1. **FASTQC** программа для определения качества прочтений на Иллюмине

conda install -c bioconda fastqc - установить FAST QC

fastqc -h получить информацию, убедиться в установке

на вход принимаются файлы формата .fastq. Для запуска надо указать пути до файла (-ов)

fastqc -o. <путь файла 1> <путь файла 2>

Пример:

fastqc -o. /home/card/Meta/raw\_data/amp\_res\_1.fastq

/home/card/Meta/raw\_data/amp\_res\_2.fastq

На выходе будет html файл, который можно открыть в браузере и посмотреть результат

1. **Trimmomatic** Программа для отсева прочтений плохого качества, обрезки адаптеров, фильтрации прочтений по длине и другим показателям. Функционирует в двух режимах: SE для односторонних прочтений и PE для парных прочтений.

conda create -n trimmomatic trimmomatic - установка в отдельное окружение

java -jar PE <файл1.fastq> <файл2.fastq> <тримминг.fq> ОПЦИИ

Возможные параметры запуска тримминга, согласно Trimmomatic manual:

* ILLUMINACLIP. Удалить адаптеры или другие специфичные для Иллюмины сиквенсы из ридов, требуется файл с информацией о последовательностях адаптеров;
* SLIDINGWINDOW: тримминг по методу скользящего окна. Обрезает риды, Если среднее значение качества в пределах окна ниже заданного;
* MAXINFO: автоматическая опция фильтрации по качеству прочтения;
* LEADING: обрезает начало ридов при качестве прочтения ниже заданного;
* TRAILING: обрезает конец ридов при качестве прочтения ниже заданного;
* CROP: обрезает риды до определенной длины, обрезая нуклеотиды с конца;
* HEADCROP: обрезает заданное число нуклеотидов в начале рида;
* MINLEN: вырезает риды, не достигающие заданной длины;
* AVGQUAL: вырезает риды ,среднее качество которых ниже заданного уровня;
* TOPHRED33: конвертирует оценку качества в систему Phred-33;
* TOPHRED64: конвертирует оценку качества в систему Phred-64.

Пример:

java -jar /Users/margaritatis/miniconda3/envs/trimmomatic/share/trimmomatic-0.39-2/trimmomatic.jar PE /Users/margaritatis/Documents/itmo/meta/align/raw/amp\_res\_1.fastq /Users/margaritatis/Documents/itmo/meta/align/raw/amp\_res\_2.fastq /Users/margaritatis/Documents/itmo/meta/align/raw/1tr.fq

/Users/margaritatis/Documents/itmo/meta/align/raw/1utr.fq /Users/margaritatis/Documents/itmo/meta/align/raw/2tr.fq

/Users/margaritatis/Documents/itmo/meta/align/raw/2utr.fq

LEADING:20 TRAILING:20 MINLEN:20 SLIDINGWINDOW:10:20

В качестве вывода при запуске в режиме парных прочтений получены FASTA-файлы. Два из них (в примере 1tr.fq и 2tr.fq) - это форвард- и реверс-прочтения, прошедшие тримминг, в случае которых как форвард, так и соответствующий ему реверс, прошли фильтрацию. Ещё два (1utr.fq и 2utr.fq) - это риды, которые содержат риды, которые прошли фильтрацию, в то время как соответствующие им форвард- или реверс-риды - нет.

1. Выравнивание и компрессия полученных файлов

**BWA-MEM** - это программа для выравнивания, использующая алгоритм преобразования Барроуза-Уилера, или Burrows-Wheeler Transform (BWT). Эта программа оптимизирована для длинных ридов (100 пар нуклеотидов и более), полученных в результате секвенирования второго поколения, которые могут содержать несколько замен, вставок или делеций. Программа ищет фрагменты прочтений, максимально точно совпадающие с референсом, после чего распространяет эти фрагменты на все прочтение при помощи локального выравнивания.

Помимо BWA, на это этапе потребуется samtools - инструмент для чтения и обработки файлов формата sam.

conda install bwa  
conda install -c bioconda bwa

conda install samtools

bwa index - индексировать FASTA-файлы

Пример индексации файла:

bwa index GCF\_000005845.2\_ASM584v2\_genomic.fna

bwa mem <референс.fna> <форвард.fastq> <реверс.fastq> > <выравнивание.sam>

Пример выравнивания:

bwa mem GCF\_000005845.2\_ASM584v2\_genomic.fna amp\_res\_1.fastq amp\_res\_2.fastq > aligment.sam выравнивает 2 файла на референс

**samtools view** **-S -b** alignment.sam **>** alignment.bam - сжать .sam файл в .bam

**samtools flagstat** alignment.bam - получить базовую статистику по bam файлу.

**samtools mpileup** **-f** [reference fasta file] alignment\_sorted.bam **>** my.mpileup - mpileup - это формат, который объединяет в себе информацию по ридам, собирая информацию о позициях, которые соответствуют и не соответствуют референсу.

Применение этого этапа:

samtools sort alignment.bam -o alignment\_sorted.bam

samtools index alignment\_sorted.bam

samtools mpileup -f GCF\_000005845.2\_ASM584v2\_genomic.fna alignment\_sorted.bam > my.mpileup

1. **VarScan**

VarScan - это одна из программ для поиска замен в выровненных прочтениях.

conda install -c bioconda varscan - установка через conda

*mpileup2snp* - функция, которая извлекает из mpileup-файла информацию о заменах.

На этом этапе существует несколько путей отфильтровать полученные данные. Перечень возможных опций, согласно VarScan User's Manual:

* min-coverage - установить минимальную глубину прочтения, необходимую, чтобы выделить замену;
* min-reads2 - минимальное число прочтений замены, необходимое, чтобы выделить замену;
* min-avg-qual - минимальное качество прочтения в позиции замены, чтобы выделить замену;
* min-var-freq - порог минимальной встречаемости альтернативной аллели, например;
* min-freq-for-hom - порог минимальной встречаемости, чтобы определить образец как гомозиготу в диплоидных организмах;
* p-value - порог уровня значимости для подтверждения достоверности замены;
* strand-filter- игнорировать прочтения с поддержкой >90% на одной последовательности;
* output-vcf - вывод в формате vcf;
* variants - вывод только информации о найденных заменах.

Пример использования:

java -jar /Users/margaritatis/miniconda3/envs/trimmomatic/share/varscan-2.4.6-0/VarScan.jar mpileup2snp my.mpileup --min-var-freq 0.2 --variants --output-vcf 1 > VarScan\_results2.vcf

1. **SnpEff**

snpEff - инструмент для автоматической аннотации полученных на предыдущем этапе замен.

В качестве вводных данных требуется vcf файл с результатами поиска замен (полученный в результате применения VarScan), а также база данных, которую программа для автоматической аннотации будет использовать в качестве референса. Для создания этой базы данных можно использовать формат GenBank, данные, которые содержат в себе сиквенс и аннотацию к нему.

Шаги проведения автоматической аннотации:

1. Создать файл config, содержащий информацию об используемом геноме;
2. Создать отдельную директорию для базы данных;
3. Сохранить в эту директорию gbff-файл;
4. Создать базу данных командой snpEff build;
5. Проаннотировать командой snpEff ann <название базы данных > <vcf с результатами поиска замен> > <вывод файла vcf>

Пример использования:

touch snpEff.config

echo "k12.genome : ecoli\_K12" > snpEff.config

cp GCF\_000005845.2\_ASM584v2\_genomic.gbff data/k12/genes.gbk

snpEff build -genbank -v k12

snpEff ann k12 VarScan\_results2.vcf > VarScan\_results\_annotated1.vcf