

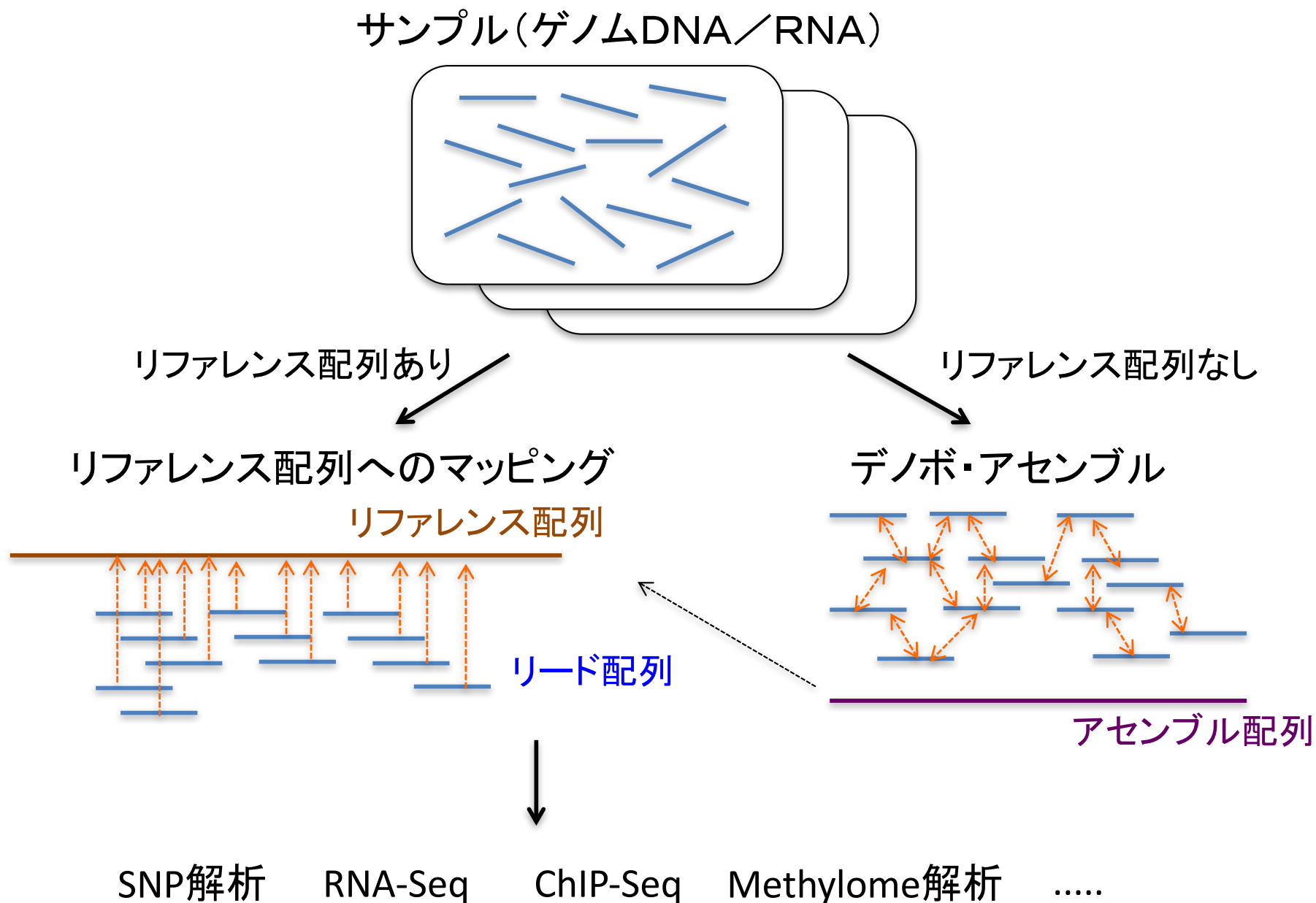
次世代シーケンサ用 データ解析コマンド

基礎生物学研究所

情報管理解析室

内山 郁夫

次世代シーケンサデータ処理の概要



次世代シーケンサデータ処理の概要

サンプル(ゲノムDNA/RNA)

FASTQ ファイル
(配列+クオリティ値)

FASTAファイル(配列)

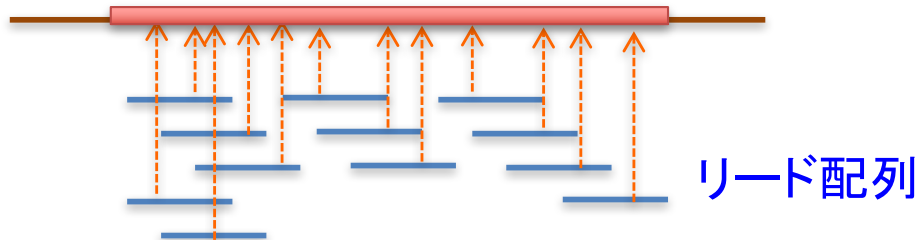
```
>chr
AGCTTTTCATTCTGACTGCAACGGGCAATATGTCT
CTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC
TTCTGAACCTGGTTACCTGCCGTGAGTAAATTA
TTTATTGACTTAGGTCACTAAATACTTTAACCAA
TATAGGCATAGCGCACAGACAGATAAAAAATTACAG
AGTACACAACATCCATGAAACGCATTAGCACCACC
ATTACCACCACCATTACCATTACCACAGGTAACGG
```

```
@SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:3
ATCCGGCTGGCGCACCGACCTATGTTCCGGGCGAATACAAGCTGGG
+SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:3
@@@AD>DDFF7DC?FFEBF@DFII<DF@AAA6AEFBDDBDCA?>A?B
@SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:3
CACCGTGTAGTACCAGCATCCTGCGTACAATCAGCAATCCCAGTCC
+SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:3
CCCFDFDFHDFHIIIEGIIHJJJGFHGGHGGHGGIJDGIJHHG
@SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:3
CAGGACATCGCCTTTGATCGGTTCCGACTTCGACCAACCTGCATT
+SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:3
CCCFDFDFAFHFHIIJGHIJJJJJHEHIIJGHIFEHIIA@FIFHG
```

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

リファレンス配列



GFF (GTF)ファイル
(遺伝子アノテーション)

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

SAM ファイル(マッピング結果)

```
@HD      VN:1.0  SO:unsorted
@SQ      SN:chr  LN:4639675
@PG      ID:bowtie2  PN:bowtie2  VN:2.2.4  CL:"/bio/bin/bowtie2-align
SRR1515276.40 0 chr 4423609 42 51M * 0 0 GGAATTCCTCACTGCCA
SRR1515276.158 16 chr 501700 42 51M * 0 0 ACGCACCGAGTGCAAAG
SRR1515276.212 4 * 0 0 * 0 0 GGCCGCTTTTCAGCGTGT
SRR1515276.319 0 chr 2922768 42 51M * 0 0 GCTTAAGTTGATTAAGG
SRR1515276.367 16 chr 2753873 42 51M * 0 0 GCGTGTCCGTCCGCAGC
```

配列データファイル: FASTA format

```
>NM_174292.2 Bos taurus crystallin gamma S (CRYGS) mRNA
TGCACCAAACATGTCTAAAGCTGGAACCAAAATTACTTTCTTTGAAGACAAAAC
TTTCAAGGCCGCCACTATGACAGCGATTGCGACTGTGCAGATTTCCACATGTACC
TGAGCCGCTGCAACTCCATCAGAGTGGAAGGAGGCACCTGGGCTGTGTATGAAAG
GCCCAATTTTGCTGG
>NM_174294.1 Bos taurus casein kappa (CSN3) mRNA
TTCACCTACAGTGGAAGGCCAACTGAACCTACTGCCAAGCAAGAGCTGACGGTC
ACAAGGAAAGGTGCAATGATGAAGAGTTTTTTCCTAGTTGTGACTATCCTGGCAT
TAACCCTGCCATTTTTTGGGTGCCCAGGAGCAAAACCAAGAACAACCAATACGCTG
TGAGAAAGATGAAAG
```

← >配列ID(説明)

← 塩基配列

- 配列ファイルの標準フォーマット
- >で始まる行がタイトル行、その後に配列が続く
- タイトル行の最初の単語が配列ID、以降は説明(省略可)
- タイトル行の長さに制限はないが、途中で改行は入らない
- 配列は途中で改行が入ってもよい

配列データファイル: FASTQ format

```
@ERR004063.1 IL33_2678:6:1:0:902/2
CTGTAAATATGTACAGGATATTTCTGACCATTTCTTC
+
CCDDCC8@DD8DDCC5?,D7??&=-4&9*&&+-.' .
@ERR004063.2 IL33_2678:6:1:0:1059/2
AAATGGAGACTTATTCCTTGTCTTTGGGTAATCAATC
+
@@@=@;>CC;<;DD>>7AA>88>DE>AC96@;&<C7>
@ERR004063.3 IL33_2678:6:1:0:1320/2
AGCATTTGGAGTGGCTTTTTTTTTTTGTTTTTTTTTAAA
+
07ECEEE=CDCA<AC108D@0(*4(, :0;6*0*(4((
```

← @配列ID (説明)
← 塩基配列
← + (配列ID 説明)
← クオリティ値

- 配列とクオリティ値をひとつにまとめたもの
- 各データの先頭は「@」、塩基配列とクオリティ値配列の間に「+」
- クオリティ値は「アスキーコード」に従って文字列で表示される
→ 数字を使う場合と比べてサイズが圧縮され、高速な処理が可能
- 塩基配列、クオリティ値配列ともに1行で書くのが基本
- @や+で始まる行がタイトル行やクオリティデータ開始行であるとは限らない(@や+はクオリティ値の中にも現れるので)

クオリティ値

- PHRED Quality: エラー率 p_e に対して、以下で定義される

$$Q_{PHRED} = -10 \log_{10}(p_e)$$

例) $Q_{PHRED}=30$ の場合、エラー率は 10^{-3}

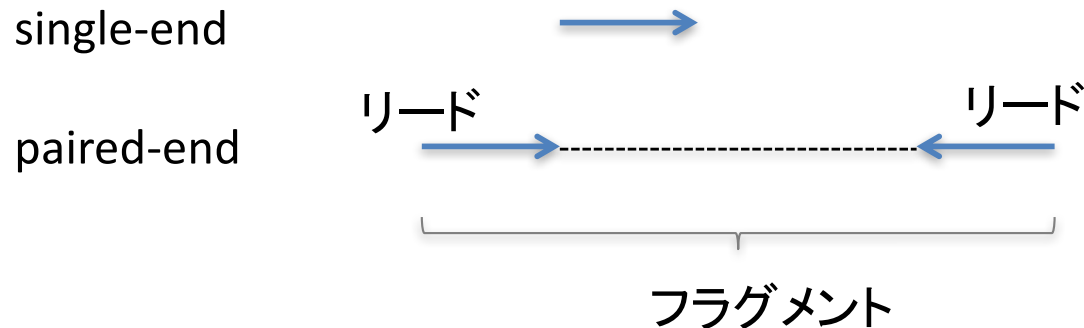
- $Q_{PHRED}+33$ の値が、以下のアスキーコード表に従って文字列として表示される

	30	40	50	60	70	80	90	100	110	120
0:		(2	<	F	P	Z	d	n	x
1:)	3	=	G	Q	[e	o	y
2:	(SP)	*	4	>	H	R	¥	f	p	z
3:	!	+	5	?	I	S]	g	q	{
4:	"	,	6	@	J	T	^	h	r	
5:	#	-	7	A	K	U	_	i	s	}
6:	\$.	8	B	L	V	'	j	t	~
7:	%	/	9	C	M	W	a	k	u	(DEL)
8:	&	0	:	D	N	X	b	l	v	
9:	'	1	;	E	O	Y	c	m	w	

リファレンス配列へのマッピング

Bowtie, BWA, SOAP などのコマンド

- 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。insert sizeに制約をつけられる



Bowtie コマンド

- BowtieとBowtie2がある。後者はギャップを考慮した検索を行うので、感度がより高い

- インデックスの作成

```
$ bowtie2-build 配列ファイル インデックス名
```

- 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1 -2 リード2 -s 出力ファイル
```


実習

データ: 大腸菌RNA-Seqデータ(GEO:SRP044366)
の一部を抜粋したもの

ディレクトリ: `~/data/3_ngs`

リードファイル: `test_fastq/ecoli.[1-12].fastq`

大腸菌ゲノム配列: `ecoli_genome.fa`

大腸菌遺伝子テーブル: `ecoli.gtf`

- 作業ディレクトリに移動

```
$ cd ~/data/3_ngs
```

Bowtie実習

- リファレンス配列 (ecoli_genome.fa) に対してインデックスを作成する

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

- リードファイル test_fastq/ecoli.1.fastq をリファレンス配列上にマッピングする

```
$ bowtie2 -x ecoli_genome  
          -U test_fastq/ecoli.1.fastq  
          -S ecoli.sam
```

(改行せずに1行で打つ)

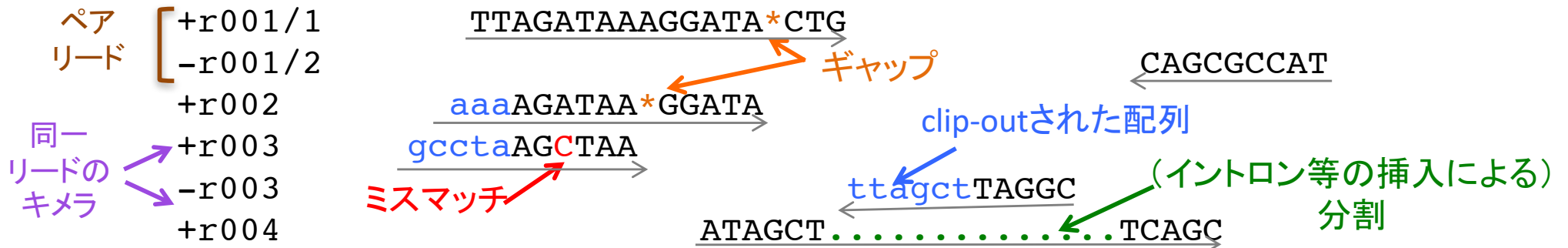
マッピング結果ファイル: SAM format

- リファレンス配列に多数の配列をマップした結果を表す形式
- 最初の@付きの行はヘッダ行、以降はタブ区切りで記述されたマッピング結果のデータ

```
@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

リファレンス配列 chr1 12345678901234 5678901234567890123456789012345
AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT



マッピング結果ファイル: SAM format

ヘッダ行

@HD VN: バージョン番号 SO:並び順

@SQ SN: リファレンス配列名 LN:リファレンス配列長

@RG リードグループの情報、@PG プログラムの情報 等

@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

SAM形式各カラムの情報

1. QNAME: クエリーテンプレート配列名

2. FLAG: 各種情報を保持したフラグ

3. RNAME: リファレンス配列名

4. POS: リードがマッピングされた左端位置

5. MAPQ: マッピング クオリティ $-10\log_{10} p_e$

6. CIGAR: CIGAR 文字列 (アライメント)

7. RNEXT: 隣接リードが載ったリファレンス配列名 (=は同一配列、*は情報なし)

8. PNEXT: 隣接リードのマッピング位置

9. TLEN: テンプレート (フラグメント) の長さ

10. SEQ: リードの配列

11. QUAL: phred quality+33 のASCII表現

12. OPT: tag:type:value で表す任意の情報

BAM format

- SAM形式のテキストファイルをバイナリファイルに変換し、かつデータ圧縮をかけてコンパクトかつ効率的に処理できるようにしたもの。
- samtoolsを使って変換する

✧ SAM→BAM変換

```
samtools view -b file.sam -o file.bam
```

- -b 出力形式がBAMファイル
- -o 出力ファイル名

✧ BAM→SAM変換

```
samtools view -h file.bam > file.sam
```

- -h 出力にヘッダを含める

samtools

- SAM/BAM形式のマッピング結果ファイルを扱うためのコマンド群
- `samtools <command> [options]`

2番目の引数の<command>によって機能を切り替える

- `view` SAM \leftrightarrow BAM 変換して表示
- `sort` 並べかえ (デフォルトはマップされた位置で)
- `index` ソートされたBAMファイルをインデックスづけ
- `idxstats` 各リファレンス配列にマップされたリード数を表示
- `depth` リファレンスの各位置にマップされたリード数を表示
- `mpileup` リファレンスの各位置にマップされた塩基を表示

samtools 実習1

- SAMファイルからBAMファイルの作成

```
$ samtools view -b ecoli.sam -o ecoli.bam
```

- BAMをSAMに変換して表示

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

samtools 実習2

BAMファイルは、ソートしてインデックスづけを行うことによって、より便利に使えるようになる。

- BAMファイルをソート

```
$ samtools sort ecoli.bam -o ecoli_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 ecoli_sorted.bam |less
```

- ソートされたBAMファイルに対してインデックスを作成

```
$ samtools index ecoli_sorted.bam
```


samtools 実習3

以下は、インデックスづけされたBAMファイルを使う

- 指定した領域内にマッピングされたリードのみを表示

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

染色体名:開始位置-終了位置

- 各染色体にマッピングされたリード数を表示

```
$ samtools idxstats ecol_i_sorted.bam
```

マッピングされなかったリードは、染色体名が '*' として表示される

- mpileupコマンドで、リファレンスの位置ごとにマッピングされた塩基を表示

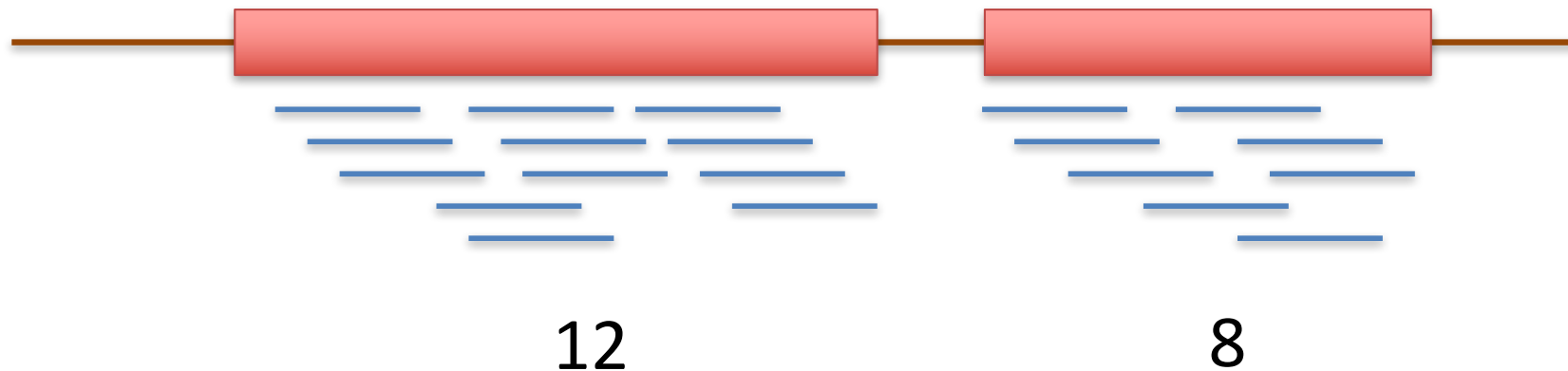
```
$ samtools mpileup -f ecol_i_genome.fa ecol_i_sorted.bam | less
```

(参考)bcftoolsと組み合わせて多型部位の抽出を行う

```
$ samtools mpileup -u -f ecol_i_genome.fa ecol_i_sorted.bam | bcftools call -mv
```

RNA-Seq 解析

- 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) または
FPKM (Fragment Per Kilobase per Million mapped reads)

遺伝子アノテーションファイル: GFF format

- ゲノム上の特徴配列(Feature segment)をタブ区切りテキスト形式で表現する標準的なフォーマット
- 最後のカラムに任意の情報を格納できるため、様々な特徴配列の情報を記述できるが、とくに遺伝子構造を記述するために特化した形式を **GTF format** という

1	2	3	4	5	6	7	8	9
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";

1. Seqname 配列名
2. Source 予測プログラム、データベース名など
3. Feature 特徴セグメントの種類
4. Start 開始位置
5. End 終了位置

6. Score スコア、省略可 (.)
7. Strand スtrand(+/-)、省略可 (.)
8. Frame 読み枠(0/1/2)、省略可 (.)
9. Attribute (Optional)
セミコロンで区切られたタグ=値の対

マッピング結果を領域ごとに 集計するコマンド: htseq-count

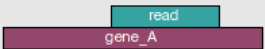
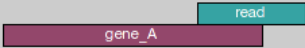


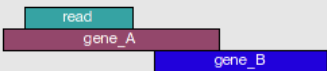
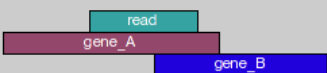

- htseq-count マッピングファイル(SAM) 遺伝子ファイル(GFF)

オプション

- m 照合モード
- t 参照するフィーチャータイプ
- i 遺伝子IDとして使う属性値

遺伝子アノテーション(GFF)ファイルの中で、どのFeatureの行を使うかはオプション -t で、遺伝子IDとして何を使うかについては-i で指定する。標準的なGTF形式のファイルであれば、デフォルトのまま(Featureがexonの行を使い、遺伝子IDはgene_idを使う)で動作するようになっている。

照合モード

	union	intersection_strict	intersection_nonempty
	gene_A	gene_A	gene_A
	gene_A	no_feature	gene_A
	gene_A	no_feature	gene_A
	gene_A	gene_A	gene_A
	gene_A	gene_A	gene_A
	ambiguous	gene_A	gene_A
	ambiguous	ambiguous	ambiguous

htseq-countを用いた実習

- アノテーションテーブルecoli.gtfを使って、各遺伝子にマッピングされたリード数をカウント

```
$ htseq-count ecoli.sam ecoli.gtf > ecoli.htseq
```

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";

デフォルトでは遺伝子の領域として、Featureがexonの行を使い、遺伝子のIDとしてはgene_id の値を使う

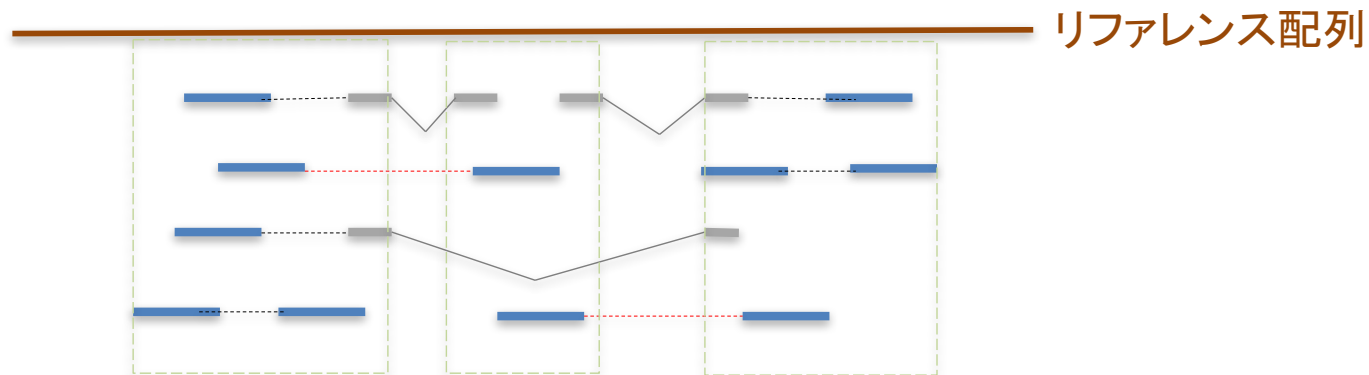
出力結果

b0001	11
b0002	117
b0003	33
b0004	44
b0005	3
b0006	14
b0007	4

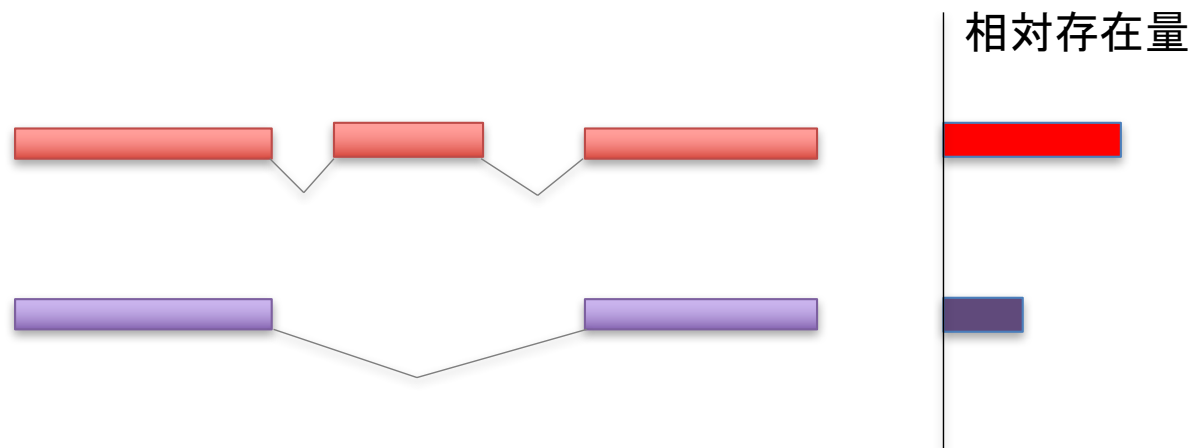
スプライスバリエントを持つ遺伝子への対応

Tophat/Cufflinks

TopHat: 新規スプライス部位を考慮したリードのマッピング



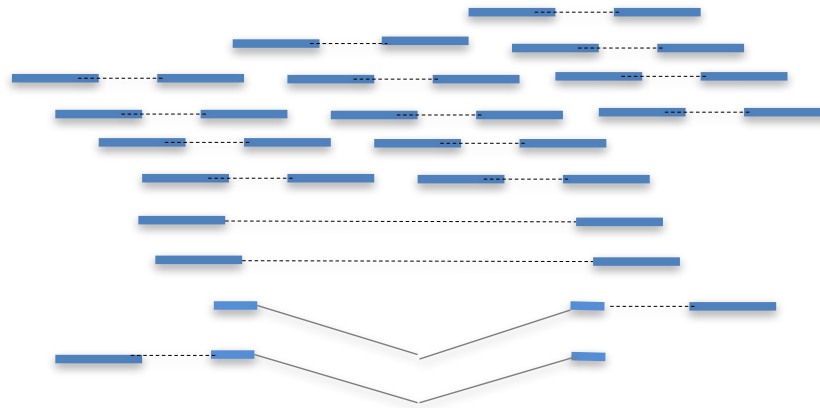
Cufflinks: 転写配列をアセンブルして各バリエントの存在量を推定



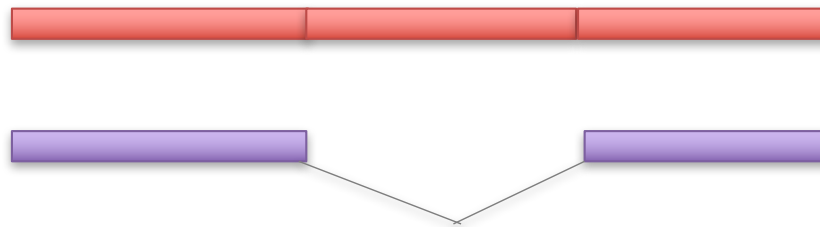
デノボ・アセンブルによるRNA-Seq解析

Trinity

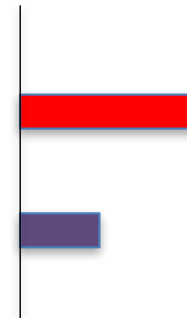
Trinity: デノボ・アセンブルによる転写配列の構築



アセンブル配列



RSEM: 各バリアントの
存在量の推定



生物情報解析システムで 利用可能なソフトウェア

次世代シーケンサ解析

- マッピング
 - Bowtie, BWA, SOAP
- RNA-Seq解析
 - TopHat, Cufflinks, Trinity
- アセンブラ
 - Velvet, ABySS, AllPaths-LG
- ユーティリティ
 - samtools, bamtools, BEDtools, cutadapt
SRA toolkit

その他のツール

- ホモロジー検索
 - BLAST, FASTA
- 遺伝子予測
 - GeneMark, GenScan, Augustus
- ゲノムアライメント
 - lastz, MUMmer, BLAT
- マルチプルアライメント
 - ClustalW, Muscle, MAFFT
- 系統樹解析
 - Phylip, PhyML, MrBayes
- モチーフ解析
 - InterProScan, HMMER, MEME
- データベース検索
 - DBGET
- 統合配列解析
 - EMBOSS

Module コマンド

BIAS上で利用可能なプログラムの一部は、moduleによってバージョンごとに管理されている。

- module avail
 - 利用可能なモジュールのリストを表示
- module add モジュール名
 - 特定のプログラム(の特定バージョン)のモジュールをロード
- module rm モジュール名
 - モジュールをアンロード
- module list
 - ロードされたモジュールのリスト表示

実行例) EMBOSSをロードして、infoseqコマンドを実行

```
$ module add EMBOSS  
$ infoseq ecoli_genome.fa
```

生物情報解析システムで 利用可能なデータベース

/bio/db の下に様々な形式のデータベースファイルが置いてある

次世代シーケンサ解析向け
ゲノムデータベース

公的配列データベース

/bio/db/fastadb 名 FASTA形式

/bio/db/ideas/db 名 オリジナルの形式

iGenomes

/bio/db/igenomes/生物種名

Bowtie, BWAなどのインデックスづけが
されたゲノムデータベースが利用可能

例) ヒトゲノムbuild37.2のbowtie2イン
デックスづけされたもの

/bio/db/igenomes/Homo_sapiens/NCBI/
build37.2/Sequence/Bowtie2Index/

データベース名	アクセス名	説明	フォーマット	更新型
GenBank	genbank	核酸塩基配列	DBGET	定期
GenBank-upd	genbank-upd	核酸塩基配列	DBGET	日々
EMBL	embl	核酸塩基配列	DBGET	定期
EMBL-upd	embl-upd	核酸塩基配列	DBGET	日々
RefSeq	refseq	refnuc + refpep	DBGET	日々
RefSeq Nuc.	refnuc	核酸塩基配列	DBGET	日々
RefSeq Pep.	refpep	タンパク質アミノ酸配列	DBGET	日々
RefSeq Protein	refseq_protein	タンパク質アミノ酸配列	FASTA, BLAST	日々
RefSeq Genomic	refseq_genomic	ゲノム配列	FASTA, BLAST	日々
RefSeq RNA	refseq_rna	RNA塩基配列	FASTA, BLAST	日々
EST_human EST_mouse EST_others	est_human est_mouse est_others	核酸塩基配列	FASTA, BLAST	定期
NCBI nr-nt	nt	非冗長核酸塩基配列	FASTA, BLAST	定期
gss	gss	核酸塩基配列	FASTA, BLAST	定期
HTGS	htgs	核酸塩基配列	FASTA, BLAST	定期
dbsts	dbsts	核酸塩基配列	FASTA, BLAST	定期
patnt	patnt	核酸塩基配列	FASTA, BLAST	定期
env_nt	env_nt	核酸塩基配列	FASTA, BLAST	定期
pdnt	pdnt	核酸塩基配列	FASTA, BLAST	定期
NCBI nr-aa	nr	非冗長アミノ酸配列	FASTA, BLAST	定期
UniProt	uniprot	TrEMBL + Swissprot	DBGET, FASTA, BLAST	日々
TrEMBL	trembl	タンパク質アミノ酸配列	DBGET, FASTA, BLAST	日々
Swissprot	swissprot	タンパク質アミノ酸配列	DBGET, FASTA, BLAST	日々
pataa	pataa	タンパク質アミノ酸配列	FASTA, BLAST	定期
env_nr	env_nr	タンパク質アミノ酸配列	FASTA, BLAST	定期
pdbaa	pdbaa	タンパク質アミノ酸配列	FASTA, BLAST	定期
PDB	pdb	タンパク質立体構造	DBGET	定期