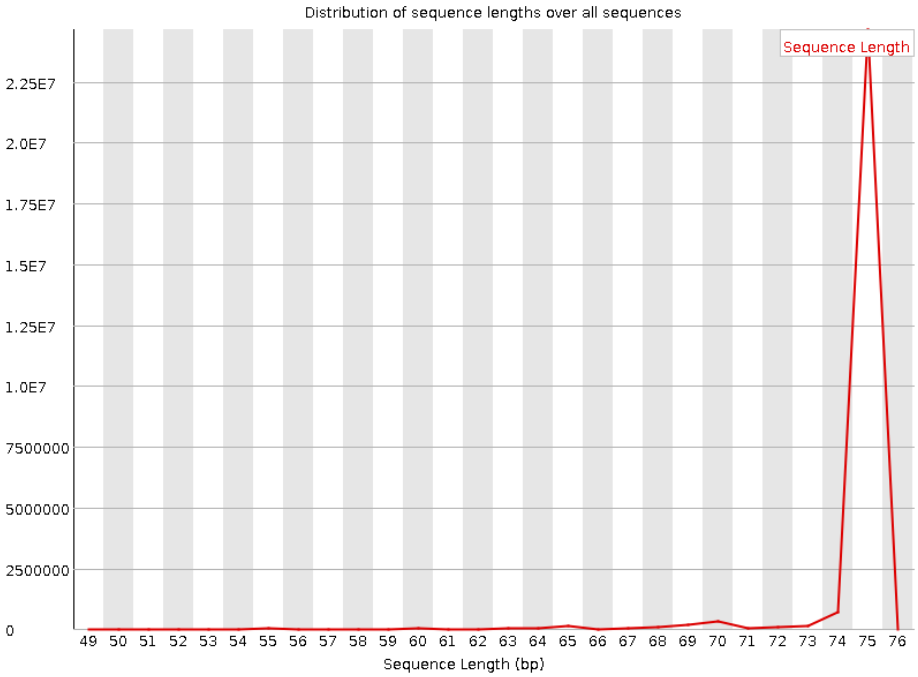


Качество прямых ридов хорошее, качество обратных непарных ридов пониже

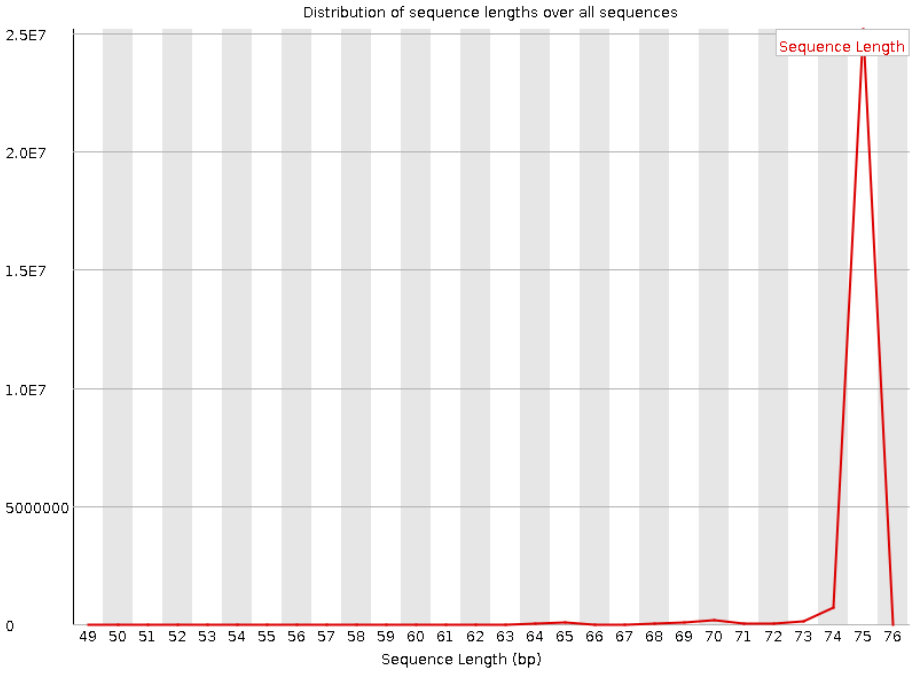
d. Качество ридов несколько возросло после триммирования

e. На следующих изображениях показаны распределения длин парных ридов:

Распределение длин прямых парных ридов



Распределение длин прямых обратных ридов



После триммирования появились риды с длиной меньше 75

Практикум 12

Картирование ридов на референсный геном

Я картировал риды на референсный геном следующей командой

hisat2 -x ../indexed/chr11 -1 ../reads/trimmed_1f_paired.fq.gz -2 ../reads/trimmed_2r_paired.fq.gz -p 4 --no-spliced-alignment -S paired.sam 2> hisat2_err.log

Значения параметров hisat2:

1. −x base_file_path: принимает имя базового файла (chr11)
2. −1 forward_reads. fq. gz: принимает файл с прямыми ридами
3. −2 backward_reads. fq. gz: принимает файл с обратными ридами
4. −p 4: количество потоков
5. −S output_file_name. sam: имя выходного файла
6. − − no − spliced − alignment: отключение сплайсинга

Конвертация в bam

Я конвертировал файл в бинарный формат при помощи команды

samtools sort -o paired.bam paired.sam

Программа принимает .sam файл и записывает результат в .bam файл, имя которого указано параметром -o

a. Файл .sam весит 11.04 Gb
b. Файл .bam весит 3.42 Gb

Индексация bam файла

samtools index paired.bam

Индексатор принимает имя .bam файла и возвращает бинарный проиндексированный файл .bam.bai

Анализ bam файла

Я получил файл для анализа следующей командой

samtools flagstat paired.bam > paired_flagged.txt

Программа принимает имя .bam файла и пишет свой output в stdout

a. Mapped: 2987088 (5.46%)
b. Properly paired: 2220406 (4.09%)

Получение ридов, картированных на хромосому

Я получил риды, картированные на мою хромосому

samtools view -h -bS paired.bam 11 > paired_chr11.bam

Параметры картировщика:

a. –h: включить хэдер
b. –b name.bam: имя .bam файла
c. –S: автоматическое определение формата input'а
d. 11: номер хромосомы

Получение только правильно картированных ридов

samtools view -f 0x2 -bS paired_chr11.bam > paired_chr11_proper.bam

a. –f 0x2: флаг PROPER_PAIR
b. –b name.bam: имя .bam файла
c. –S: автоматическое определение формата input'а

Получение файла для анализа

samtools flagstat paired_chr11_proper.bam > paired_chr11_proper_flagged.txt

Флагстат принимает имя .bam файла и выводит флаги в stdout

Properly paired: 2220406 (100.00%)

Дальше я проиндексировал .bam файл командой

samtools index paired_chr11_proper.bam

Практикум 13

Получение вариантов

Были получены варианты командой

bcftools mpileup -f ../chr11.fa ../mapped/paired_chr11_proper.bam | bcftools call -mv -o paired_chr11_proper.vcf

a. На вход mpileup подается имя .bam файла (опция -f)

b. mpileup генерирует файл формата .vcf с вероятностями вариантов

c. Программа call анализирует созданный .vcf файл и ищет конкретные варианты

e. Имя файла для вывода результатов (опция -o)

f. call использует дефолтный метод поиска (опция -m), выводит сайты вариантов в stdout (опция -v)

Структура vcf файла

Заголовок (строки, начинающиеся на ##) и аннотации столбцов таблицы (#)

Дальше я получил файл для анализа командой

bcftools stats paired_chr11_proper.vcf > paired_chr11_proper_stats.txt

Программа принимает имя .vcf файла и пишет свой output в stdout

a. Variants: 52209
b. SNPs: 52209
c. Indels: 1339

Фильтрация вариантов

Далее я отфильтровал варианты командой

bcftools filter -i "%QUAL>30 && DP>50" paired_chr11_proper.vcf > paired_chr11_proper_filtered.vcf

Программа принимает имя .vcf файла, строковый параметр -i задаёт критерии отбора (QUAL – качество, DP – длина) и выводит результат в stdout

Затем я получил файл для анализа

bcftools stats paired_chr11_proper_filtered.vcf > paired_chr11_proper_filtered_stats.txt

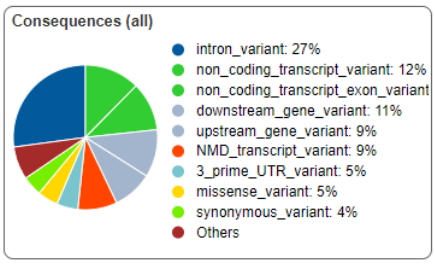
Программа принимает имя .vcf файла и пишет свой output в stdout

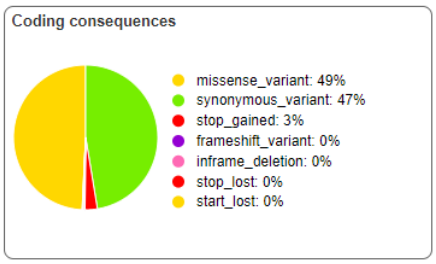
a. Variants: 1418
b. SNPs: 1399
c. Indels: 19

Аннотация вариантов

При помощи сервиса VEP был проанализирован файл paired_chr11_proper_filtered_stats. txt. Результаты из раздела Summary statistics приведены на изображении ниже

Category	Count
Variants processed	1418
Variants filtered out	0
Novel / existing variants	350 (24.7) / 1068 (75.3)
Overlapped genes	537
Overlapped transcripts	2061
Overlapped regulatory features	95





a. Variants with HIGH IMPACT: 78
b. Variants with MODIFIER IMPACT: 7484
c. Variants with LOW IMPACT: 775
d. Variants with MODERATE IMPACT: 540

Практикум 14

Описание образца

a. ID: ENCFF975AUW

b. Ссылка: https://www.encodeproject.org/files/ENCFF975AUW/

c. Организм и ткань: Homo sapiens heart tissue male embryo (120 days)

d. Стратегия: polyA plus RNA-seq

e. Риды: одноконцевые

f. Цепь-специфичность: unstranded

Проверка качества исходных чтений

Я скопировал риды в папку /mnt/scratch/NGS/yaz008/pr14/reads

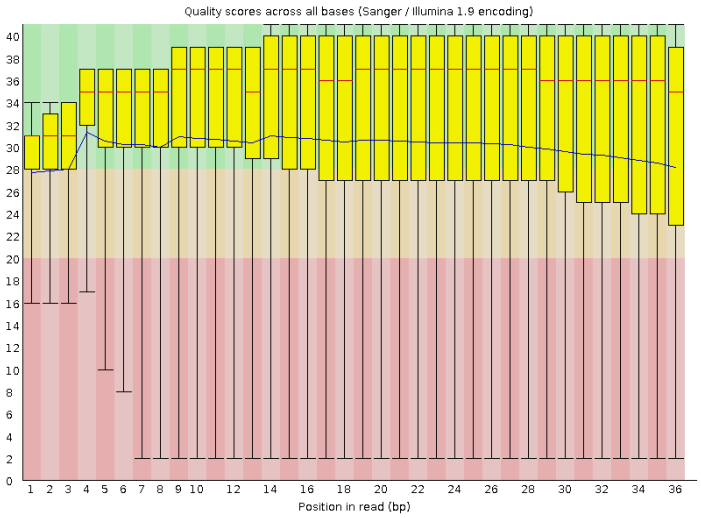
cp /mnt/scratch/NGS/DATA/rna_reads/ENCFF975AUW.fastq.gz ENCFF975AUW.fq.gz

Затем я получил файлы для анализа качества ридов

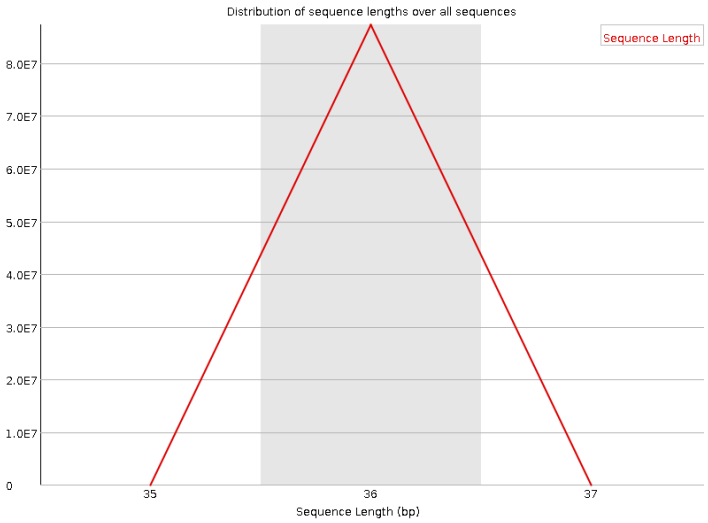
fastqc ENCFF975AUW.fq.gz

Фасткьюси принимает имя файла с ридами name.fq.gz и создаёт файлы name_fastqc.zip и name_fastqc.html

a. Reads: 87265266
b. Качество ридов не очень высокое, что видно на изображении ниже



c. Длина всех ридов: 36



Картирование ридов на референс

hisat2 -x ../../pr11-13/indexed/chr11 -k 3 -U ../reads/ENCFF975AUW.fq.gz -S rna.sam 2> hisat2_err.log

Значения параметров hisat2:

1. –x base_file_path: принимает имя базового файла (chr11)
2. –U reads. fq. gz: принимает файл с ридами
3. –k 3: программа будет искать по 3 выравнивания с лучшими score для каждого рида
4. –S output_file_name. sam: имя выходного файла

Ридов закартировалось: 5263918 + 672431 = 5936349 (6.80%)

Далее я получил бинарный .bam файл при помощи команды

samtools sort -o rna.bam rna.sam

Затем я проиндексировал его командой

samtools index rna.bam

И после этого я картировал риды на выделенную мне хромосому (11)

samtools view -h -bS rna.bam 11 > rna_chr11.bam

Новый полученный .bam файл был проиндексирован:

samtools index rna_chr11.bam

И при помощи флагстата я получил файл для дальнейшего анализа

samtools flagstat rna_chr11.bam > rna_chr11_flagged.txt

Поиск экспрессирующихся генов

Я скопировал файл генной разметки

cp /mnt/scratch/NGS/DATA/genes/Homo_sapiens.GRCh38.110.chr.gtf marking/marking.gtf

.gtf файл содержит заголовок (строки начинаются с #) и таблицу особенностей

После этого для каждого гена я посчитал количество картированных на него ридов

htseq-count -f bam -s no -m union -t exon -o marked_rna_chr11.sam rna_chr11.bam ../marking/marking.gtf 1> marked_rna_chr11.txt 2> htseq_count_err.log

Значения параметров hisat-count:

1. –o: принимает имя .bam файла
2. –f bam: расширение входного файла
3. –t exon: тип гена из разметки (exon'ы в данном случае)
4. –s no: читает риды и с прямой, и с обратной цепей
5. –m union: объединять перекрывающиеся риды

Из выходного файла видно, что

a. В нужные гены не попало 1216050 ридов (__no_feature)
b. В несколько генов попало 567732 ридов (__ambiguous)
c. Соответствующий ген не определён однозначно для 672431 ридов (__alignment_not_unique)

В границы генов попало ридов: 3480136

Подготовка программного сценария

```
#!/bin/bash

# Input:

if [[ "$1" == "-h" ]] || [[ "$1" == "--help" ]]; then

 echo "Usage: ./script.sh ID N" && exit 0

fi

ID=$1

N=$2

# Check dir:

[ -d for_script ] && echo "Error: directory name must be \"for_script\"" && exit 1

mkdir for_script

cd for_script # for_script

# Create "ref" dir:

mkdir ref

cd ref # for_script/ref

# Copy the chromosome into "for_script" dir:

cp /mnt/scratch/NGS/DATA/hg38/Homo_sapiens.GRCh38.dna.chromosome.${N}.fa chr${N}.fa

# Indexing:

mkdir indexed

hisat2-build chr${N}.fa indexed/chr$N

samtools faidx chr${N}.fa

cd .. # for_script

# Reads:

mkdir reads

cd reads # for_script/reads

cp /mnt/scratch/NGS/DATA/dna_reads/${ID}_1.fastq.gz ${ID}_1f.fq.gz # Copy forward reads

cp /mnt/scratch/NGS/DATA/dna_reads/${ID}_2.fastq.gz ${ID}_2r.fq.gz # Copy backward reads

fastqc ${ID}_1f.fq.gz ${ID}_2r.fq.gz

# Trimming:

TrimmomaticPE -threads 4 -phred33 -trimlog trimlog.txt ${ID}_1f.fq.gz ${ID}_2r.fq.gz trimmed_1f_paired.fq.gz
trimmed_1f_unpaired.fq.gz trimmed_2r_paired.fq.gz trimmed_2r_unpaired.fq.gz TRAILING:20 MINLEN:50
```

```
fastqc trimmed*

cd .. # for_script

mkdir mapped

cd mapped # for_script/mapped

hisat2 -x ../ref/indexed/chr$N -1 ../reads/trimmed_1f_paired.fq.gz -2 ../reads/trimmed_2r_paired.fq.gz -p 4 --no-spliced-alignment -S paired.sam 2> hisat2_err.log

samtools sort -o paired.bam paired.sam

# !!!: Should remove "paired.sam"

samtools index paired.bam

samtools flagstat paired.bam > paired_flagged.txt

samtools view -h -bS paired.bam $N > paired_chr${N}.bam

samtools view -f 0x2 -bS paired_chr${N}.bam > paired_chr${N}_proper.bam

samtools flagstat paired_chr${N}_proper.bam > paired_chr${N}_proper_flagged.txt

samtools index paired_chr${N}_proper.bam

cd .. # for_script

# Variants:

mkdir variants

cd variants # for_script/mapped

bcftools mpileup -f ../ref/chr${N}.fa ../mapped/paired_chr${N}_proper.bam | bcftools call -mv -o paired_chr${N}_proper.vcf

bcftools stats paired_chr${N}_proper.vcf > paired_chr${N}_proper_stats.txt

bcftools filter -i '%QUAL>30 && DP>50' paired_chr${N}_proper.vcf > paired_chr${N}_proper_filtered.vcf

bcftools stats paired_chr${N}_proper_filtered.vcf > paired_chr${N}_proper_filtered_stats.txt

echo "Output: ./for_script/variants/paired_chr${N}_proper_filtered.vcf"

less ./paired_chr${N}_proper_filtered.vcf
```