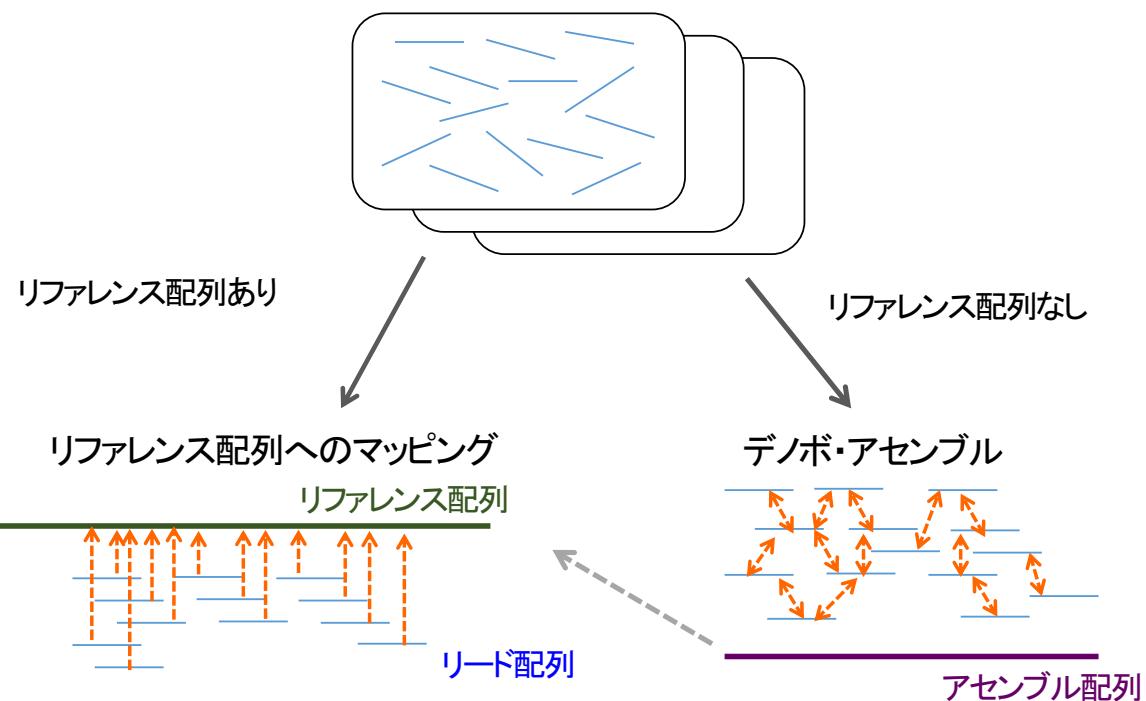


クオリティコントロールと NGS基本ツール

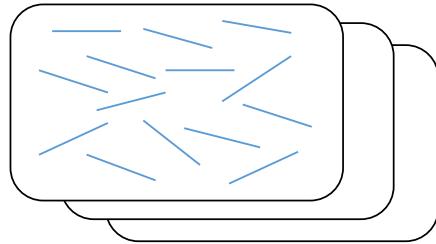
基礎生物学研究所
生物機能解析センター
山口勝司

NGSデータ処理の概要

シーケンスリード(DNA/RNA由来)



シーケンスリード(DNA/RNA由来)



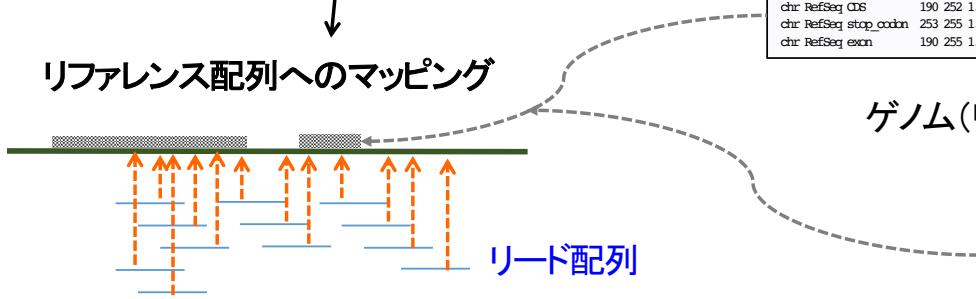
FASTQファイル(配列+クオリティ)

```
@SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:3625:88631 length=51
AICCGCTTGGCCGACCGCUCAGTCAGGTTGGGGGAAATCAACCTGGTGAG
+SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:3625:88631 length=51
@@@D>DFF7DC7FEEF@FTII<DF@AAA2AEFBDBCA>A/B>B:>
+SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:3871:88513 length=51
CAAGGTTGAGTCACAGCAGTCAGGTTGGGGAAATCAACCTGGTGAG
+SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:3871:88513 length=51
COOPEDDEHDEHIIIBGHUUUUGFHGGHGGHGLIDGLHHGGHHH
+SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:3950:88530 length=51
CAGGCAATGCCCTTCACTGTTACACTTGACCAACCTGATTTCAG
+SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:3950:88530 length=51
COOPEDDEHDEHIIIBGHUUUUGFHGGHGGHGLIDGLHHGGHHH
```

リファレンス配列あり



リファレンス配列へのマッピング



遺伝子アノテーション GFF(GTF)ファイル

```
chr RefSeq start_codon 190 192 1.000 + . gene_id "b0001"; transcript_id "b0001";
chr RefSeq CDS 190 252 1.000 + 0 gene_id "b0001"; transcript_id "b0001";
chr RefSeq stop_codon 253 255 1.000 + . gene_id "b0001"; transcript_id "b0001";
chr RefSeq exon 190 255 1.000 + . gene_id "b0001"; transcript_id "b0001";
```

ゲノム(リファレンス)配列

FASTAファイル

```
>chr
AGTTTGTTCGAGCAGCAGGAAAGTGCT
CTGGCTGGTTAAANAAAAGAGCTCTGTTGCGC
TTCTGAGCTGTTGCCCTCCCTGAGTAATTTAA
TTTTTTGACTTGGCTACTTAACTTCTAACAA
TTGGGGGGGGGGGAGCGCAGTGAATAATTTGAG
AGTTCAGACATCTGGAGAGGTTTGCCTCCNCC
ATTCACACCCCATCTGGAGGTTTGCCTCCNCC
```

| CHD | VN:1.0 | SO:unsorted |
|----------------|------------|-------------|
| BSQ | SN:chr | IN:4639675 |
| @FG | ID:bowtie2 | PN:bowtie2 |
| SRR1515276.40 | 0 chr | 4423609 |
| SRR1515276.158 | 16 chr | 501700 |
| SRR1515276.212 | 4 * | 0 |
| SRR1515276.319 | 0 chr | 2922768 |
| SRR1515276.367 | 16 chr | 2753873 |
| SRR1515276.411 | 0 chr | 3440721 |
| SRR1515276.434 | 0 chr | 4198737 |

マッピング結果 SAM ファイル

クオリティーコントロール

NGSデータ解析におけるクオリティーコントロールの重要性

- ・作製したライブラリーに問題はなかったか
→**検証する手段**

アダプター配列ばかりではないか
コンタミ配列の有無
PCR増幅の適性度
短いライブラリーほどクラスター増幅されやすい
GC率に偏りがあるものは増幅されにくい

- ・得られるdataのクオリティーは同一ではない
→**可能な範囲で揃える**

シークエンサーの調子 エアーかみ
作製ライブラリーのサイズ分布
シークエンサー間の性能差

シークエンスのクオリティーチェックツール FASTQC

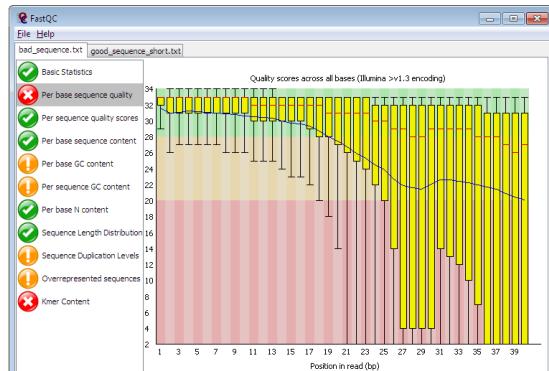
Babraham Bioinformatics

About | People | Services | Projects | Training | Publications

FastQC

| | |
|-----------------|--|
| Function | A quality control tool for high throughput sequence data. |
| Language | Java |
| Requirements | A suitable Java Runtime Environment The Picard BAM/SAM Libraries (included in download) |
| Code Maturity | Stable. Mature code, but feedback is appreciated. |
| Code Released | Yes, under GPL v3 or later . |
| Initial Contact | Simon Andrews |

[Download Now](#)



Documentation

A [copy of the FastQC documentation](#) is available for you to try before you buy (well download.).

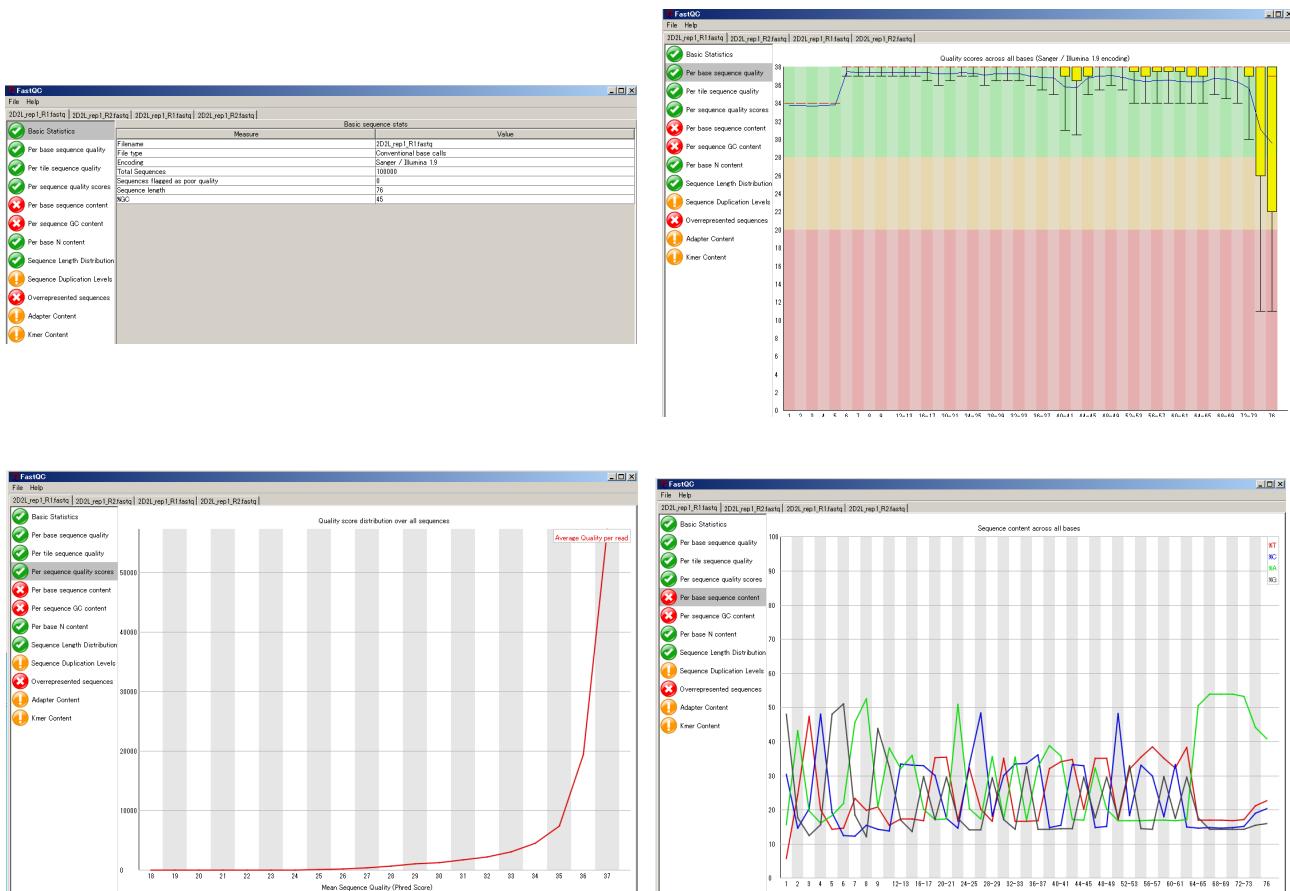
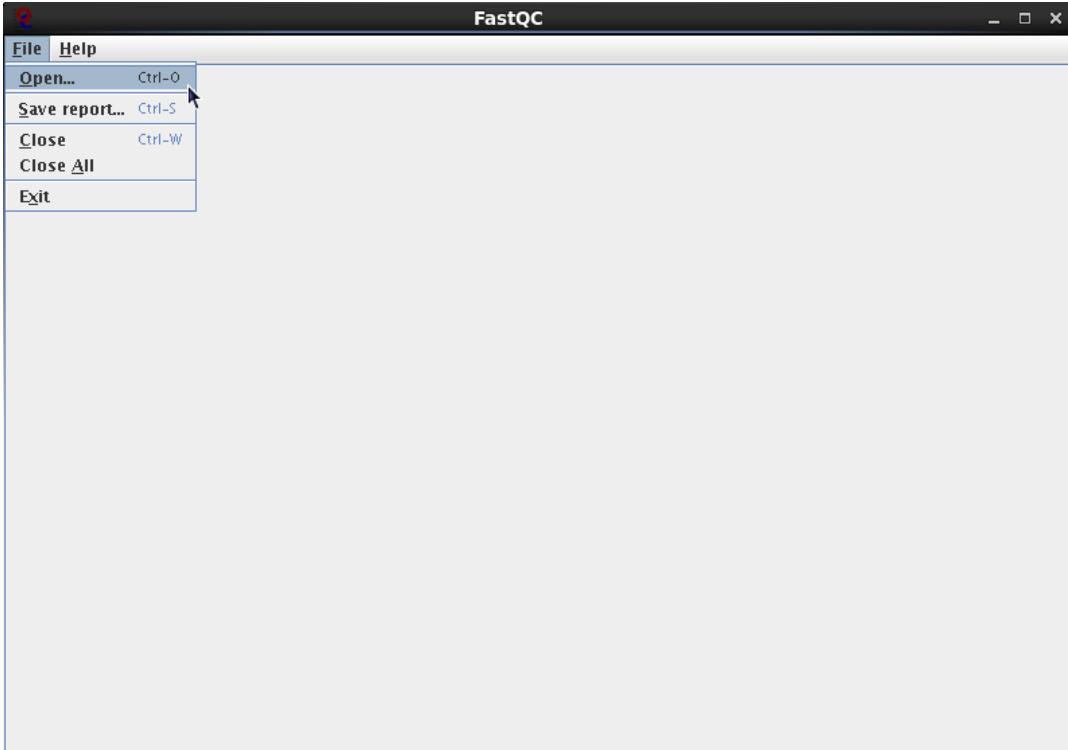
Example Reports

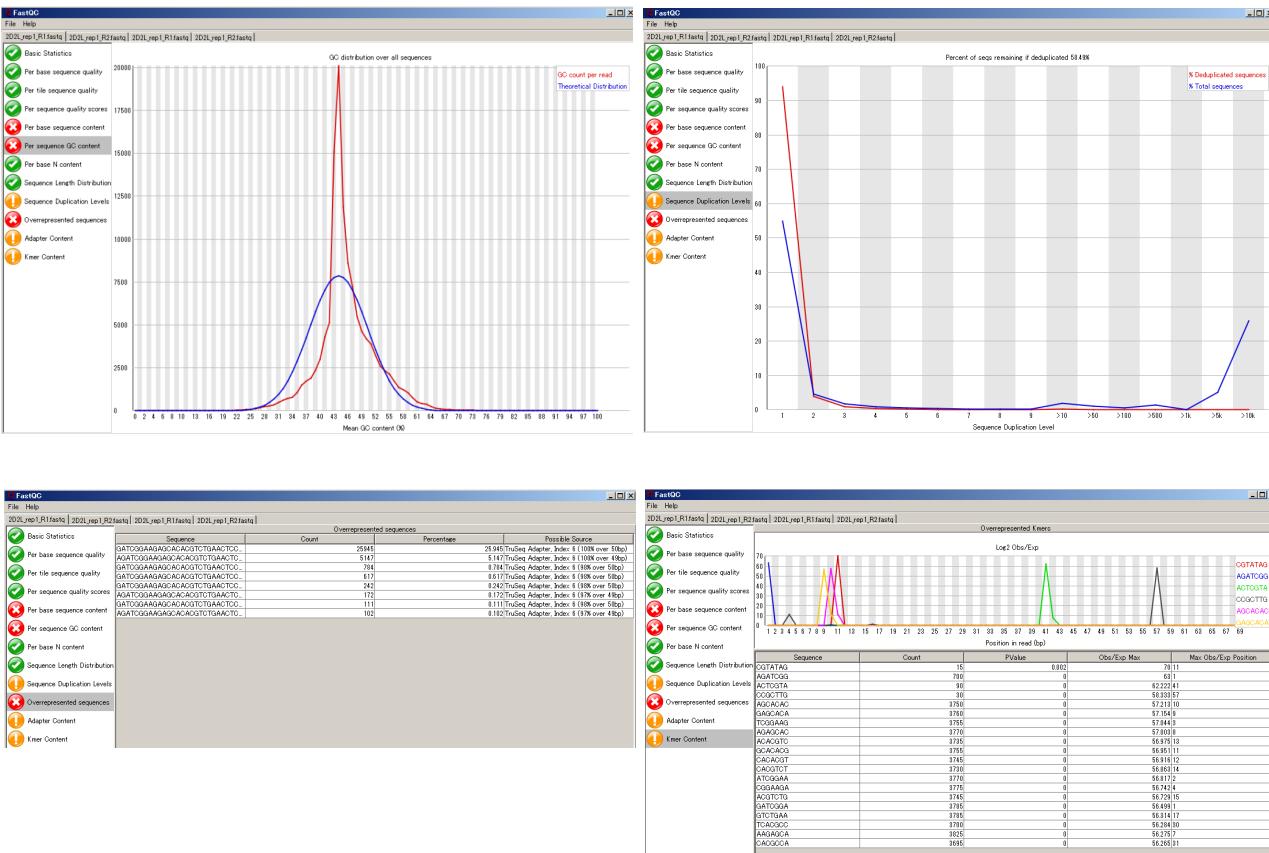
- Good Illumina Data
- Bad Illumina Data
- Adapter dimer contaminated run
- Small RNA with read-through adapter
- Reduced Representation BS-Seq
- PacBio
- 454

FASTQC使用法

GUI

java jdkを予めインストールしておく必要がある。





CUI

```
$ fastqc -h
```

FastQC - A high throughput sequence QC analysis tool

SYNOPSIS

```
fastqc seqfile1 seqfile2 .. seqfileN
```

```
fastqc [-o output dir] [--(no)extract] [-f fastq|bam|sam]
[-c contaminant file] seqfile1 .. seqfileN
```

gzファイルなら
--extract

実習 1 FASTQC

実習用ディレクトリ `~/data/5_ngs` に移動して `ls` で中を見る

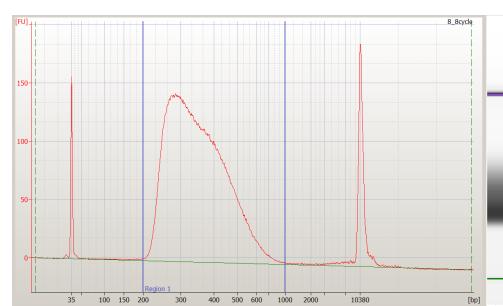
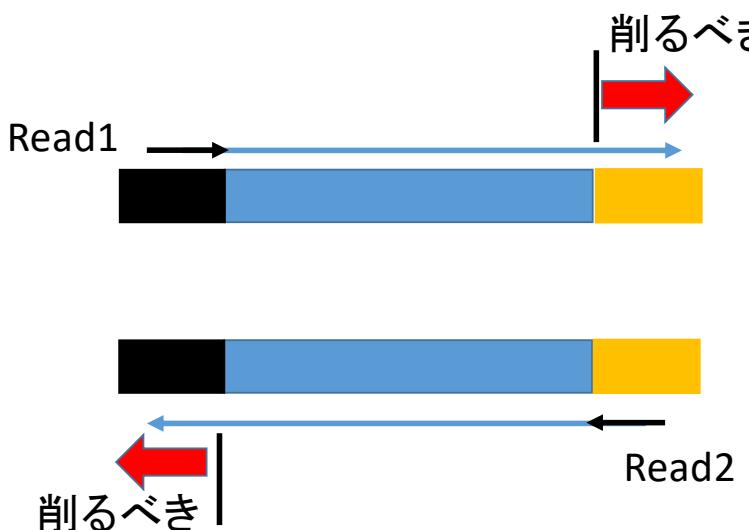
```
$ cd ~/data/5_ngs  
$ ls
```

- read結果 `2D2L_rep1_R1.fastq`

これをFASTQCに読み込ませてクオリティーを確認しよう

NGSデータのPre-processingの必要性

- ・余計な配列(アダプター配列)があるとリファレンス配列へのmappingに影響
- ・余計な配列(アダプター配列)がゲノム配列と誤認される



通常イルミナRNA-Seqライブラリーは200baseくらいの長さから存在するうち両端にアダプター63baseずつすなわち75base程度しかinsert配列がないライブラリーが存在する。

Pre-processing tools

以下の2ツールが有名

- Cutadapt
- Trimmomatic

生データを処理することで、アダプター配列を除去し、一定のクオリティーを確保したデータとなる

The screenshot shows the official Cutadapt documentation page. On the left, there's a sidebar with navigation links like 'Installation', 'User guide' (which is expanded), and 'Basic usage'. The main content area has a header 'User guide' and a sub-section 'Basic usage'. It contains text about trimming 3' adaptors and provides a command-line example: 'cutadapt -a AACCGGT -o output.fastq input.fastq'. Below this, there's more detailed information about how Cutadapt handles adapter sequences across all reads.

- adapter配列を除去
- 一定クオリティー以下の部位を除去
- 任意の配列部位を除去

Paired end readに対応
(ver. 1.8以降)
片方のreadが非常に
短くしか残らない場合、
pair read両方とも除去する。

<http://cutadapt.readthedocs.org/en/stable/guide.html>

MacOSXでのcutadaptのインストール

Cutadapt install手順

Xcode (Command line tool) のインストール
\$ xcode-select --install

homebrewインストール
\$ /usr/bin/ruby -e "\$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

python3のインストール
\$ brew install python3

バージョン確認
\$ python3 -V
Python 3.7.3

Cutadaptインストール
\$ pip3 install --user --upgrade cutadapt

パスを通す
以下にインストールされるので、
~/Library/Python/3.7/bin/cutadapt cutadapt
パスの通っている場所、例えば/usr/local/binにリンクを貼る
\$ ln ~/Library/Python/3.7/bin/cutadapt /usr/local/bin/cutadapt

インストールできたことを確認
\$ which cutadapt
/usr/local/bin/cutadapt

\$ cutadapt --version
2.3

現状最新はver. 2.3

<http://cutadapt.readthedocs.io/en/stable/installation.html>

Cutadapt

Removing adapters

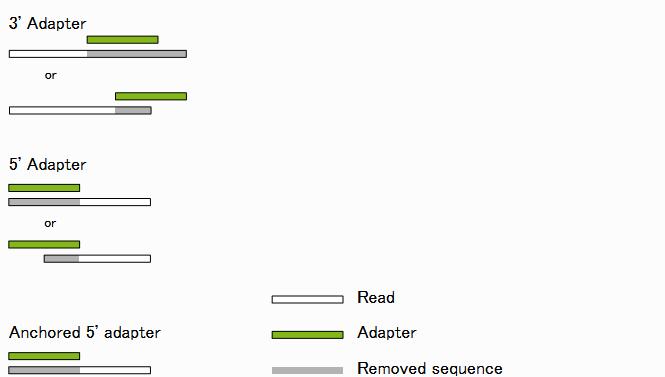
Cutadapt supports trimming of multiple types of adapters:

| Adapter type | Command-line option |
|--------------------------|---------------------|
| 3' adapter | -a ADAPTER |
| 5' adapter | -g ADAPTER |
| Anchored 3' adapter | -a ADAPTER\$ |
| Anchored 5' adapter | -g ^ADAPTER |
| 5' or 3' (both possible) | -b ADAPTER |

Here is an illustration of the allowed adapter locations relative to the read and depending on the adapter type:

Cutしたいアダプター配列の
位置関係など詳細に指定可能

fastqファイルはgz圧縮してあってもよい
fastaファイルも可



```
$ cutadapt -h
cutadapt version 2.3
```

Copyright (C) 2010-2019 Marcel Martin <marcel.martin@scilifelab.se>

cutadapt removes adapter sequences from high-throughput sequencing reads.

Usage:

```
cutadapt -a ADAPTER [options] [-o output.fastq] input.fastq
```

For paired-end reads:

```
cutadapt -a ADAPT1 -A ADAPT2 [options] -o out1.fastq -p out2.fastq in1.fastq in2.fastq
```

その他、有用なパラメータ

- j --cores
- q --quality-cutoff
- m --minimum-length
- O --overlap

- 使うCPU core数 defaultは1 0を指定しておくと自動検出
- クオリティー cutoffするQV値を指定
- 指定する長さ以下にcutされたものはreadそのものを削除
- 指定する配列とのオーバーラップを最小何baseとするか

crude_fastqフォルダーに生シーケンス配列
trim_fastqフォルダーにcutadaptにかけた配列
を用意してあります

Single readの場合

```
$ cutadapt ¥
-a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC ¥
-o hoge_read1.cut.fastq ¥
hoge_read1.fastq
```

Paired end readの場合

```
$ cutadapt ¥
-a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC ¥
-A AGATCGGAAGAGCGTCGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATC ¥
-o hoge_read1.cut.fastq ¥
-p hoge_read2.cut.fastq ¥
hoge_read1.fastq ¥
hoge_read2.fastq
```

実習 2 cutadapt

実習用ディレクトリ `~/data/5_ngs` に移動して `ls` で中を見る

```
$ cd ~/data/5_ngs
$ ls
```

- read結果 `2D2L_rep1_R1.fastq`

adapterがどの程度残っているか概算してみる

```
$ cat 2D2L_rep1_R1.fastq | grep 'AGATCGGAAGAGCAC' | wc
$ cat 2D2L_rep1_R1.fastq | wc
```

実際にcutadaptにかけて見よう

```
$ cutadapt ¥
-a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC ¥
-o 2D2L_rep1_R1.fastq.cut.fastq ¥
2D2L_rep1_R1.fastq
```

発展) 圧縮されたQV値

fastqファイルに記載されるQV値は、近年のシーケンサーではストレージ容量削減の為、圧縮効率の良い圧縮されたqv値が用いられている。

この場合、quality valueの階調数が減らされているので、それを考慮して-q値を指定すべき。

圧縮されたQV値で表記されたfastq

```
@E00441:177:HHNWCCXY:6:1101:13433:25464 1:N:0:GTGTATTA
AATGTGCGTTGTTGGGATAGGACATTGTCAGCTACGCGCCGGCTCTGTGAAGTAATTGGTTGAATGAATAAA
+
AAFFFJJJF-AFAF7A<<FJJF<AFJJJJJ7<FJJJJF-7F-7FJFFJ-A-FFAFJF-FFJJA-FJAFJJ<AAFA
```

- 12 A 32 数が限定されている
7 22 F 37 -q 31と28を比較しても
< 27 J 41 同じことになる



NGS基本ツール

- Seqkit
- Bowtie2
- SAMtools
- HT-Seq

SeqKitを使って見よう

fasta/fastqに関する様々な操作が可能なツール



RESEARCH ARTICLE

SeqKit: A Cross-Platform and Ultrafast Toolkit for FASTA/Q File Manipulation

Wei Shen¹, Shuai Le¹, Yan Li^{2*}, Fuquan Hu^{1*}

¹ Department of Microbiology, College of Basic Medical Sciences, Third Military Medical University, 30# Gaotanyan St., Shapingba District, Chongqing, China, ² Medical Research Center, Southwest hospital, Third Military Medical University, 29# Gaotanyan St., Shapingba District, Chongqing, China

SeqKitで出来ること、類似ツールとの比較

Features comparison

<https://bioinf.shenwei.me/seqkit/>

| Categories | Features | seqkit | fasta_utilities | fastx_toolkit | pyfaidx | seqmagick | seqtk |
|-----------------|-------------------------|--------|-----------------|---------------|---------|-----------|-------|
| Formats support | Multi-line FASTA | Yes | Yes | -- | Yes | Yes | Yes |
| | FASTQ | Yes | Yes | Yes | -- | Yes | Yes |
| | Multi-line FASTQ | Yes | Yes | -- | -- | Yes | Yes |
| | Validating sequences | Yes | -- | Yes | Yes | -- | -- |
| Functions | Supporting RNA | Yes | Yes | -- | -- | Yes | Yes |
| | Searching by motifs | Yes | Yes | -- | -- | Yes | -- |
| | Sampling | Yes | -- | -- | -- | Yes | Yes |
| | Extracting sub-sequence | Yes | Yes | -- | Yes | Yes | Yes |
| | Removing duplicates | Yes | -- | -- | -- | Partly | -- |
| | Splitting | Yes | Yes | -- | Partly | -- | -- |
| | Splitting by seq | Yes | -- | Yes | Yes | -- | -- |
| | Shuffling | Yes | -- | -- | -- | -- | -- |
| | Sorting | Yes | Yes | -- | -- | Yes | -- |
| | Locating motifs | Yes | -- | -- | -- | -- | -- |
| | Common sequences | Yes | -- | -- | -- | -- | -- |
| | Cleaning bases | Yes | Yes | Yes | Yes | -- | -- |
| | Transcription | Yes | Yes | Yes | Yes | Yes | Yes |
| | Translation | Yes | Yes | Yes | Yes | Yes | -- |
| | Filtering by size | Yes | Yes | -- | Yes | Yes | -- |
| | Renaming header | Yes | Yes | -- | -- | Yes | Yes |
| Other features | Cross-platform | Yes | Partly | Partly | Yes | Yes | Yes |
| | Reading STDIN | Yes | Yes | Yes | -- | Yes | Yes |
| | Reading gzipped file | Yes | Yes | -- | -- | Yes | Yes |
| | Writing gzip file | Yes | -- | -- | -- | Yes | -- |

類似のツールと比較して、
より多くのコマンドが利用でき、
高速である。

唯一、出来なかったtranslateが
v0.10より可能になった。

gz圧縮にも対応している。

最新はv0.10.1

```

$ seqkit
SeqKit -- a cross-platform and ultrafast toolkit for FASTA/Q file manipulation

Version: 0.10.1

Author: Wei Shen <shenwei356@gmail.com>

Documents : http://bioinf.shenwei.me/seqkit
Source code: https://github.com/shenwei356/seqkit
Please cite: https://doi.org/10.1371/journal.pone.0163962

Usage:
  seqkit [command]

Available Commands:
  common      find common sequences of multiple files by id/name/sequence
  concat      concatenate sequences with same ID from multiple files
  convert     convert FASTQ quality encoding between Sanger, Solexa and Illumina
  duplicate   duplicate sequences N times
  faidx      create FASTA index file and extract subsequence
  fq2fa       convert FASTQ to FASTA
  fx2tab     convert FASTA/Q to tabular format (with length/GC content/GC skew)
  genautocomplete generate shell autocompletion script
  grep        search sequences by ID/name/sequence/sequence motifs, mismatch allowed
  head        print first N FASTA/Q records
  help        Help about any command
  locate      locate subsequences/motifs, mismatch allowed
  mutate      edit sequence (point mutation, insertion, deletion)
  range       print FASTA/Q records in a range (start:end)
  rename      rename duplicated IDs
  replace     replace name/sequence by regular expression
  restart    reset start position for circular genome
  rmdupe    remove duplicated sequences by id/name/sequence
  sample      sample sequences by number or proportion
  seq         transform sequences (reverse, complement, extract ID...)
  shuffle     shuffle sequences
  sliding     sliding sequences, circular genome supported
  sort        sort sequences by id/name/sequence/length
  split       split sequences into files by id/seq region/size/parts (mainly for FASTA)
  split2      split sequences into files by size/parts (FASTA, PE/SE FASTQ)
  stats       simple statistics of FASTA/Q files
  subseq     get subsequences by region/gtf/bed, including flanking sequences
  tab2fx     convert tabular format to FASTA/Q format
  translate   translate DNA/RNA to protein sequence (supporting ambiguous bases)
  version     print version information and check for update

```

seqkitと打つとサブコマンドリストが出る
 サブコマンドリストまで打って-hで、
 その使い方や説明

Seqkitコマンド例

fastq.fastaファイルのstatatisticsを見る

```
$ seqkit stats hoge.fastq
```

fastqファイルからfastaファイルに変換する

```
$ seqkit fq2fa hoge.fastq > hoge.fasta
```

fastq.fastaファイルを複数のファイルに分割する

```
$ seqkit split -p 2 hoge.fastq
```

fastq.fastaファイルから一部のsamplingする

-nで指定する数は厳密なsampling数とは合致しないので注意

```
$ seqkit sample -n 100 hoge.fastq > hoge_100.fastq
```

fastq.fastaファイルのread順番をシャッフルする

```
$ seqkit shuffle hoge.fastq > shf_hoge.fastq
```

実習 3 seqkit

実習用ディレクトリ ~/data/5_ngs に移動して ls で中を見る

```
$ cd ~/data/5_ngs  
$ ls
```

- read結果 2D2L_rep1_R1.fastq

statsを確認する

```
$ seqkit stats 2D2L_rep1_R1.fastq
```

fastqからfastaへ変換する

```
$ seqkit fq2fa 2D2L_rep1_R1.fastq > 2D2L_rep1_R1.fasta
```

fastaファイルを3つに分割する

```
$ seqkit split -p 3 2D2L_rep1_R1.fasta
```

フォルダーが新規に作成され、分割されたファイルが出来ている

発展) read dataのランダムサンプリング

イルミナ社のシーケンサーではread結果はスキャンクラスターの場所ごとに並ぶことになる。

よって、headなどのコマンドで先頭のデータをサンプリングすると、隅のクラスターのシーケンス結果を得ることになる。

一般に隅のクラスターは試薬流路上、クオリティーの低いシーケンスリードが多く、これを全リードの代表として扱うのは問題である。

代表的なシーケンスreadを得るには、seqkitのsampleやshuffleの機能が使える。

Bowtieを使って見よう

- Burrows-Wheeler 変換に基づくインデックスを利用したショートリードのマッピングプログラム
- BowtieとBowtie2がある。後者はギャップを考慮した検索を行い、感度がより高い。また、検索の方針が単純化されて分かりやすくなるなど、多くの点で改良されている。
- シーケンスのリード長が長い(50bp以上)時はBowtie2の方が一般に検索効率がよく、精度も高い。リード長が短い(50bp未満)時はBowtieの方が検索効率または精度がいい場合もある。

The screenshot shows the official website for Bowtie. At the top, there's a dark blue header with the "BOW TIE" logo on the left, the word "Bowtie" in white, and "An ultrafast memory-efficient short read aligner" below it. To the right is the Johns Hopkins University logo. Below the header, a grey box contains text about Bowtie being an ultrafast aligner that indexes the genome with a Burrows-Wheeler index. It mentions a rate of over 25 million 35-bp reads per hour and a small memory footprint (about 2.2 GB for the human genome). To the right of this text is the "OSI certified" logo. Further down, another section for "Bowtie 2" is shown, featuring the same "BOW TIE" logo, the word "Bowtie 2" in white, and "Fast and sensitive read alignment" below it. The Johns Hopkins University logo is also present here. A grey box describes Bowtie 2 as an ultrafast tool for aligning sequencing reads to long reference sequences, noting its performance on mammalian genomes and its support for various alignment modes.

リファレンス配列へのマッピング

Bowtie, BWA, SOAP など

- 長大なリファレンス配列に、大量の短いリード配列を若干のミスマッチを許して照合する
- リファレンス配列に対して、あらかじめ全文検索インデックスを作成することにより高速に検索を行う
- paired-end read に対応。



インデックスとは



辞書における
インデックスタブ

- 索引、目次、見出し
- ファイルのどの辺りに何が書いてあるかの指標
- インデックスを作成すると別ファイルができるのは、分厚い本の「別冊目次」ができるイメージ
 - 欲しい情報を探すのにファイル(本)を先頭から総ナメして探さなくてもよい

NGSリファレンス配列のインデックスを作成 **bowtie2-build**

`bowtie2-build リファレンス配列ファイル インデックス名`

- 実行すると、インデックスとして、
 - ✓ インデックス名.n.bt2 (n=1-4)
 - ✓ インデックス名.rev.m.bt2 (m=1-2)の、計6つのファイルが作成される
- 配列ファイルはカンマで区切って複数を指定可能

実習 4 bowtie2-build

実習用ディレクトリ `~/data/5_ngs` に移動して `ls` で中を見る

```
$ cd ~/data/5_ngs  
$ ls
```

- リファレンス用ゲノムデータ(FASTA形式) `ecoli_genome.fa`

- bowtie2用インデックスの作成(インデックス名: `eco`)

```
$ bowtie2-build ecoli_genome.fa eco
```

- インデックスから元の配列データを再構築

```
$ bowtie2-inspect eco | less
```

NGSマッピングの実行 bowtie2

- マッピングの実行

- ✓ single-end read の場合

```
bowtie2 -x インデックス名 -U リードファイル -S 出力ファイル
```

- ✓ paired-end read の場合

```
bowtie2 -x インデックス名 -1 リードファイル1  
-2 リードファイル2 -S 出力ファイル
```

(実際は改行せずに1行で打つ)

- リードファイルはカンマ区切りで複数を指定可能

実習 5 bowtie2

- リード配列(FASTQ 形式, single-end read)
`ecoli.fastq`

リファレンス配列のインデックス名(実習4で作ったもの)
`eco`

- bowtie2の実行

```
$ bowtie2 -x eco -U ecoli.fastq -S eco_bowtie2.sam
```

マッピング結果: SAMフォーマットファイル

```
$ less -S eco_bowtie2.sam
```

```
@HD VN:1.3 SO:coordinate
@SQ SN:ref LN:45
r001 163 chr1 7 30 8M2I4M1D3M = 37 39 TTAGATAAAGGATACTG *
r002 0 chr1 9 30 3S6M1P1I4M * 0 0 AAAAGATAAGGATA *
r003 0 chr1 9 30 5H6M * 0 0 AGCTAA * NM:i:1
r004 0 chr1 16 30 6M14N5M * 0 0 ATAGCTTCAGC *
r003 16 chr1 29 30 6H5M * 0 0 TAGGC * NM:i:0
r001 83 chr1 37 30 M = 7 -39 CAGCGCCAT *
```

テンプレート名 フラグ マップ結果 アライメント(CIGAR) 対となるリードの位置情報 リードの配列 オプション

Bowtie2: その他のオプション

- **-h** ヘルプを表示する
 - **-a** 全てのアライメントを表示する
 - **-p 整数** 指定した数のCPUコアを使って実行する
 - **-f** リードがFASTA形式のファイルである
- 他、Bowtie2 マニュアル詳細
<http://bowtie-bio.sourceforge.net/bowtie2/manual.shtml>

Samtoolsを使って見よう

Samtools SAM<->BAM等の変換、データのソート、検索付け、特定readの抽出、統計情報収集などができる

Samtools is a suite of programs for interacting with high-throughput sequencing data. It consists of three separate repositories:

| | |
|-----------------|---|
| Samtools | Reading/writing/editing/indexing/viewing SAM/BAM/CRAM format |
| BCFtools | Reading/writing BCF2/VCF/gVCF files and calling/filtering/summarising SNP and short indel sequence variants |
| HTSlib | A C library for reading/writing high-throughput sequencing data |

Samtools and BCFtools both use HTSlib internally, but these source packages contain their own copies of htllib so they can be built independently.

Download
Source code releases can be downloaded from [GitHub](#) or [Sourceforge](#):
[Source release details](#)

Workflows
We have described some standard workflows using Samtools:

- WGS/WES Mapping to Variant Calls
- Using CRAM within Samtools

Documentation

- Manuals
- Specifications
- Duplicate Marking
- Zlib Benchmarks
- CRAM Benchmarks
- Publications

Support

- Mailing Lists

現行の最新はv1.9
バージョンによってオプションの与え方が変わっているコマンドに注意

NGSデータを扱うための最も基盤となるツール

Samtoolsの起動

\$ samtools

```
[kyamaguc@raid2016 ~]$ samtools
Program: samtools (Tools for alignments in the SAM format)
Version: 1.9 (using htslib 1.9)
Usage: samtools <command> [options]
Commands:
-- Indexing
    dict      create a sequence dictionary file
    faidx     index/extract FASTA
    fqidx     index/extract FASTQ
    index     index alignment
-- Editing
    calmd     recalculate MD/NM tags and '=' bases
    fixmate   fix mate information
    reheader  replace BAM header
    targetcut cut fosmid regions (for fosmid pool only)
    addreplacerg adds or replaces RG tags
    markdup   mark duplicates
-- File operations
    collate   shuffle and group alignments by name
    cat       concatenate BAMs
    merge    merge sorted alignments
    mpileup  multi-way pileup
    sort     sort alignment file
    split    splits a file by read group
    quickcheck quickly check if SAM/BAM/CRAM file appears intact
    fastq   converts a BAM to a FASTQ
    fasta   converts a BAM to a FASTA
-- Statistics
    bedcov   read depth per BED region
    depth    compute the depth
    flagstat simple stats
    idxstats BAM index stats
    phase    phase heterozygotes
    stats    generate stats (former bamcheck)
-- Viewing
    flags    explain BAM flags
    tview   text alignment viewer
    view    SAM<->BAM<->CRAM conversion
    depad   convert padded BAM to unpadded BAM
```

オプション/引数
なしで起動すると
Samtools の基本的な
使い方が表示される

実習しながら
進めます

Samtools の起動: コマンド簡易マニュアル

基本的な使い方: \$ samtools *command options*

\$ samtools view

```
Usage: samtools view [options] <in.bam>|<in.sam>|<in.cram> [region ...]

Options:
  -b          output BAM
  -C          output CRAM (requires -T)
  -f          use fast BAM compression (implies -b)
  -u          uncompressed BAM output (implies -b)
  -h          include header in SAM output
  -H          print SAM header only (no alignments)
  -c          print only the count of matching records
  -o FILE    output file name [stdout]
  -U FILE    output reads not selected by filters to FILE [null]
  -t FILE    FILE listing reference names and lengths (see long help) [null]
  -L FILE    only include reads overlapping this BED FILE [null]
  -r STR     only include reads in read group STR [null]
  -R FILE    only include reads with read group listed in FILE [null]
  -q INT     only include reads with mapping quality >= INT [0]
  -l STR     only include reads in library STR [null]
  -m INT     only include reads with number of CIGAR operations consuming
             query sequence >= INT [0]
  -f INT     only include reads with all of the FLAGS in INT present [0]
  -F INT     only include reads with none of the FLAGS in INT present [0]
  -G INT     only EXCLUDE reads with all of the FLAGS in INT present [0]
  -s FLOAT   subsample reads (given INT.FRAC option value, 0.FRAC is the
             fraction of templates/read pairs to keep; INT part sets seed)
:
```

- コマンドを付けてオプション無しで実行するとそのコマンドのマニュアルが表示される
- 詳細は <http://www.htslib.org/doc/samtools.html> を参照のこと

SAM/BAM変換

samtools view *options...*

- SAMファイルからBAMファイルの作成

```
$ samtools view -bS eco_bowtie2.sam -o eco_bowtie2.bam
```

- BAMをSAMに変換して less コマンドで表示

```
$ samtools view eco_bowtie2.bam | less
```

- BAMファイルを less で読もうとする…？

```
$ less eco_bowtie2.bam
```

- SAMファイルに比べてBAMファイルのサイズは？

```
$ ls -l eco_bowtie2.*
```

Samtoolsによるsort

samtools sort *options...*

```
$ samtools sort
Usage: samtools sort [options...] [in.bam]
Options:
  -l INT      Set compression level, from 0 (uncompressed) to 9 (best)
  -m INT      Set maximum memory per thread; suffix K/M/G recognized [768M]
  -n          Sort by read name
  -t TAG      Sort by value of TAG. Uses position as secondary index (or read name if -n is set)
  -o FILE     Write final output to FILE rather than standard output
  -T PREFIX   Write temporary files to PREFIX.nnnn.bam
  --input-fmt-option OPT[=VAL]
              Specify a single input file format option in the form
              of OPTION or OPTION=VALUE
  -O, --output-fmt FORMAT[,OPT[=VAL]]...
              Specify output format (SAM, BAM, CRAM)
  --output-fmt-option OPT[=VAL]
              Specify a single output file format option in the form
              of OPTION or OPTION=VALUE
  --reference FILE
              Reference sequence FASTA FILE [null]
  -@, --threads INT
              Number of additional threads to use [0]
```

マッピングデータをリファレンス配列上の位置順に並び替える

これをしないとindexを付けられない

BAM ファイルのソート

samtools sort options...

```
$ samtools sort eco_bowtie2.bam -o  
eco_bowtie2_sorted.bam
```

- samからの直接変換も可能 (v1.3以降)

```
$ samtools sort eco_bowtie2.sam -o  
eco_bowtie2_sorted.bam
```

- ソートされたBAMファイルをSAMに変換してlessで表示

```
$ samtools view eco_bowtie2_sorted.bam | less
```

- 元のSAMファイルの表示と比較してみよう

```
$ less eco_bowtie2.sam
```

BAMファイルにインデックスを付ける

samtools index options...

- 先にソートされている必要がある
- インデックスは .bai という拡張子付きの別ファイルで生成される。
- 「bamファイル名.bai」が作成されたのを ls コマンドで確認

```
$ samtools index eco_bowtie2_sorted.bam
```

```
$ ls eco_bowtie2_sorted*
```

ここから先はソート & インデックス付与したbamファイルを使う

ソート & インデックス付与したbamファイルを使って

指定した領域内のマッピング結果を表示

```
$ samtools view eco_bowtie2_sorted.bam chr:200-500
```



染色体名:開始位置 - 終了位置

マッピング統計情報収集 1

samtools idxstats options...

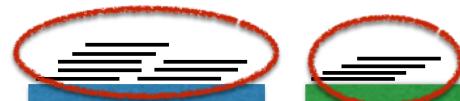
- 染色体毎にマップされたリード数を得る

```
$ samtools idxstats eco_bowtie2_sorted.bam
```

| 染色体名 | 染色体配列長 | マップされたリード数 | 片側のみマップされたリード数 |
|------|---------|------------|----------------|
| chr | 4639675 | 326754 | 0 |
| * | 0 | 0 | 3364 |



マップされなかったリード数
染色体名が '*' として表示される



マッピング統計情報収集 2

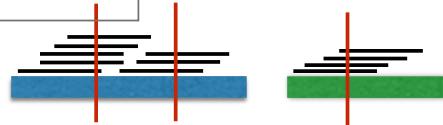
samtools depth options...

- 深度(マップされた回数)の統計情報を得る

```
$ samtools depth eco_bowtie2_sorted.bam
```

| 染色体名 | 位置 | 深度(マップされた回数) |
|------|----|--------------|
|------|----|--------------|

| | | |
|-----|---------|------|
| chr | 2753929 | 1533 |
| chr | 2753930 | 1470 |
| chr | 2753931 | 1446 |
| chr | 2753932 | 1101 |
| chr | 2753933 | 922 |
| chr | 2753934 | 918 |



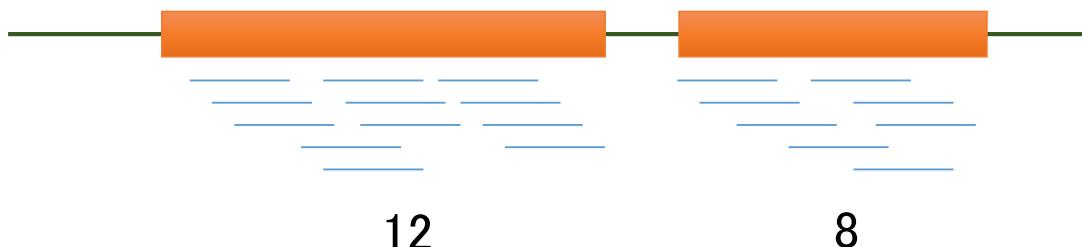
紹介したSamtools コマンドまとめ

Samtools

| | |
|----------|-------------------|
| view | リードを抽出, SAM/BAM変換 |
| sort | ソート |
| index | .bamのインデックス作成 |
| idxstats | 染色体毎のマッピング状況 |
| depth | 位置毎のマッピング深度 |

RNA-Seq解析に向けて

- ゲノム上にマッピングされたリードを遺伝子領域ごとに集めて数をカウント



通常、カウントした数を遺伝子の長さ、およびマップされたリード全体の数で割って標準化する

RPKM (Read Per Kilobase per Million mapped reads)

FPKM (Fragment Per Kilobase per Million mapped reads)

統計解析を含めた内容は実践編にて。

今回使ったツール・ファイルのまとめ

