

# RNA-seq 入門

NGS の基礎から de novo 解析まで  
準備編・実践編

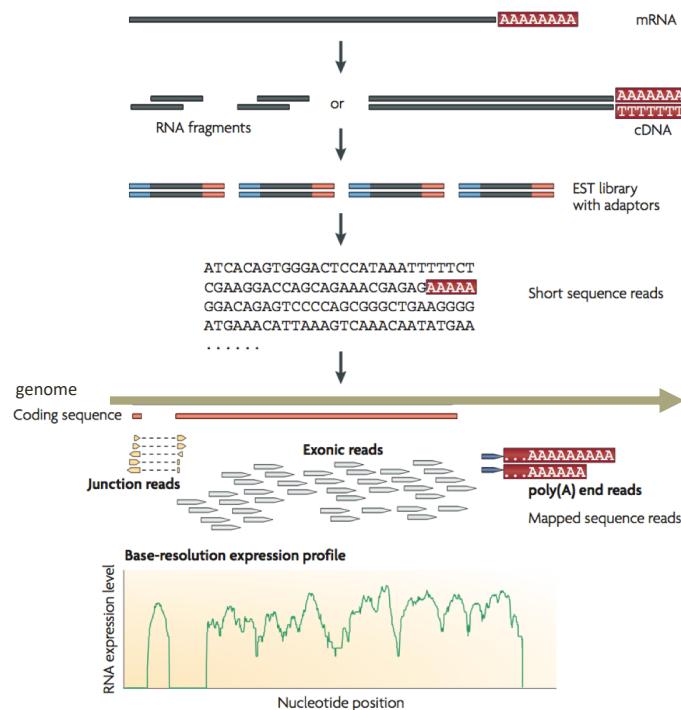
## Overview

March 8-9, 2018 @ NIBB (Okazaki)

重信秀治 / Shuji Shigenobu

- サポート Wiki  
<https://github.com/nibb-gitc/gitc2018mar-rnaseq/wiki>

## RNA-seq



(Wang 2009 with modifications)

# Two major goals

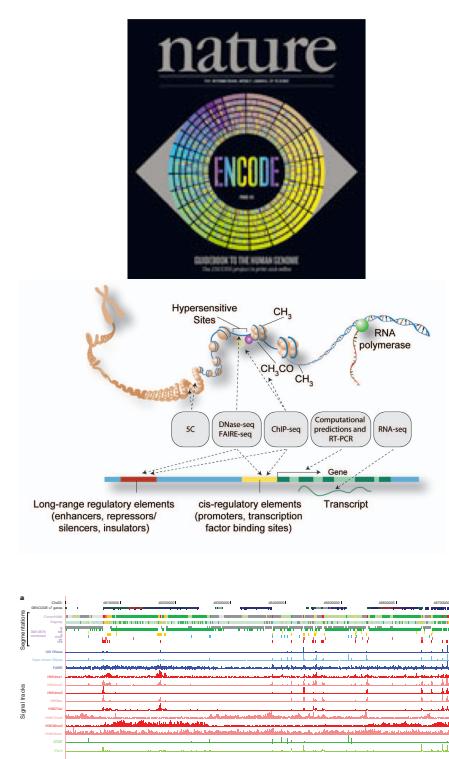
- Gene cataloguing
- Gene expression analysis

RNA-seq is unraveling complexities of eukaryotic transcriptomes in model and non-model organisms

- Gene expression analysis
- Novel gene discovery (model org.)
  - Coding and non-coding genes
- Gene cataloguing (non-model org.)
- Anti-sense transcripts
- RNA editing
- Novel splicing variants & fusion genes
- Allele-specific expression

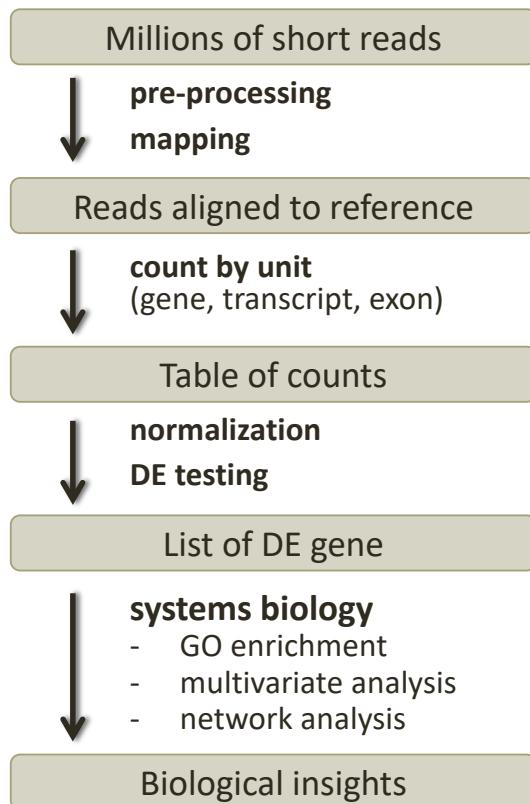
Beyond transcriptome

- DB for proteome analysis
- SNP finding
- *and more ...*



# RNA-seq analysis pipeline for DE

Differential Expression analysis



## 解析ツールの現状: RNA-seq

- 全てのプロセスをこなせる万能ツールはない。
- それぞれのステップに特化したツール群が次々に登場している。

### 基本戦略

- 各ステップに最適なツールをチョイス、組み合わせた、解析パイプラインの構築。

### Pipeline

- 本コースで学ぶオススメの2つのパイプライン
  - Transcriptome-based: Bowtie/eXpress/edgeR
  - Genome-based: HISAT2/StringTie/edgeR

# NGS 基本フォーマットとツール 復習と補足

基礎生物学研究所  
ゲノムインフォマティクストレーニングコース  
内山 郁夫 ([uchiyama@nibb.ac.jp](mailto:uchiyama@nibb.ac.jp))

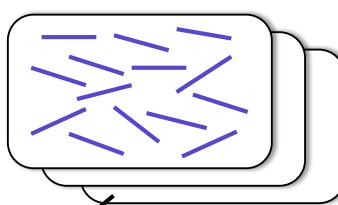
## ショートリードのマッピング

ゲノム配列  
(リファレンス reference 配列)

形式(配列)  

```
>chr
AGCTTTCAATTCTGACTGCAACGGGCAATATGCTT
CTGTGTGGATTAAAAAAAGACTGTCIGATAGCAGC
TTCTGAACCTGGTTACCTGCCCTGAGTAATTTAAA
TTTATTGACTTAGGTCACTAAATACTTTAACCAA
TATAGGCATAGCCACAGACAGATAAAATTACAG
AGTACACAACATCCATGAAAGCGATTAGCACCAACC
ATTACCACCAACCATCACCATTACCAACAGGTAACGG
```

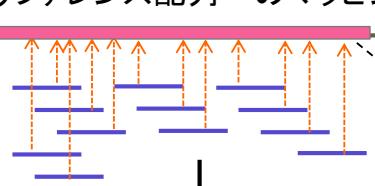
サンプル(ゲノムDNA／RNA)  
(リード read 配列)



形式  
(配列 + クオリティ値)  

```
@SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:
ATCCGGCTGGCCACCGACCTATGTTCCGGCGAATACAAGCTGG
+SRR1515276.1 HWI-ST808:151:D2D13ACXX:2:1207:
@@@AD>DFFF7DC?FEBFB#FII<DF@AA6AEFBDBCA?>A?
@SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:
CACCGCTAGTACCAAGCATCCTGCCGTACAAATCAGCAATCCAGTC
+SRR1515276.2 HWI-ST808:151:D2D13ACXX:2:1207:
CCCCFPDFPHDPFHIIIEGHJJJCJPFHGHHGGGILJDGLJHHE
@SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:
CAGGACATCGCCTTTGATCGGTTCAAGACTTCGGACCAACCTGCAT
+SRR1515276.3 HWI-ST808:151:D2D13ACXX:2:1207:
CCCCFFDFAFHFHIJGHIIJJIJUHEHIIJGHIFEHIIIA@FIFEE
```

リファレンス配列へのマッピング



クオリティチェック  
アダプター除去

形式(遺伝子アノテーション)

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

形式(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-alig
SRR1515276.40 0 chr 4423609 42 51M * 0 0 GGAATTCCCACTGGCCA
SRR1515276.158 16 chr 501700 42 51M * 0 0 ACGCACCGACTGCAAAG
SRR1515276.212 4 * 0 0 * * 0 0 GGCCGCTTTCAGCGTGT
SRR1515276.319 0 chr 2922768 42 51M * 0 0 GCTTAAGTGATTAAGG
SRR1515276.367 16 chr 2753873 42 51M * 0 0 GCGTGTCCGTCCGCAGC
SRR1515276.411 0 chr 3440721 42 51M * 0 0 ACGGCATAATTTCCTTGA
```

# 復習: **cutadapt**による アダプターの除去

実習用ディレクトリ ~/data/IU

入力

- リード配列(FASTQ 形式; paired-end)  
etec\_1.fq  
etec\_2.fq
- アダプター配列 (それを3'端から除去)

Adapter1: AGATCGGAAGAGCGGGTT

Adapter2: AGATCGGAAGAGCGTCG

## ◆ アダプター除去の実行

除去後のデータ(FASTQ形式)は etec\_1.cut.fq, etec\_2.cut.fqとする

\$ cutadapt

# 復習: **cutadapt**による アダプターの除去

実習用ディレクトリ ~/data/IU

入力

- リード配列(FASTQ 形式; paired-end)  
etec\_1.fq  
etec\_2.fq
- アダプター配列 (それを3'端から除去)

Adapter1: AGATCGGAAGAGCGGGTT

Adapter2: AGATCGGAAGAGCGTCG

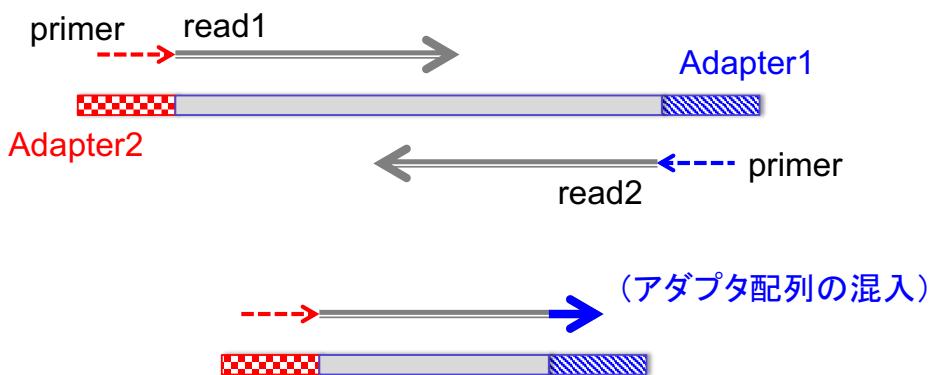
## ◆ アダプター配列除去の実行

除去後のデータ(FASTQ形式)は etec\_1.cut.fq, etec\_2.cut.fqとする。

\$ cutadapt

	AGATCGGAAGAGCGGGTT		AGATCGGAAGAGCGTCG
etec_1.cut.fq		etec_2.cut.fq	
etec_1.fq		etec_2.fq	

# Illuminaにおけるアダプター配列



Adapter1: AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC

Adapter2: AGATCGGAAGAGCGTCGTAGGGAAAGAGTGTA

cutadapt -a (-A) オプションでは、指定した配列とマッチした箇所以降の3'側を切り捨てるので、アダプタ配列は全長を指定しなくてもよい。

## cutadapt その他のオプション

- **-q [5' cutoff,] 3' cutoff** (例: -q 20)
  - ・クオリティ値が指定したカットオフより低い塩基を3'端から除く(カンマ区切りでカットオフを2つ指定した場合は5'端からも除く)
- **-m min\_length** (例: -m 30)
  - ・アダプター除去後の配列長が指定した長さ以下になつたら配列全体を捨てる。
  - ・ペアエンドの場合、ペアのどちらかが捨てられる場合は両方を捨てる。  
→2つのファイルで対応する配列の出現順が揃うようにする。
- **-O overlap\_length** (例: -O 5)
  - ・アダプターとリードとの間で、マッチしたと見なす最低のオーバーラップ長を指定。デフォルトは3。



## 復習: **bowtie2** 用インデックスの作成

実習用ディレクトリ `~/data/IU`

入力

- ゲノムデータ (FASTA形式)

`eco_o139.fa` 腸管毒素原性大腸菌(ETEC) O139:H28のゲノム配列

- ◆ **bowtie2**用インデックスの作成(インデックス名は `etec`)

\$ `bowtie2-build`

## 復習: **bowtie2**の実行 (paired-end)

実習用ディレクトリ `~/data/IU`

入力

- リード配列(FASTQ 形式; paired-end; アダプターを除去したもの)

`etec_1.cut.fq`

`etec_2.cut.fq`

- リファレンス配列のインデックス名

`etec` (先ほど作ったもの)

- ◆ **bowtie2**によるマッピングの実行 (出力:`etec_bowtie2.sam`)

\$ `bowtie2`  `etec`  `etec_1.cut.fq`  `etec_2.cut.fq`  
 `etec_bowtie2.sam`

# マッピング結果ファイル(SAMファイル)

ヘッダ(@で始まる)

リファレンス配列に関する情報									
@HD VN:1.0 SO:unsorted									
@SQ SN:ETEC_chr LN:4979619									
@SQ SN:pETEC_80 LN:79237									
@SQ SN:pETEC_35 LN:34367									
@SQ SN:pETEC_73 LN:70609									
@SQ SN:pETEC_6 LN:6199									
@SQ SN:pETEC_74 LN:74224									
@SQ SN:pETEC_5 LN:5033									
@PG ID:bowtie2 PN:bowtie2									
SRR345261.25 89 ETEC_chr 3758170 1 49M = 3758170 0 ACACGGCGCATGGCTG... #####ED=EBDBDE,E... AS:i:-1 XS:i:-1 XN:i:0									
SRR345261.25 133 ETEC_chr 3758170 0 * = 3758170 0 NNNNNNNNNNNNNNNNN... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:-1 XS:i:0 XN:i:0									
SRR345261.50 73 ETEC_chr 4361458 1 49M = 4361458 0 CAACCGTTAACCGAA... :HEGDFHHH@BGC=B... AS:i:0 XS:i:0 XN:i:0									
SRR345261.50 133 ETEC_chr 4361458 0 * = 4361458 0 NNNNNNNNNNNNNNN... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:-2 XS:i:-2 XN:i:0									
SRR345261.75 73 ETEC_chr 4362922 1 49M = 4362922 0 CGGTGGATGCCCTGGC... DDDDBD6<DB>BB> > AS:i:-2 XS:i:-2 XN:i:0									
SRR345261.75 133 ETEC_chr 4362922 0 * = 4362922 0 NNNNNNTTNTGG... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:0 XS:i:0 XN:i:0									
SRR345261.100 73 ETEC_chr 679991 42 49M = 679991 0 GTGGTTTAATGAGTC... GGGGGGGG=ED=EEG... AS:i:0 XN:i:0 XM:i:0									
SRR345261.100 133 ETEC_chr 679991 0 * = 679991 0 NNNNNNCACCGNTAGT... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:0 XN:i:0 XM:i:0									
SRR345261.125 133 ETEC_chr 4376280 42 49M = 4376280 0 CTCAGGATGAGGTCA... EEEE-B<@BDEDE:... AS:i:0 XN:i:0 XM:i:0									
SRR345261.125 133 ETEC_chr 4376280 0 * = 4376280 0 NNNNNNTTCCNTTAG... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:0 XN:i:0 XM:i:0									
SRR345261.150 89 ETEC_chr 779844 42 49M = 779844 0 TTCAAGAAAACCCTGAA... B@RD=ECC?BG@ECC>... AS:i:-5 XN:i:0 XM:i:1									
SRR345261.150 133 ETEC_chr 779844 0 * = 779844 0 CNCGGAGTACNTTGA... #####ED=EBDBDE,E... YT:Z:UP YF:Z:NS AS:i:0 XN:i:0 XM:i:0									
SRR345261.175 83 ETEC_chr 3605306 42 49M = 3605113 -242 CCGCTTGC CGGGCCA... EDE<8?;?@DGDDDE... AS:i:0 XN:i:0 XM:i:0									
SRR345261.175 163 ETEC_chr 3605113 42 49M = 3605306 242 CCGGGTTCTGCTGTG... DGGDGF DGGGGEGD... AS:i:-3 XN:i:0 XM:i:3									
SRR345261.200 77 * 0 0 * 0 0 AAAAAA..... #####ED=EBDBDE,E... YT:Z:UP AS:i:0 XN:i:0 XM:i:0									
SRR345261.200 141 * 0 0 * 0 0 AAAAAA..... #####ED=EBDBDE,E... YT:Z:UP AS:i:-5 XN:i:0 XM:i:1									
SRR345261.225 83 ETEC_chr 2879707 1 49M = 2879600 -156 CACACACCGAGCTGAC... 8?DBEBB DGGGBGC... AS:i:0 XN:i:0 XM:i:0									
SRR345261.225 163 ETEC_chr 2879707 1 49M = 2879707 156 CCCACCTTCCTCCAGT... GGGBDGE GG@GG<G8... AS:i:-1 XN:i:-1 XN:i:0									
SRR345261.250 99 ETEC_chr 4361346 1 49M = 4361525 228 GTACTTTCA CGGGGGA... ECE=>EC?FDG>EGDA... AS:i:0 XN:i:0 XM:i:0									
SRR345261.250 147 ETEC_chr 4361346 1 49M = 4361346 -228 CCGGGCTCAACCTGGG... #####ED=EBDBDE,E... AS:i:0 XN:i:0 XM:i:0									
FLAG	マップされた染色体と位置(*はマップされなかった)	MAPQ	ペアの相手がマップされた染色体(同じなら=)と位置、フラグメントの長さ	CIGAR	リード配列	配列クオリティ値	オプション	AS	アライメントスコア
同じ名前のリード =ペアエンドのリード対							XS	他の位置でのベストスコア	
							YF	リードがfiltering outされた理由	

## 復習: SAMからBAMへの変換

実習用ディレクトリ ~/data/IU

入力

- SAMファイル (さきほどbowtie2によって作成されたもの)  
etec\_bowtie2.sam

- SAMからBAMへ変換する(出力ファイル名: etec\_bowtie2.bam)

```
$ samtools view etec_bowtie2.sam > etec_bowtie2.bam
```

- 作成したBAMファイルをヘッダ付きでSAMに変換してlessで表示する

```
$ samtools view etec_bowtie2.bam > less
```

## 復習:BAMファイルのインデックスづけと検索

実習用ディレクトリ ~/data/IU

入力

- BAMファイル (さきほどSAMからの変換によって作成されたもの)  
etec\_bowtie2.bam

- ◆ リファレンス配列上の位置の順にソートする  
(出力ファイル:etec\_bowtie2\_sorted.bam)

```
$ samtools [ ] etec_bowtie2.bam  
[ ] etec_bowtie2_sorted.bam
```

- ◆ ソートされたBAMファイルに対してインデックスを作成する

```
$ samtools [ ] etec_bowtie2_sorted.bam
```

- ◆ インデックスを使って、リファレンスの染色体配列(染色体名:ETEC\_chr)の10000-12000 の範囲にマッピングされた結果のみを表示する

```
% samtools [ ] etec_bowtie2_sorted.bam  
[ ]
```

## SAM/BAM フォーマット補足

- Bowtie2のデフォルトオプションでマッピングした結果のSAM/BAMファイルは、もとのFASTQファイルに含まれている各リードの配列とクオリティデータをすべて含んでいる。以下のコマンドでSAM/BAMファイルからFASTQファイルを作成できる。

```
$ samtools fastq etec_bowtie2.bam -1 r1.fq -2 r2.fq
```

- リファレンス配列を参照して、配列を記録する代わりにリファレンス上の位置とアライメント情報のみを記録することによって、さらに圧縮率を高めたバイナリ形式としてCRAM形式がある。

```
$ samtools view -C etec_bowtie2.sam -T eco_o139.fa  
-o etec_bowtie2.cram
```

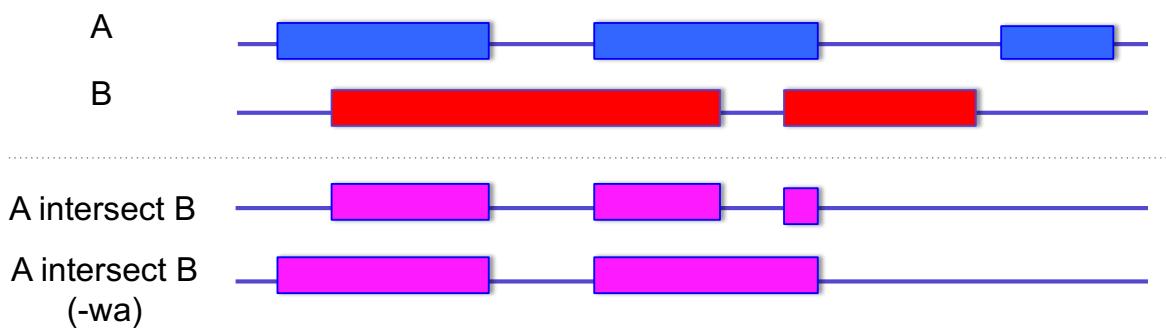
# Bedtools

- BED, GFF(GTF), BAM 形式で記述された「ゲノム上の領域の集合」に対して様々な操作を行うツールを集めたもの。

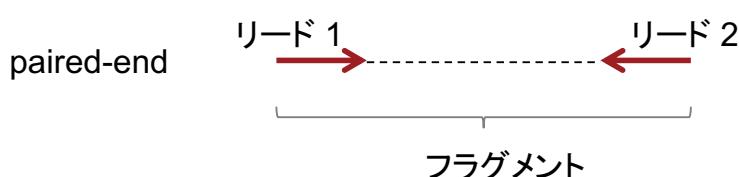
例) intersection

BAM形式のマッピング結果(etec\_bowtie2.bam)の中で、コード領域(eco\_o130\_cds.gff)とオーバーラップしているリードを抽出して、BED形式で表示する。

```
$ bedtools intersect -abam etec_bowtie2.bam  
-b eco_o130_cds.gff -wa -bed | less
```



## Bowtie2のオプション1 ペアエンドリード対の検索



- **-I int** フラグメント長の最小値(default: 0)
- **-X int** フラグメント長の最大値(default: 500)
- **--fr / --rf / --ff** リード1とリード2の相対的な向き (default:fr)  
--fr                    --rf                    --ff  
→ ←                    ← →                    → →

- 条件を満たさない(discordant)リード対もデフォルトでは出力される。その際、2カラム目(FLAG)の2ビット目(ペアが正しくアラインされたか?)に0がセットされる。

# フラグ(FLAG)

- True/Falseの2状態を1/0で表した変数。複数のフラグをまとめて、2進数の数値で表現される。
- フラグ値は10進数で表示されるが、2進数に変換することで解釈される。

## FLAG値

10進数	2進数	解釈
83	01010011	ペアリードである 各リードが適切にアラインされている 逆鎖にマップされている 1番目のリードである

# unix コマンドによる 10進数→2進数の変換

```
% echo 'obase=2;83' | bc
```

```
1010011
```

# samtools を使ったフラグの解釈

```
% samtools flags 83
```

```
0x53 83 PAIRED,PROPER_PAIR,REVERSE,READ1
```

# 各フラグの説明を表示

```
% samtools flags
```

## Paired end readでのFLAG値



2進数表記	samファイルの記載は	10進数表記
01010011	Read1	83
10010011	Read2	147
11111111		255

通常のpaired end seqで  
consistentにアラインしていれば  
この4通りになる

片方しかアラインしていない場合

どちらもアラインしていない場合

0 1 0 1 0 0 1 1	01010011	83
0 1 1 0 0 0 1 1	01100011	99
1 0 0 1 0 0 1 1	10010011	147
1 0 1 0 0 0 1 1	10100011	163
0 1 0 0 1 0 0 1	01001001	73
0 1 0 1 1 0 0 1	01011001	89
0 1 1 0 0 1 0 1	01000101	69
0 1 1 1 0 0 1 0 1	01100101	101
1 0 0 0 1 0 0 1	10001001	137
1 0 0 1 1 0 0 1	10011001	153
1 0 0 0 0 1 0 1	10000101	133
1 0 0 1 0 0 1 0 1	10100101	165
0 1 0 0 0 1 1 0 1	01001101	77
1 0 0 0 0 1 1 0 1	10001101	141

# Samtoolsを用いた フラグによるフィルタリング

- **samtools view -f フラグ値 BAMファイル**

指定した**フラグ値**中で1であるフラグが、**BAMファイル**中のフラグ値でもすべて1になっている行のみを抜き出す。

例) ペアリードでかつ両方が適切にアラインされている行のみを抜き出す

```
% samtools view -f 3 etec_bowtie2_sorted.bam
```

3は2進数で 11 だから、1番目と2番目のフラグが1である行を抜き出す(それ以外のフラグは無視する)。

- **samtools view -F フラグ値 BAMファイル**

指定した**フラグ値**中で1であるフラグが、**BAMファイル**中のフラグ値ではすべて0になっている行のみを抜き出す。

例) ペアリードの両方が適切にアラインされていない行のみを抜き出す

```
% samtools view -F 2 etec_bowtie2_sorted.bam
```

2番目のフラグが0である行を抜き出す。

## Bowtie2のオプション2 アライメント出力のモード

- 一般に、1つのリードは複数の箇所にマップされる。



- **default (best one mode)**

条件を満たすアライメントを検索し、最高スコアのものを1つ出力  
(ただし、検索は完全でないので、最高スコアを取りこぼす可能性はある)  
上記の例では、BまたはD (どちらかがランダムに選ばれる)

- **-k <int>**

条件を満たすアライメントを、見つかった順に指定した数だけ出力  
上記の例で、-k 2 のとき、左から順に見つかるとすると、AとB  
(実際には位置の順に見つかるわけではない)

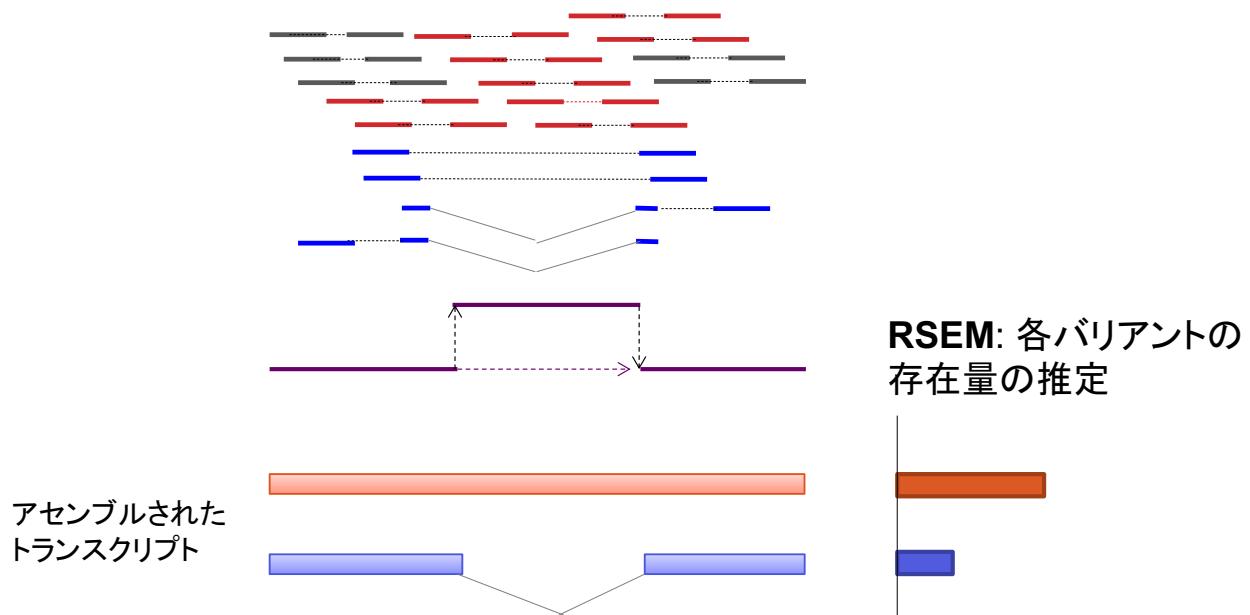
- **-a**

条件を満たすアライメントをすべて出力  
上記の例では、A,B,C,D,E

- **-k や -a を指定したとき、最高スコアでないアライメントには9番目のフラグ (secondary alignment) に1がセットされる**

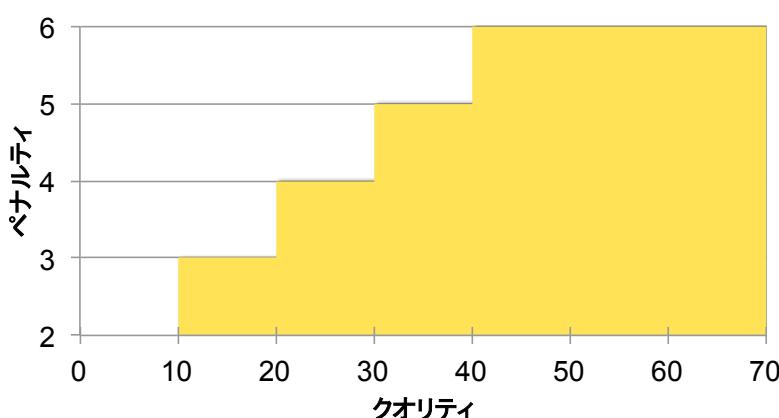
## (参考)デノボ・アセンブルによるRNA-Seq解析

### デノボ・アセンブルによる転写配列の構築



## (参考)Bowtie2におけるアライメントスコア

- マッチは0で、ミスマッチにマイナスのペナルティ(最高スコアが0点)
- ミスマッチペナルティは、クオリティ値に応じて $-2$ から $-6$ の値をとる(下図)
- あいまい塩基(N)のペナルティは $-1$
- ギャップペナルティは、ギャップの長さ $n$ に対して $-(5 + 3n)$
- スコアのカットオフは、長さ $L$ に対して $-0.6(L+1)$



## マッピングクオリティ(MAPQ)

- マッピングクオリティ(MAPQ)値は以下の式で計算される。

$$\text{MAPQ} = -10 \log_{10}(P_e)$$

ただし、 $P_e$ はリードが間違った位置にマップされている確率の推定値。

- MAPQは、リードがその位置にどの程度ユニークにマップされたかを示す指標であり、その位置でのアライメントスコアが、他のすべての位置におけるスコアよりもずっと大きいときに大きくなる。
- Bowtie2のデフォルトでは同じスコアのアライメントが複数の位置で得られた場合、ランダムに一つの位置を出力し、MAPQに低い値を設定する。
- MAPQが低いアライメントの位置は信用できないので、下流の解析の際には捨てた方が良い場合もある。

## Samtoolsを用いた MAPQによるフィルタリング

- `samtools view -q 閾値 BAMファイル名`

MAPQの値が閾値より小さい行を除く

例) MAPQが20以上の行のみを出力

```
$ samtools view -q 20 etec_bowtie2.bam
```

## Bowtie2のオプション3 アライメントのモード

- **--end-to-end** リード配列全長に渡るアライメント(default)

Read:            GACTGGGCGATCTGACTTCG  
              ||||| ||||||||| |||  
Reference: GACTG--CGATCTGACATCG

- **--local** リード配列のうち、類似度の高い一部の領域のみを抜き出してアラインしたもの

Read:            ACGGTTGCCTTAA-TCCGCCACG  
              ||||||| ||| |||||  
Reference: TAACTTGCCTTAAATCCGCCCTGG

## CIGAR文字列

- リードとリファレンス配列とのアライメントの詳細を表す。
- ギャップなしでアラインされている場合、 $nM$  ( $n$ はリード配列の長さ)となる。
- ギャップが入っている場合、 $nD$ (欠失)または $nI$ (挿入) ( $n$ は挿入・欠失の長さ)が入る。

**5M2D4M1I5M**

ref AGACGAGATTA-GCATG  
      ::: :::: :::: :::  
read ACACG--ATTAGGCTTG

- ローカルアライメントのとき、両端の除かれる部分は  $nS$  で、また TopHat などのスプライシングを考慮するアライメントにおいて、イントロンとしてスキップされるリファレンス配列上の領域は  $nN$  で表される。

**5S4M1I5M**

ref ACGGCTGATTA-GCATG  
      :::: :: :::  
read taaccATTAGGCTTG

# インデックスを使った高速検索 ハッシュテーブル

ゲノム配列

ACACGTTACGGT.....

リード配列

CGTTGCA



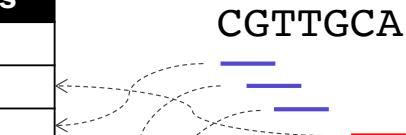
## ①インデックス作成

ハッシュテーブル  
各2-merの出現位置を記録

2-mer	positions
AC	1, 3, 8
CA	2
CG	4, 9
GG	10
GT	5, 11
TA	7
TT	6

## ② インデックスを使った初期検索(seed検索)

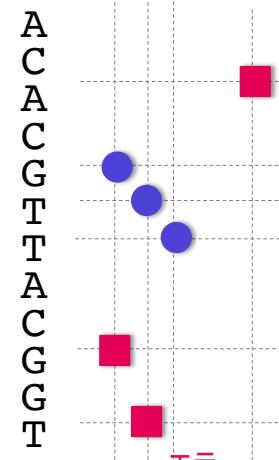
CGTTGCA



## ③ 見つかったseedを延長してアライメント

ACACGTTACGGT.....  
CGTT**GCA**

CGTTGCA



# インデックスを使った高速検索 接尾辞配列(suffix array)

ACACGTTACGGT

接尾辞

1	ACACGTTACGGT
2	CACGTTACGGT
3	ACGTTACGGT
4	CGTTACGGT
5	GTACGGT
6	TTACGGT
7	TACGGT
8	ACGGT
9	CGGT
10	GGT
11	GT
12	T

ソート

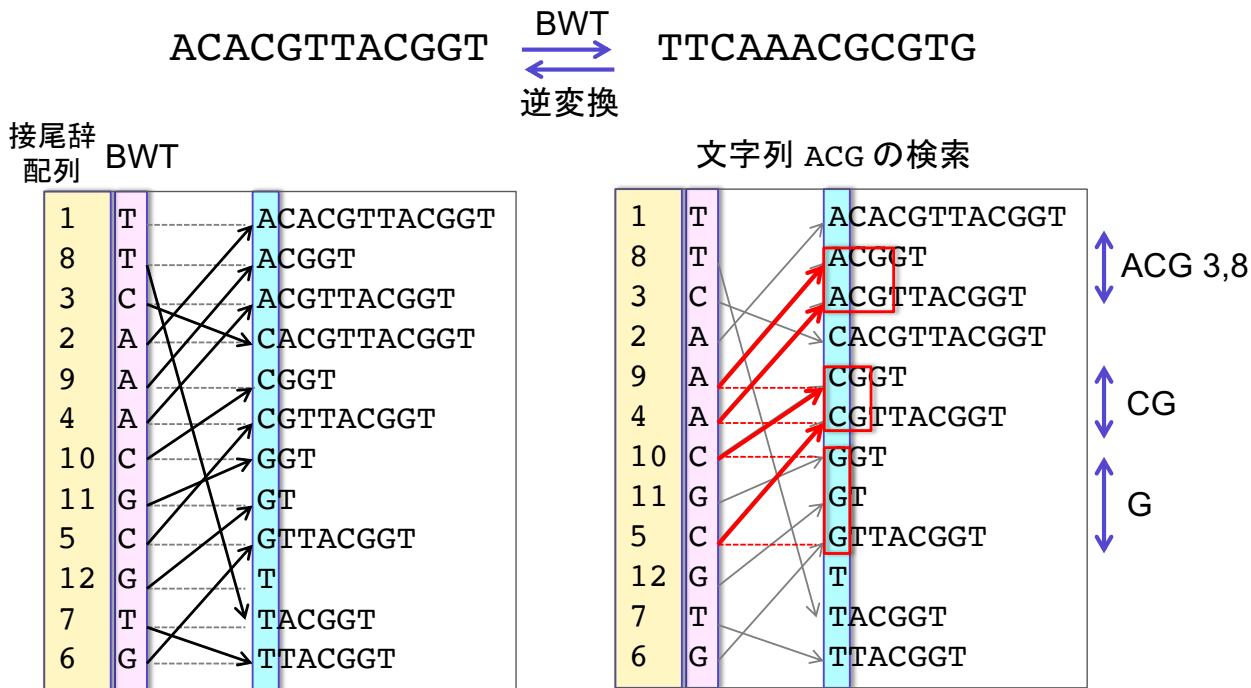
1	ACACGTTACGGT
8	ACGGT
3	ACGTTACGGT
2	CACGTTACGGT
9	CGGT
4	CGTTACGGT
10	GGT
11	GT
5	GTTACGGT
12	T
7	TACGGT
6	TTACGGT

ACG 3,8

CG 4,9

接尾辞  
配列

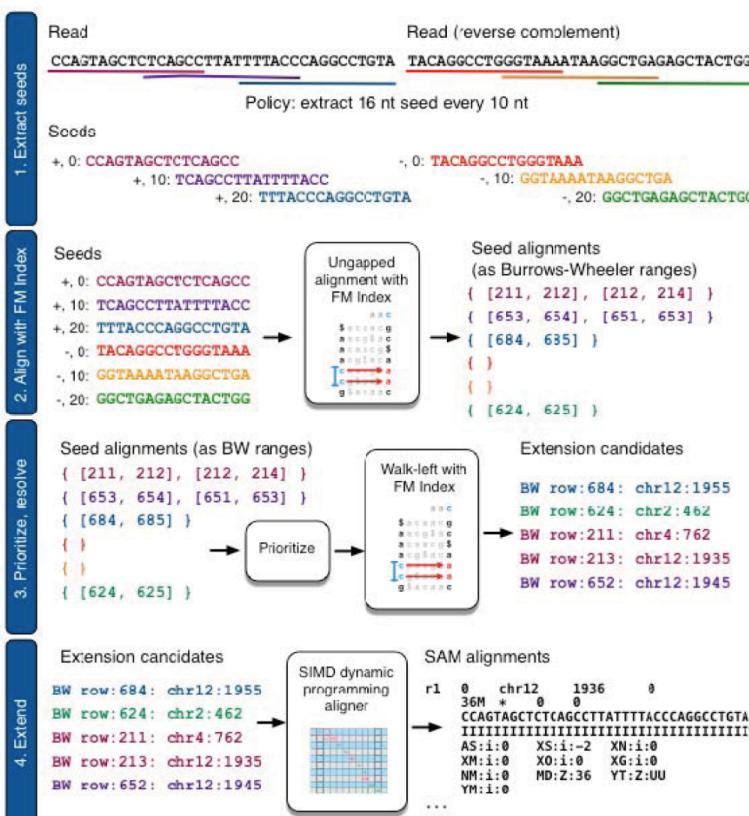
# Burrows-Wheeler 変換 (BWT)に基づく インデックス(FM-Index)



矢印(LF mapping)を辿って元の配列を再構築できる(逆変換)。

メモリ使用量、計算量とも効率のよい検索の実現

# Bowtie2 アルゴリズムの詳細



## 1. Seed 配列の抽出

各リード配列およびその相補配列から  $i$  塩基ごとに  $L$  塩基の配列を抽出してseed配列とする(図では $i=10$ ,  $L=16$ )。

## 2. FM index を用いた検索

各seed配列がゲノム上に出現する位置がBW rangeとして得られる。最大1つのミスマッチを考慮した検索が可能。

### 3. ヒットの優先付け、位置の取得

BW rangeの幅が小さいヒットに高い優先度をつけて、ランダムに候補をピックアップし、ゲノム上の位置を取得。

#### 4. アライメントの計算

得られた位置の周辺で、ギャップ入りのアライメントスコアを計算。これを各候補位置について繰り返して、最高スコアを与えるゲノム上の位置を出力。

## Bowtie2のオプション4

### 検索の精度と速度に関するオプション

- `-N int` `seed` 検索時にミスマッチを許す数(0 or 1)
- `-L int` `seed` の長さ
- `-i func` `seed` をとる間隔(リード長を基に決める式を指定)
- `-D int` 最高スコアが更新されないときアライメント計算を打ち切るまでの回数
- `-R int` リードが高反復のseedをもつときにre-seedを行う最大回数

上記のオプションを同時に設定するpreset optionがある。高速(低感度)→高感度(低速)の順に4段階のオプションが用意されている。

- end-to-endモードの場合 (default: `sensitive`)  
`--very-fast` / `--fast` / `--sensitive` / `--very-sensitive`
- localモードの場合 (default: `sensitive-local`)  
`--very-fast-local` / `--fast-local` / `--sensitive-local` / `--very-sensitive-local`

# NGS基本ツールIGV

基礎生物学研究所  
生物機能解析センター  
山口勝司

## データ可視化ツール・IGVの紹介・実習

The screenshot shows the homepage of the Integrative Genomics Viewer (IGV) website. On the left is a sidebar with links for Home, Downloads, Documents, Hosted Genomes, FAQ, IGV User Guide, File Formats, Release Notes, IGV for iPad, Credits, Contact, and a search bar. Below the sidebar is the Broad Institute logo and copyright information. The main content area features a large banner with the text "Integrative Genomics Viewer" and a screenshot of the software interface. Below the banner are sections for "Overview", "Downloads", and "Funding". The "Overview" section describes IGV as a high-performance visualization tool for exploring genomic datasets. The "Downloads" section provides a download link. The "Funding" section credits the National Cancer Institute, National Institute of General Medical Sciences, National Institutes of Health, and the Starr Cancer Consortium. At the bottom of the page is a red URL: <https://www.broadinstitute.org/igv/>.

# なぜIGVを取り上げるか

## データ可視化ツール

- ・自分のパソコン(ローカル環境)にインストールして使うタイプ
- ・サーバーに構築して、ネットワークで使うタイプ

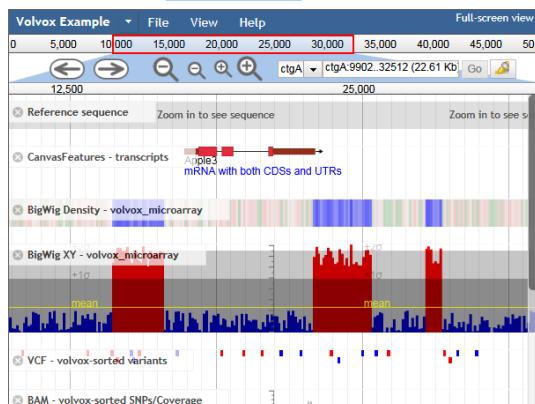
### The JBrowse Genome Browser

JBrowse is a fast, embeddable genome browser built completely with JavaScript and HTML5, with optional run-once data formatting tools written in Perl.

#### Featured Post

[Exploring structural variation using JBrowse](#) by Richard Finkers

#### Latest Release – [JBrowse 1.11.6](#)



コミュニティに広く利用、あるいはウェブ公開を目的とするには良いが、ネットワーク・情報セキュリティの高度な知識も要求される。

より大容量なデータに対応できる。

管理者的な人がいて、その人がやってくれるなら、これも良いが。

もっとお手軽なものとしてIGVを紹介

## 可視化ツールに求められるものは何か

膨大なデータを如何に直感的に理解できるようにするか  
sortや絞り込みができる表データと対比双璧

- ・配列、GC ratio、遺伝子情報
- ・遺伝子発現情報
- ・SNPの位置情報・頻度情報
- ・様々なデータの精度情報

レファレンス配列 / gene model / gene annotationとNGSデータを並べて比較  
複数のデータセットを並べて比較

色々なデータ(variant, 発現, ChIP, BSseq等々)を、様々なスケールで  
比較・統合的に解釈できるようにしたい

ゲノムviewerに自分のデータを乗せ、  
統合的直感的に解釈できること

# 可視化ツールをどう選ぶか

選択の基準

genome data viewing に求められるもの

取捨選択の基準

1. 無料 / 有料 / 基本無料
2. 個人的レベルの使用 / コミュニティーレベルの使用
3. 見るだけ/自分から色々工夫
4. アクセスのしやすさ・使いやすさ
  - 導入に必要なコンピュータスペック
  - マニュアルは分かりやすいか
  - 情報の多さ
  - 利用の簡便さ
  - 使っている人が近くにいるか

## Integrative Genomics Viewer(IGV)

### お手軽ツール

- ・アカデミックウェアで無料
- ・コミュニティーでの利用者が多いから、情報も多い
- ・javaのプログラムなので、オールプラットフォーム対応
- ・マニュアルは親切、サンプルデータのある
- ・WEBサーバーではなく、PCレベルでできる
- ・データ閲覧環境の共有が可能

誰もが簡単に使えるものが良い。

The screenshot shows the IGV website homepage. At the top left is the IGV logo and navigation links for Home, Downloads, and Documents. To the right is a large banner with the text "Integrative Genomics Viewer" and a preview of the software's interface showing multiple tracks of genomic data. Below the banner are sections for Overview, Downloads, and Citing IGV. The Citing IGV section contains two citations with a red box around the second one:

**Overview**

The Integrative Genomics Viewer (IGV) is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets. It supports a wide variety of data types, including array-based and next-generation sequence data, and genomic annotations.

**Downloads**

Download the IGV desktop application and igvtools.

**Citing IGV**

To cite your use of IGV in your publication:

James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. [Integrative Genomics Viewer](#). *Nature Biotechnology* 29, 24–26 (2011)

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. [Integrative Genomics Viewer \(IGV\): high-performance genomics data visualization and exploration](#). *Briefings in Bioinformatics* 14, 178–192 (2013).

The screenshot shows a Nature Biotechnology journal article. The header features the magazine's logo and navigation links for Login, Search, and Advanced search. The main title is "Integrative genomics viewer". Below the title is a brief summary of the research, the authors' names (James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, & Jill P. Mesirov), and the publication details (Nature Biotechnology 29, 24–26 (2011) | doi:10.1038/nbt.1754). The article was published online on 10 January 2011. A "To the Editor" section follows. The text discusses the rapid improvements in sequencing and array-based platforms, the volume of data, and the need for efficient and intuitive visualization tools. To the right of the article are several sidebar boxes: one for "Subscribe today, save 50% and receive 51 weekly issues of Nature in print, online and mobile.", another for "Citations to this article" (Crossref (10), Scopus (12), Web of Science (0)), and a third for "Science jobs from naturejobs" listing "Faculty Position" at Harvard Medical School and "Ramalingaswami Re-Entry Fellowship" at the Ministry of Science & Technology, Government of India.

The screenshot shows the IGV website homepage. At the top left is the IGV logo and navigation links for Home, Downloads, and Documents. A sidebar on the left contains links for Hosted Genomes, FAQ, IGV User Guide (which is highlighted with a red border), File Formats, Release Notes, Credits, Contact, and a search bar. Below the sidebar is a copyright notice for © 2013-2018 Broad Institute, and the Regents of the University of California. The main content area features a large banner with the text "Integrative Genomics Viewer" and a screenshot of the software interface. Below the banner are sections for Overview, Downloads, Citing IGV, and Funding.

**Overview**

The Integrative Genomics Viewer (IGV) is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets. It supports a wide variety of data types, including array-based and next-generation sequence data, and genomic annotations.

**Downloads**

Download the IGV desktop application and igvtools.

**Citing IGV**

To cite your use of IGV in your publication:

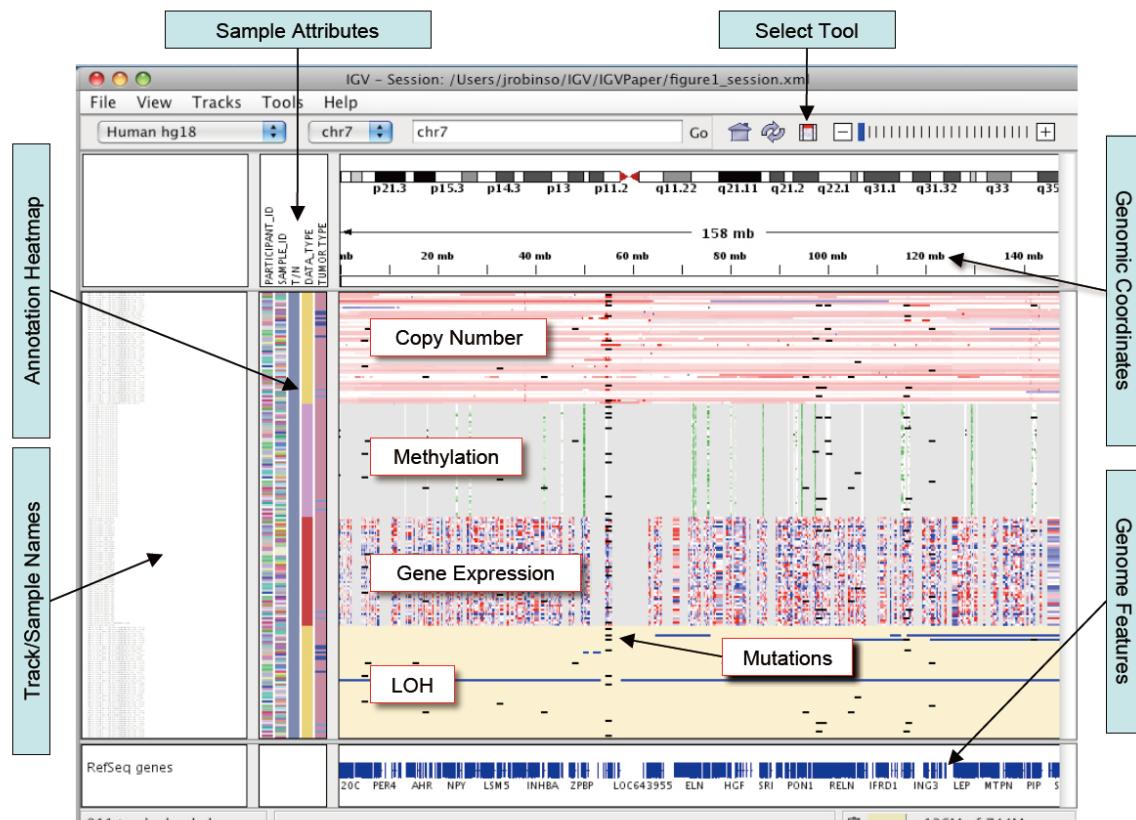
James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. *Integrative Genomics Viewer*. *Nature Biotechnology* 29, 24–26 (2011)

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. *Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration*. *Briefings in Bioinformatics* 14, 178–192 (2013).

The screenshot shows the IGV User Guide page. At the top left is the IGV logo and navigation links for Home, Downloads, and Documents. A sidebar on the left contains a detailed tree view of the User Guide chapters, including Hosted Genomes, FAQ, IGV User Guide (which is expanded), User Interface, Starting IGV, Navigating, Loading a Genome, External Control of IGV, Viewing the Reference Genome, Loading Data and Attributes, Viewing Data, Viewing Alignments, Viewing Variants, Gene List View, Regions of Interest, Sample Attributes, Sorting, Grouping, and Filtering, Saving and Restoring Sessions, Server Configuration, Motif Finder, igvtools, BLAT search, File Formats, Release Notes, and Credits. Below the sidebar is a link to a printer-friendly HTML version of the whole User Guide. The main content area lists several user interface topics.

- [User Interface](#)
- [Starting IGV](#)
- [Navigating](#)
- [Loading a Genome](#)
- [External Control of IGV](#)
- [Viewing the Reference Genome](#)
- [Loading Data and Attributes](#)
- [Viewing Data](#)
- [Viewing Alignments](#)
- [Viewing Variants](#)
- [Gene List View](#)
- [Regions of Interest](#)
- [Sample Attributes](#)
- [Sorting, Grouping, and Filtering](#)
- [Saving and Restoring Sessions](#)
- [Server Configuration](#)
- [Motif Finder](#)
- [igvtools](#)
- [BLAT search](#)

[User Interface >](#)



Nature Biotech. 29:24–26 (2011) Supplement figureからの抜粋

**Home**

# Integrative Genomics Viewer

The Integrative Genomics Viewer (IGV) is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets. It supports a wide variety of data types, including array-based and next-generation sequence data, and genomic annotations.

**Downloads**

Download the IGV desktop application and igvtools.

**Citing IGV**

To cite your use of IGV in your publication:

James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. [Integrative Genomics Viewer](#). *Nature Biotechnology* 29, 24–26 (2011)

Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov. [Integrative Genomics Viewer \(IGV\): high-performance genomics data visualization and exploration](#). *Briefings in Bioinformatics* 14, 178–192 (2013).

レジストレーションが必要

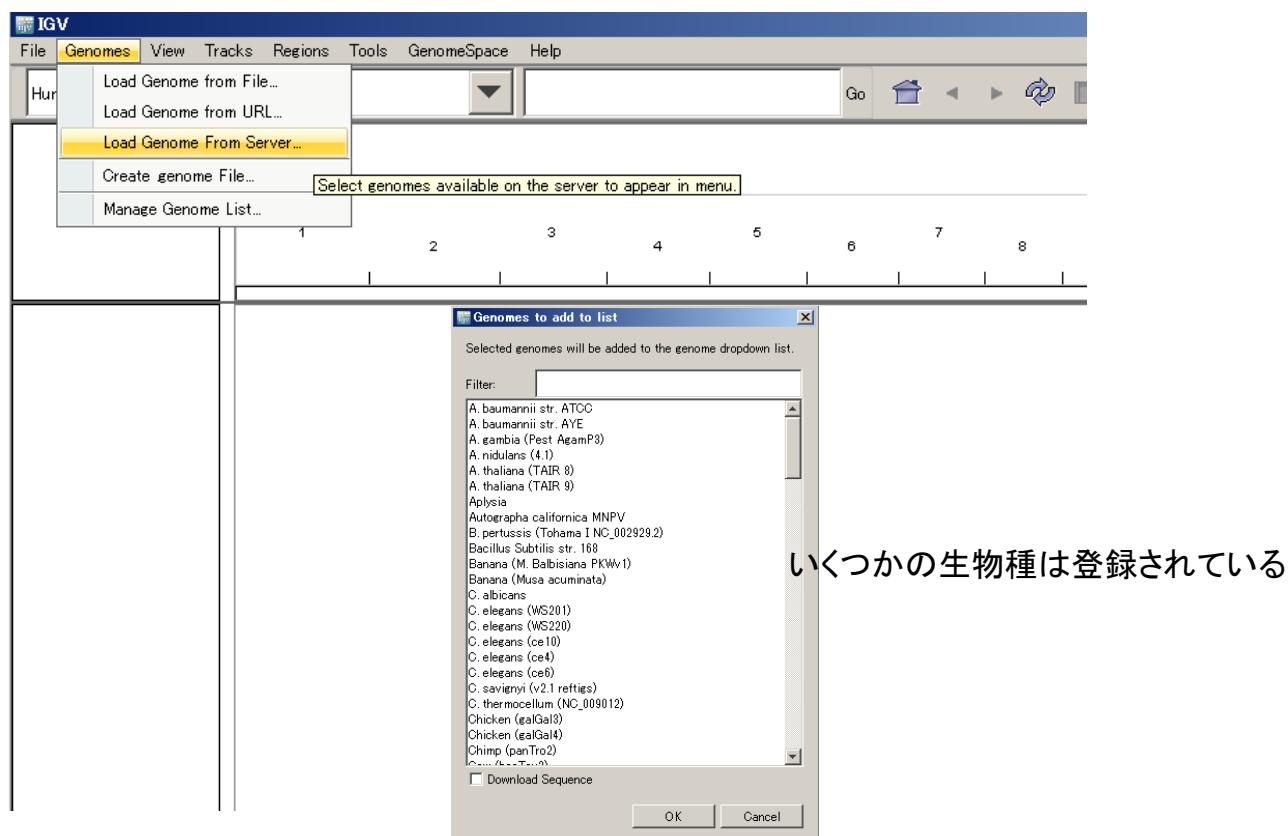
## Downloads

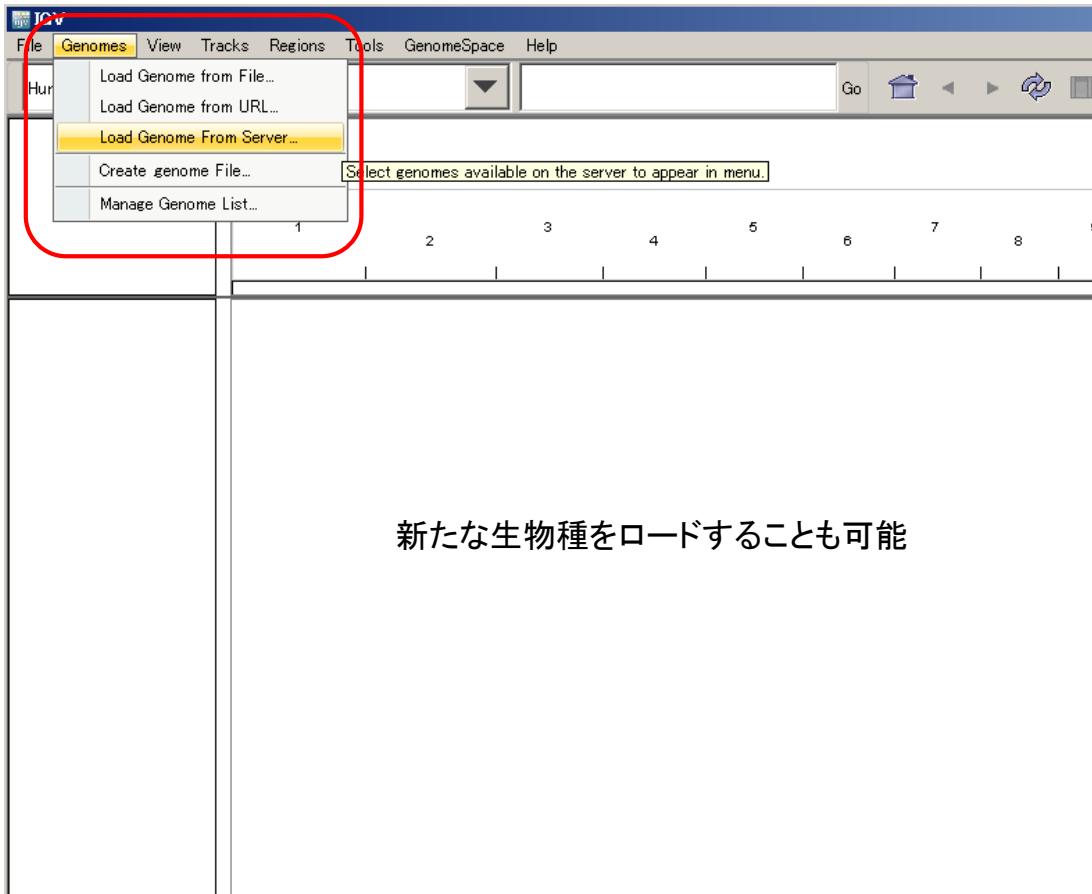
### Integrative Genomics Viewer - IGV 2.4

#### Install IGV

Use one of the following 4 options to install and run the current version of IGV. **NOTE: IGV 2.4.x requires Java 8 or later.**

1.  Download and unzip the [Mac App Archive](#), then double-click the IGV application to run it. The application can be moved to the Applications folder, or anywhere else.
2.  Download and unzip the [Windows Zip Archive](#), then double-click the `igv.bat` file to start IGV. A black console window will appear, followed by the IGV application. **Note:** Windows users with **high resolution screens** should use this version -- it includes a modified Java executable for use with high-resolution screens.
3.  Download and unzip the [Binary Distribution archive](#), IGV is launched from a command prompt -- follow the instructions in the `readme` file. To launch IGV on Mac or Linux use the shell script `igv.sh`. On Windows use `igv.bat`.
4. Click on one of the *Launch* buttons below to download a `.jnlp` file and execute the file using **Java Web Start (JWS)**.  
**NOTE: this option does *not* work with Java 9.**
  - **Mac users:** If you are notified of security errors that prevent launching IGV, try the following:
    - Right-click on the downloaded `.jnlp` file; select *Open With > Java Web Start*; dismiss the warnings.
    - After IGV has been run this way at least once from the `.jnlp` file, you can double-click on the file to launch.
  - **Windows users:** To run with more than 1.2 GB of memory on Windows you must install 64-bit Java. **Most Windows installs do not include 64-bit Java by default, even if the operating system is 64-bit.** Attempting to use the 2GB or greater launch options with 32-bit Java will result in the error "could not create virtual machine".





新たな生物種をロードすることも可能

ゲノムViewerなので次世代DNAシーケンサーのデータに限定されない。  
マイクロアレイの結果や、ゲノムアノテーションの情報も随時表示できる。

対応するファイル形式に応じて、表示方法が決まる。

#### File Formats

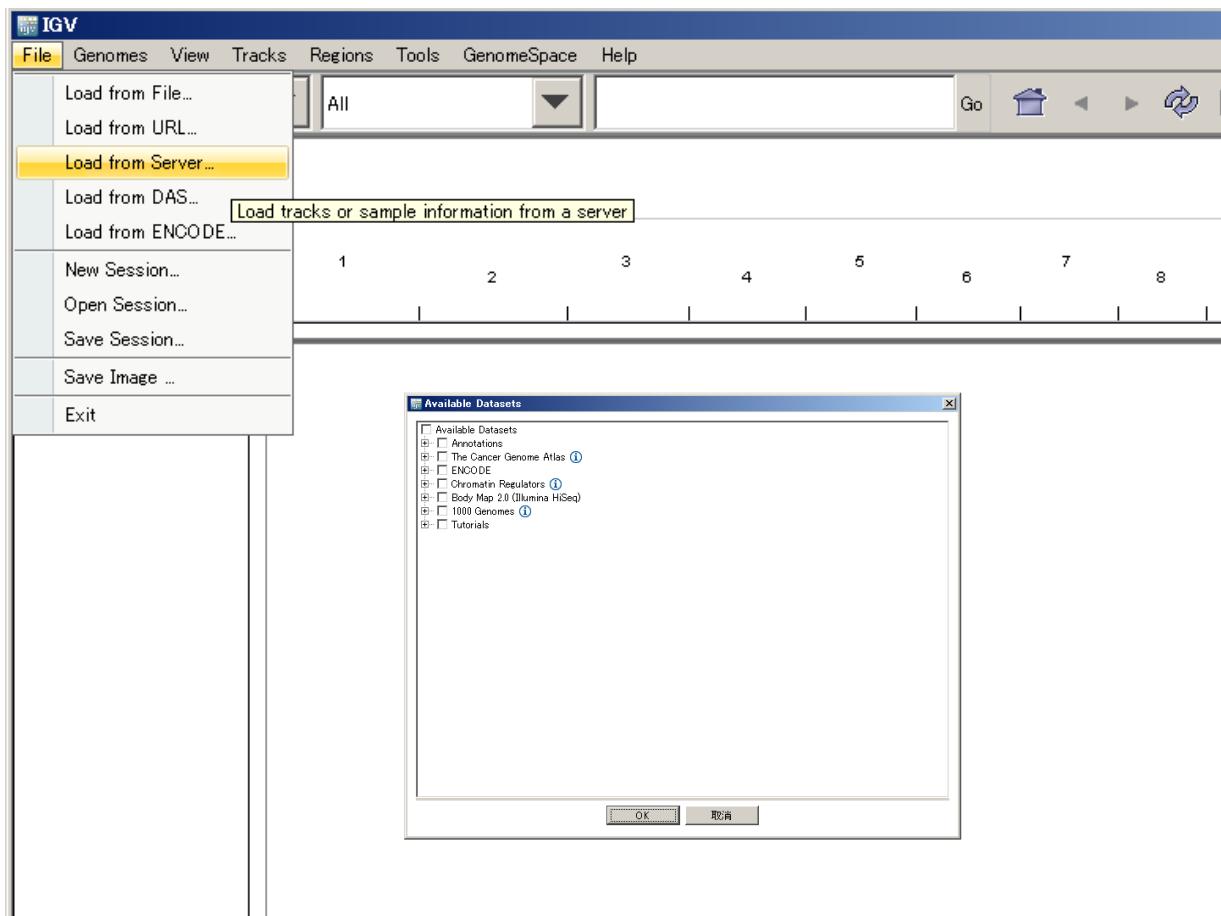
- [File Extension Identifies Format](#)
- [Recommended File Formats](#)
- [BAM](#)
- [BED](#)
- [BedGraph](#)
- [bigBed](#)
- [bigWig](#)
- [Birdsuite Files](#)
- [broadPeak](#)
- [CBS](#)
- [Chemical Reactivity Probing Profiles](#)
- [chrom sizes](#)
- [CN](#)
- [Custom File Formats](#)
- [Cytoband](#)
- [FASTA](#)
- [GCT](#)
- [CRAM](#)
- [genePred](#)
- [GFF/GTF](#)
- [GISTIC](#)
- [Goby](#)
- [GWAS](#)
- [IGV](#)
- [LOH](#)
- [MAF \(Multiple Alignment Format\)](#)
- [MAF \(Mutation Annotation Format\)](#)
- [Merged BAM File](#)
- [MUT](#)
- [narrowPeak](#)
- [PSL](#)
- [RES](#)
- [RNA Secondary Structure Formats](#)
- [SAM](#)
- [Sample Info \(Attributes\) file](#)
- [SEG](#)
- [SNP](#)
- [TAB](#)
- [TDF](#)
- [Track Line](#)
- [Type Line](#)
- [VCF](#)
- [WIG](#)

#### File Formats

IGV supports a number of different file formats for experimental data and genome annotations. For a complete list of supported formats see <http://www.broadinstitute.org/igv/FileFormats>. The following table shows the recommended file formats for a number of common data types.

Source Data	Recommended File Formats
ChIP-Seq, RNA-Seq	WIG, TDF
Copy number	CN, SNP, TDF, canary_calls (Birdsuite)
Gene expression data	GCT, RES, TDF
Genome annotations	GFF, BED, GTF, PSL, UCSC table format
GISTIC data	GISTIC
LOH data	LOH, TDF
Mutation data	MUT, MAF
Variant calls	VCF
RNAi data	GCT
Segmented data	SEG, CBS
Sequence alignment data	BAM, SAM, PSL
Any numeric data	IGV, WIG, TDF
Sample metadatada	Tab-delimited sample info file

## 公開情報のviewerとして



## その他の便利機能

### セッションの保存

表示しているデータの読み込み状況を、それごと保存。

セッションをロードすることで、意図した画面を表示できる。

データセットが揃っていること、フォルダー構造が同一である必要がある。

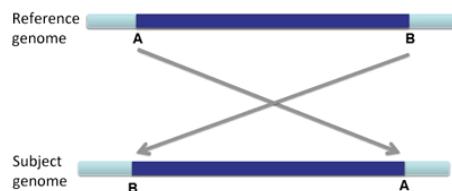
### バッチ処理

重要領域の画面スナップショットを自動で取ったりできる。

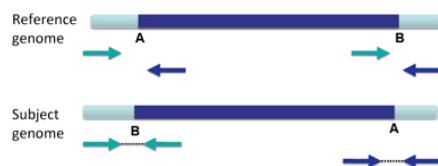
```
new
load myfile.bam
snapshotDirectory mySnapshotDirectory
genome hg18
goto chr1:65,289,335-65,309,335
sort position
collapse
snapshot
goto chr1:113,144,120-113,164,120
sort base
collapse
snapshot
```

## Inversions

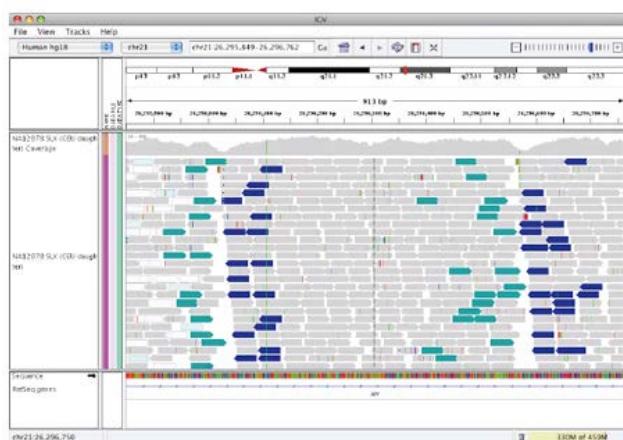
An inversion is a large section of DNA that is reversed in the subject genome compared to the reference genome.



When an inversion shows up in paired-end reads, the reads are distinctively variant from the reference genome.



This appears in IGV as shown below.



## Interpreting Color by Insert Size

The inferred insert size can be used to detect structural variants, such as:

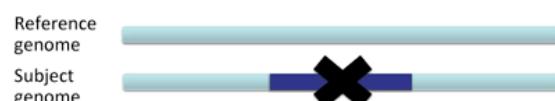
- deletions
- insertions
- inter-chromosomal rearrangements

IGV uses color coding to flag anomalous insert sizes. When you select Color alignments>by insert size in the popup menu, the default coloring scheme is:

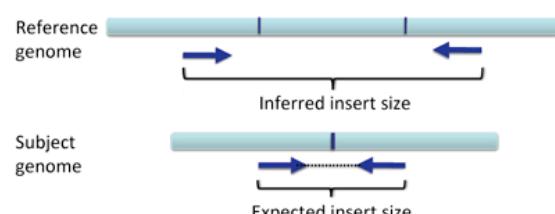
- Red arrow for an insert that is larger than expected
- Blue arrow for an insert that is smaller than expected
- Chromosome color key for paired end reads that are coded by the chromosome on which their mates can be found

## Deletions

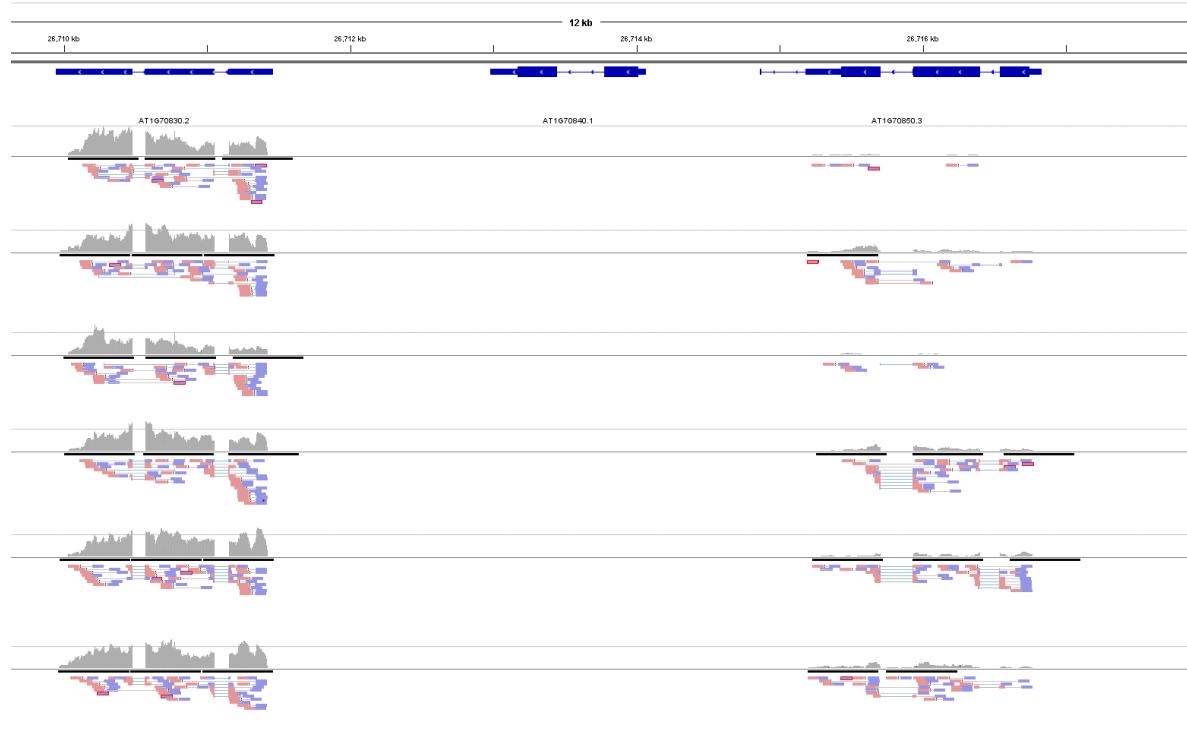
A deletion is a large section of DNA that is absent in the subject genome compared to the reference genome.



The "expected" insert size is the insert size obtained in sequencing the subject genome. The "inferred" insert size is the insert size that would result in the reference genome, assuming the same pair of reads.

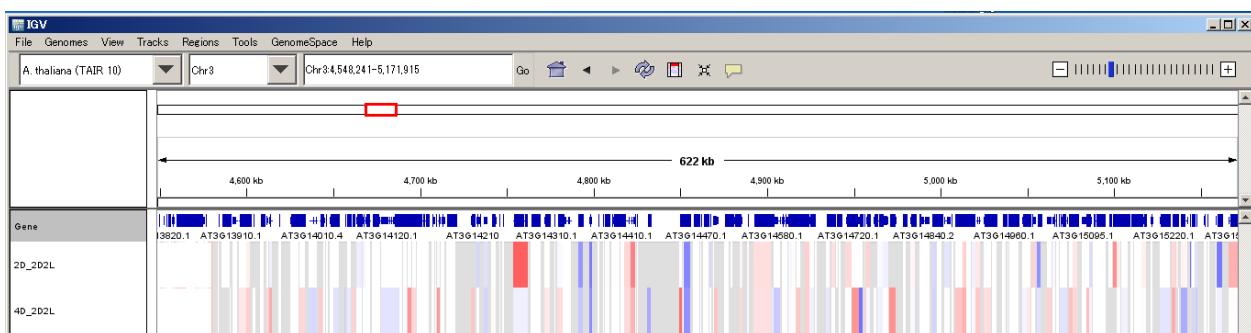


# RNA-Seqのデータ表示させる



# GCTファイルでgene ローカスの発現情報を図示

#			
#			
Name	Description	2D_2D2L	4D_2D2L
ANAC001	@Chr1:3630-5899	-2.60184	-2.60956
DCL1	@Chr1:23145-33153	-0.742675	-1.5642
MIR838A	@Chr1:23145-33153	0	0
AT1G01073	@Chr1:44676-44787	0	0
IQD18	@Chr1:52238-54692	-1.93871	-1.13128
AT1G01115	@Chr1:56623-56740	0	0
GIF2	@Chr1:72338-74737	-0.251287	-0.616679
AT1G01180	@Chr1:75582-76758	0.45929	-0.809567
AT1G01210	@Chr1:88897-89745	1.6964	0.857196
FKGP	@Chr1:91375-95651	-0.174589	0.725947
AT1G01240	@Chr1:99893-101834	-0.226384	-0.936641
AT1G01260	@Chr1:108945-111609	-0.161848	0.315699
CYP703A2	@Chr1:112262-113947	0	0
CNX3	@Chr1:114285-116108	0.111249	-0.551359
AT1G01300	@Chr1:116942-118764	-0.68348	0.108578



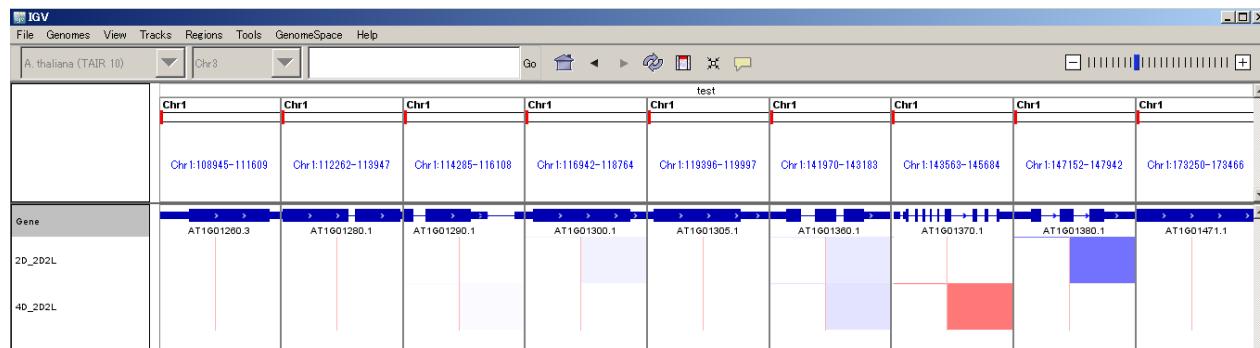
Gene listを定義して  
サンプルごと  
条件ごと  
の発現・発現変動を  
カラーマップできる

The Gene Lists functionalities in IGV allow you to view lists of genes or loci side-by-side irrespective of their genomic location.

**Loading/Defining Gene Lists**

To load or define a new gene/locus list, select Regions >Gene Lists....

This opens a window for selecting an existing list or creating a new list.



## IGV実習

### Downloads

#### Integrative Genomics Viewer - IGV 2.4

##### Install IGV

Use one of the following 4 options to install and run the current version of IGV. *NOTE: IGV 2.4.x requires Java 8 or later.*

1. Download and unzip the [Mac App Archive](#), then double-click the IGV application to run it. The application can be moved to the Applications folder, or anywhere else.
2. Download and unzip the [Windows Zip Archive](#), then double-click the `igv.bat` file to start IGV. A black console window will appear, followed by the IGV application. Note: Windows users with **high resolution screens** should use this version – it includes a modified Java executable for use with high-resolution screens.
3. Download and unzip the [Binary Distribution archive](#). IGV is launched from a command prompt -- follow the instructions in the `readme` file. To launch IGV on Mac or Linux use the shell script `igv.sh`. On Windows use `igv.bat`.
4. Click on one of the *Launch* buttons below to download a `.jnlp` file and execute the file using Java Web Start (JWS). *NOTE: this option does **not** work with Java 9.*

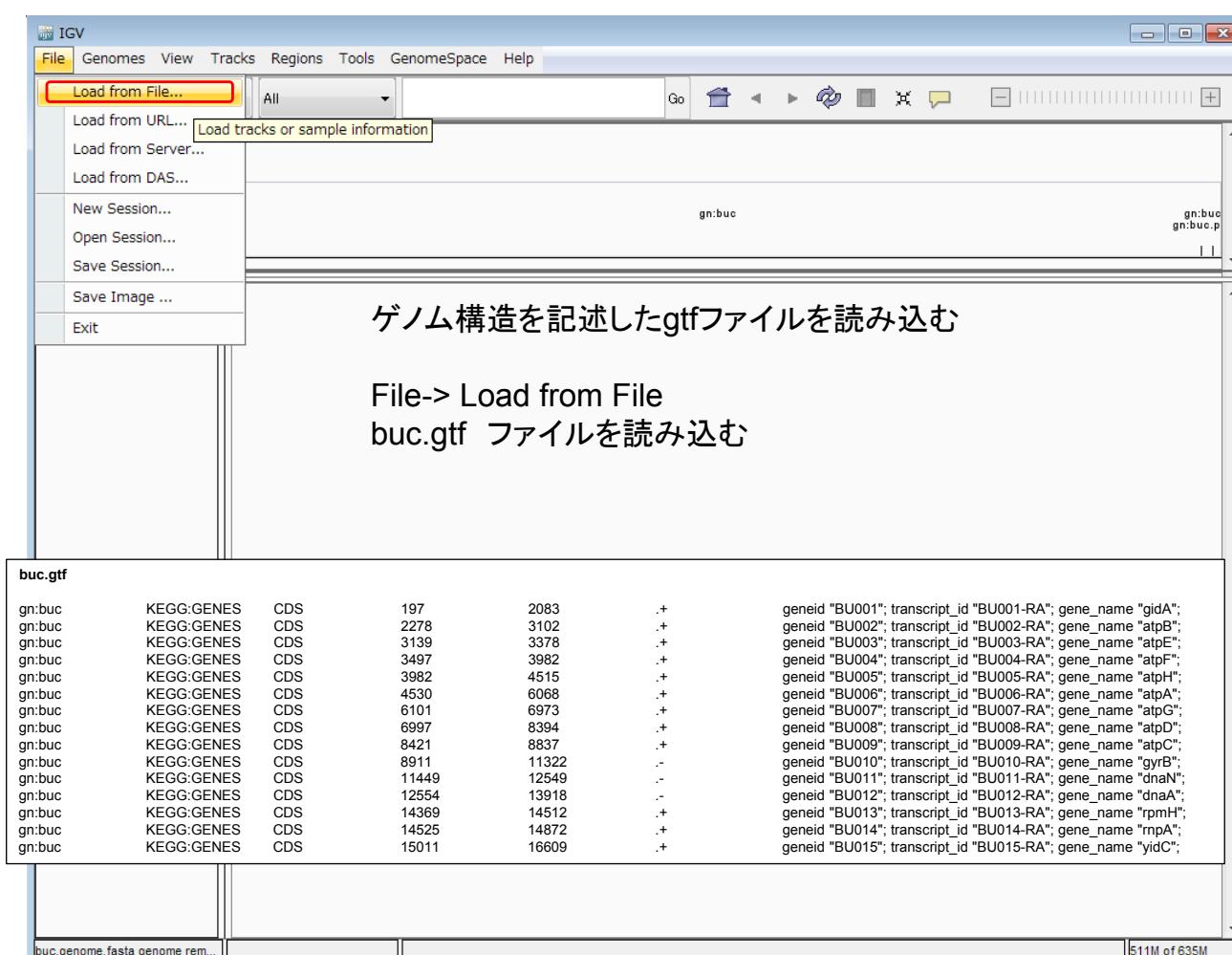
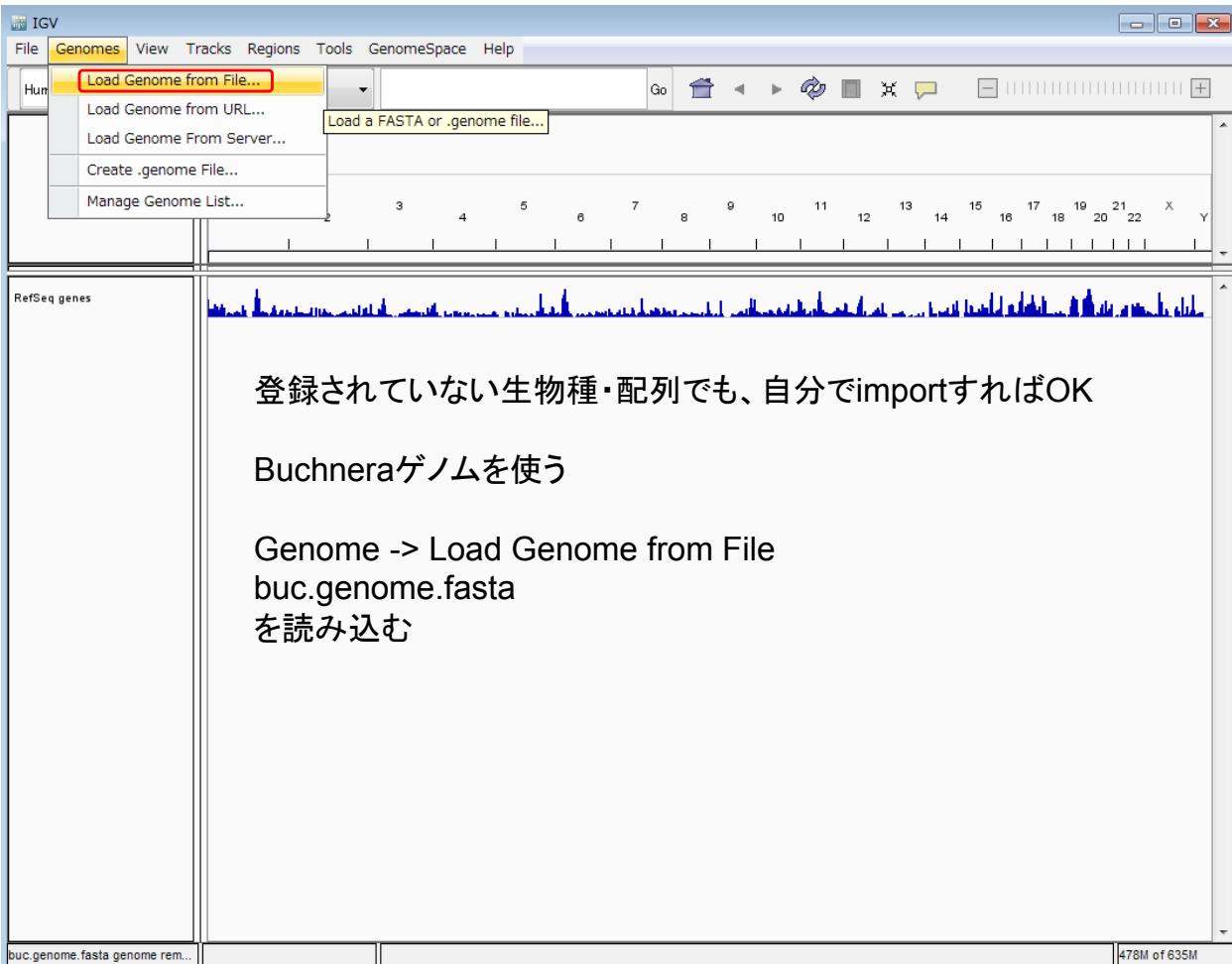
- Mac users:** If you are notified of security errors that prevent launching IGV, try the following:
  - Right-click on the downloaded `.jnlp` file, select *Open With* > *Java Web Start*, dismiss the warnings.
  - After IGV has been run this way at least once from the `.jnlp` file, you can double-click on the file to launch.
- Windows users:** To run with more than 1.2 GB of memory on Windows you must install 64-bit Java. **Most Windows installs do not include 64-bit Java by default, even if the operating system is 64-bit.** Attempting to use the 2GB or greater launch options with 32-bit Java will result in the error "could not create virtual machine".

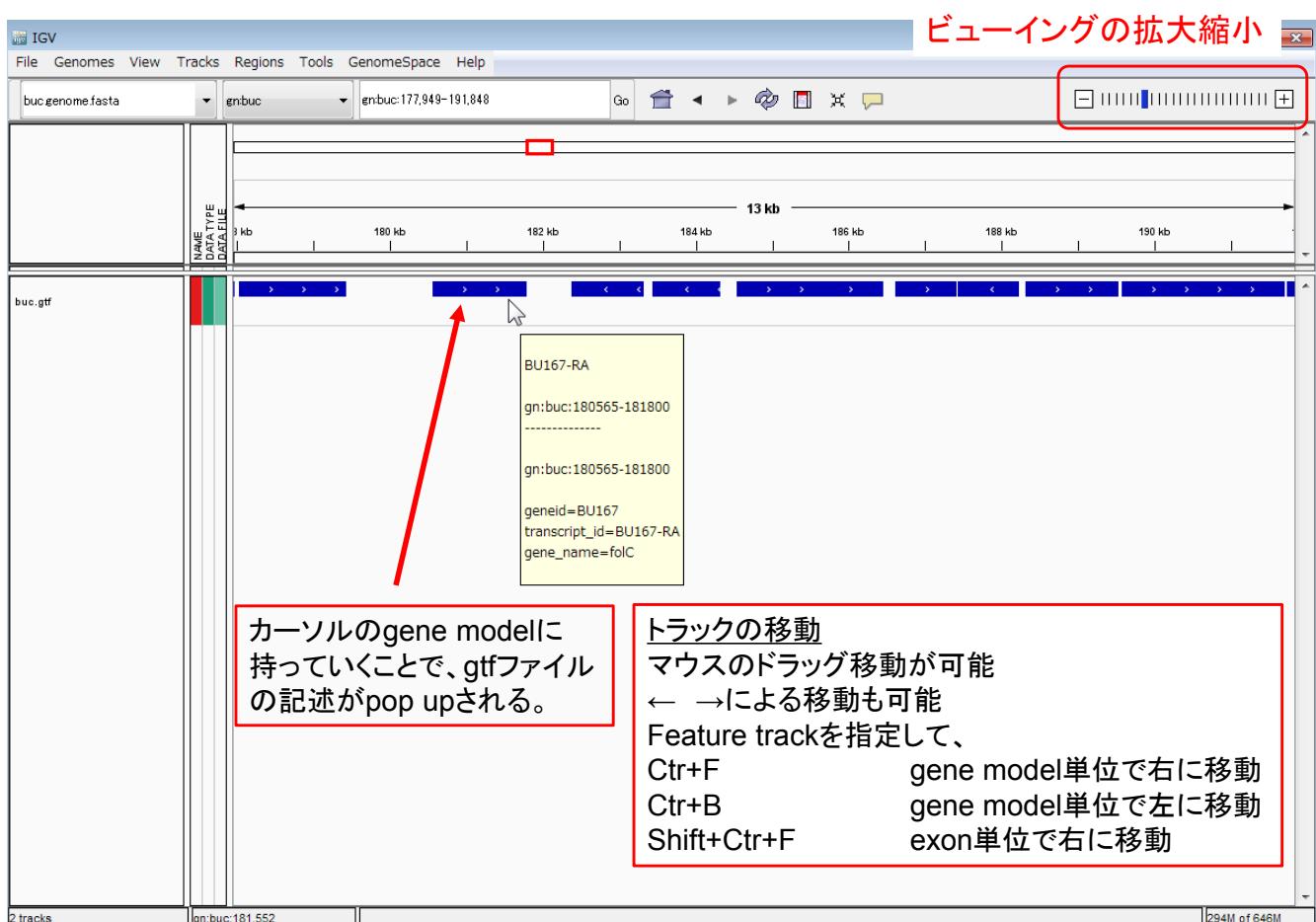
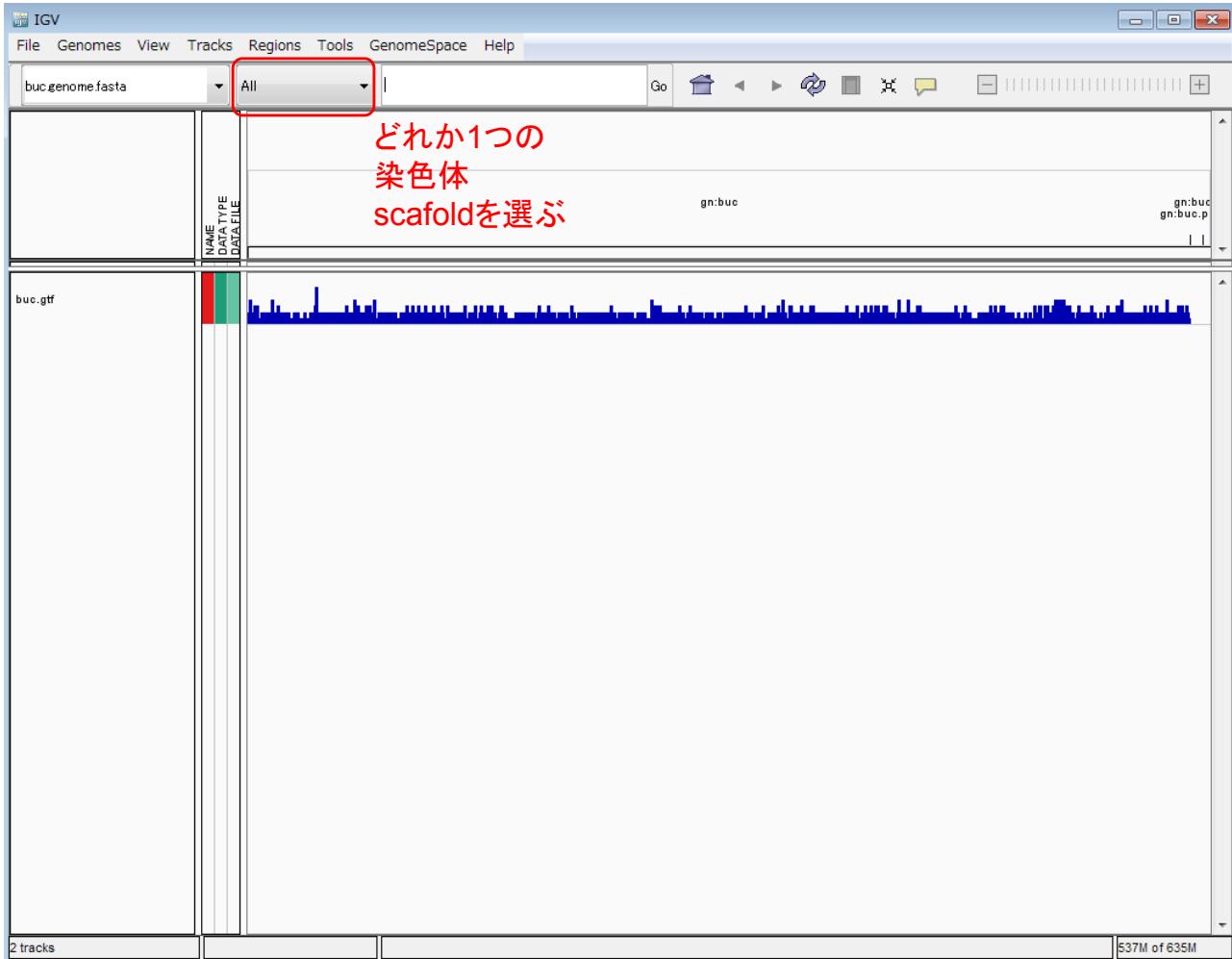


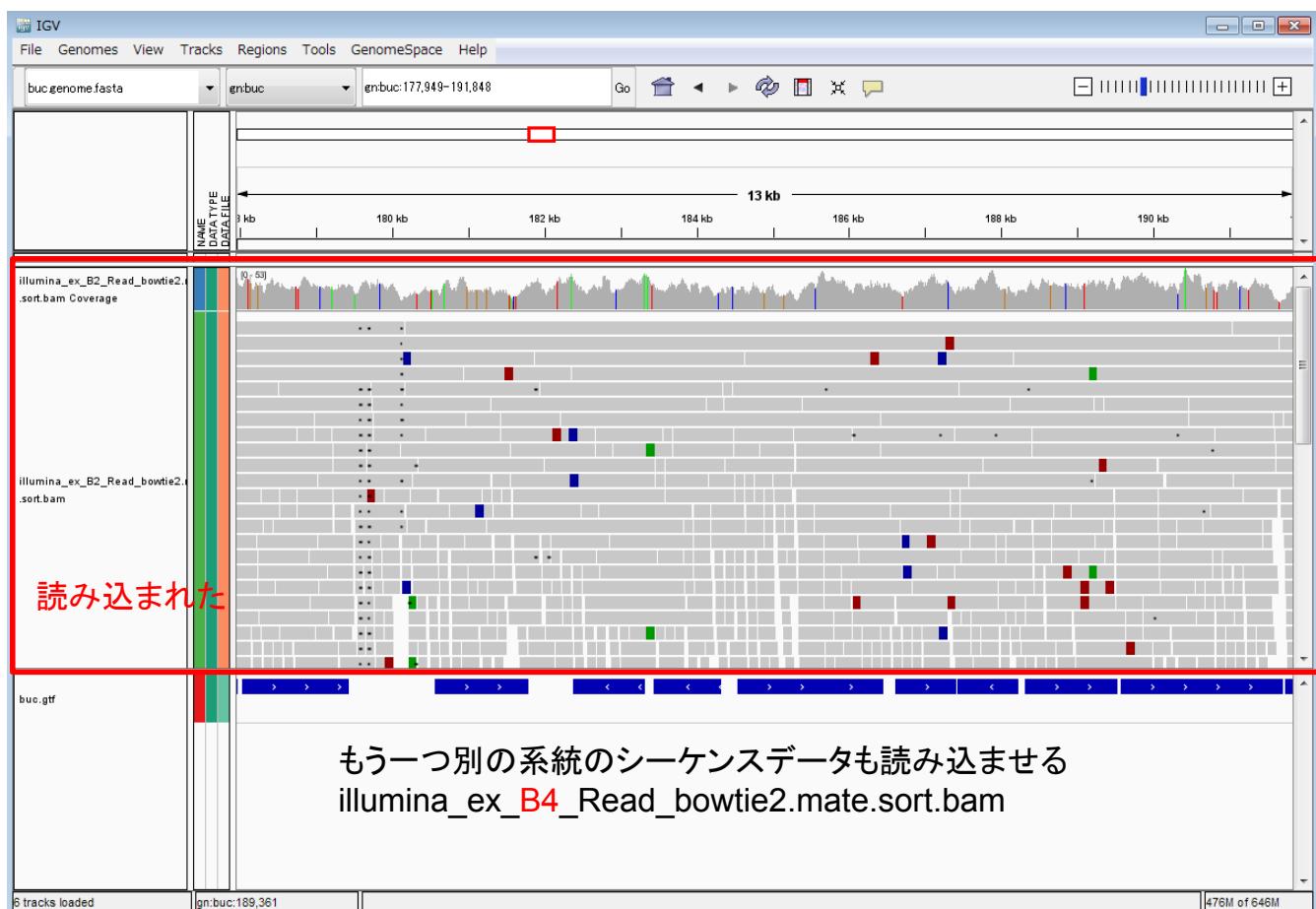
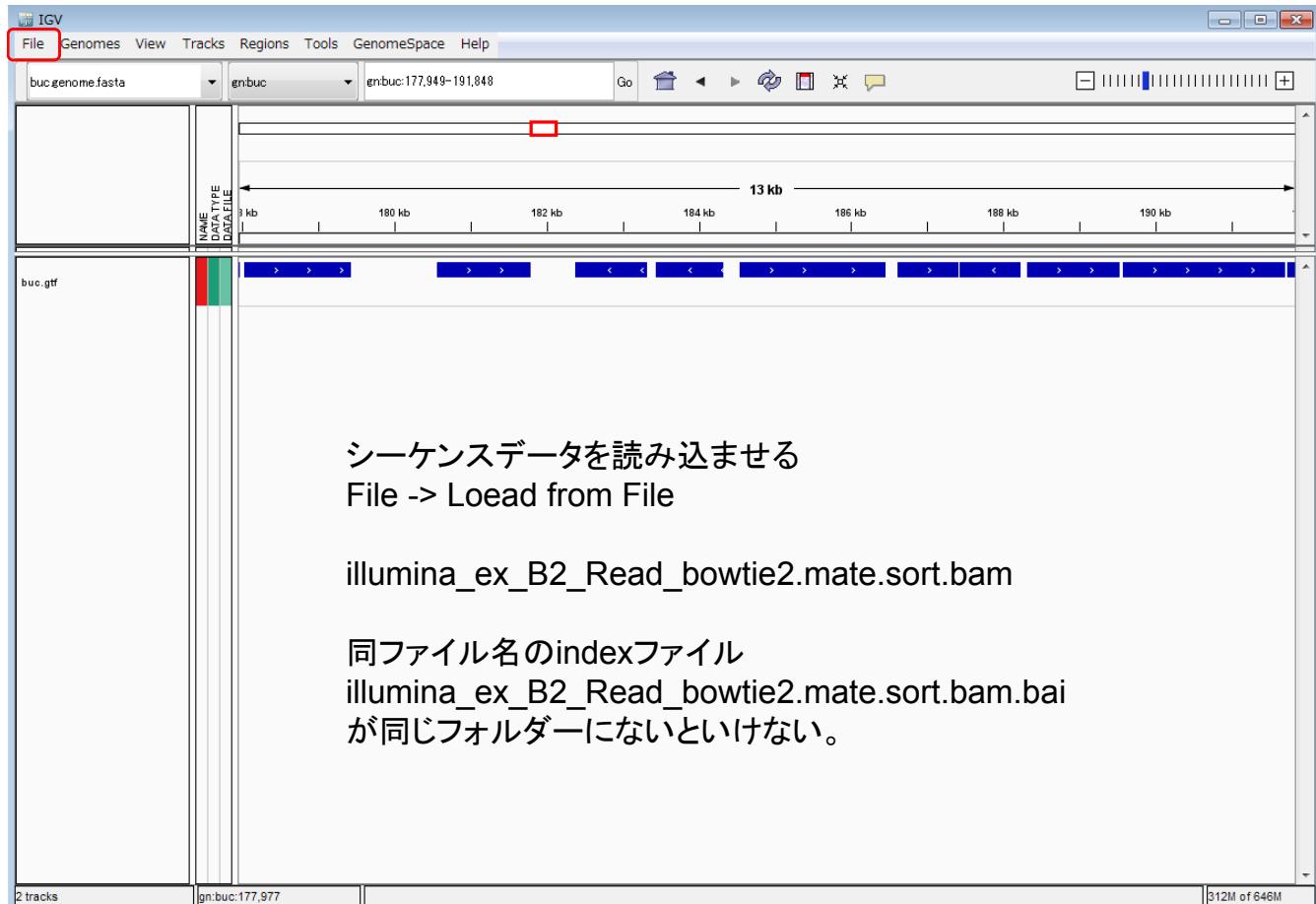
IGVの使用法を学ぶと共に  
先のファイルフォーマットも  
確認しよう

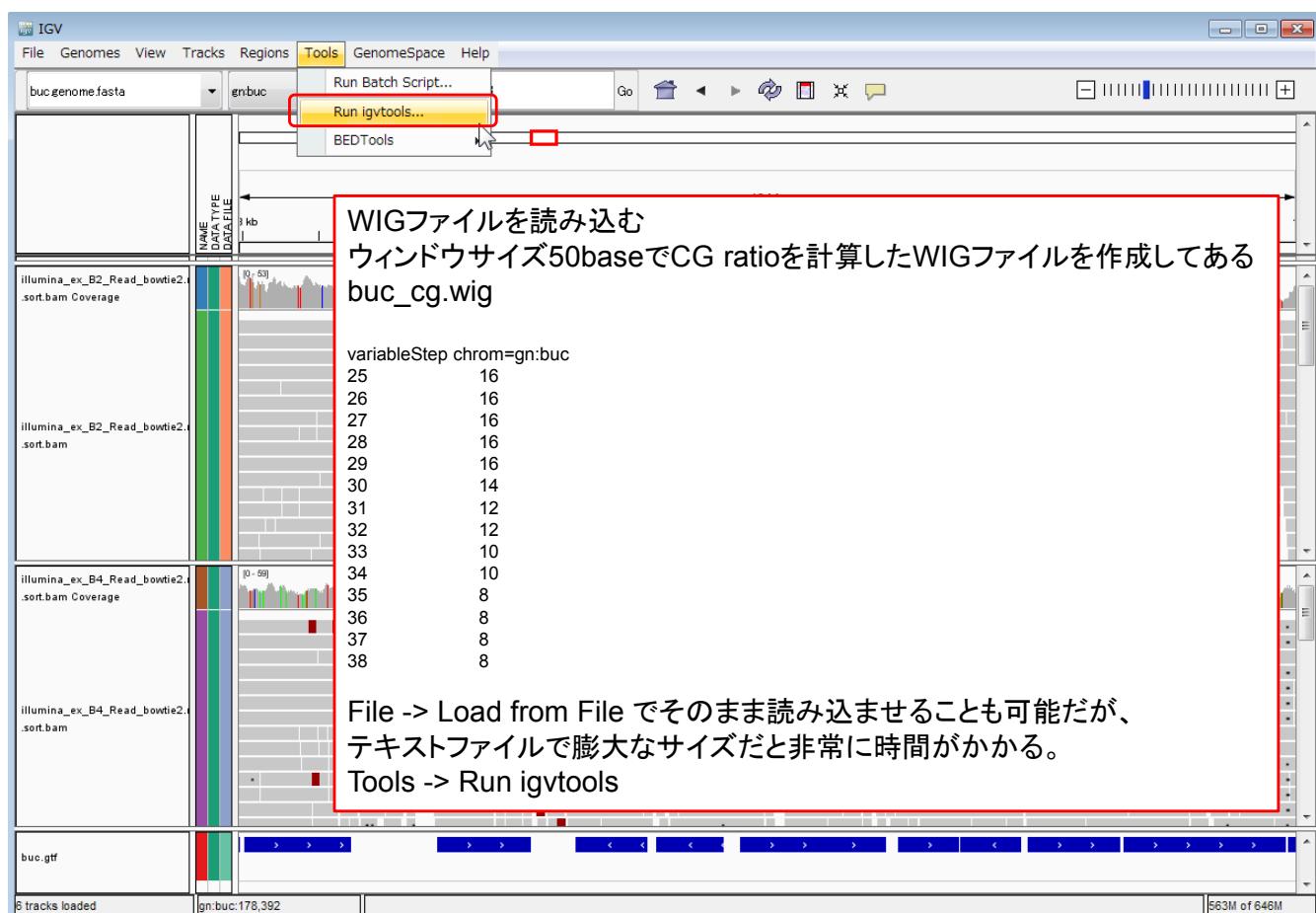
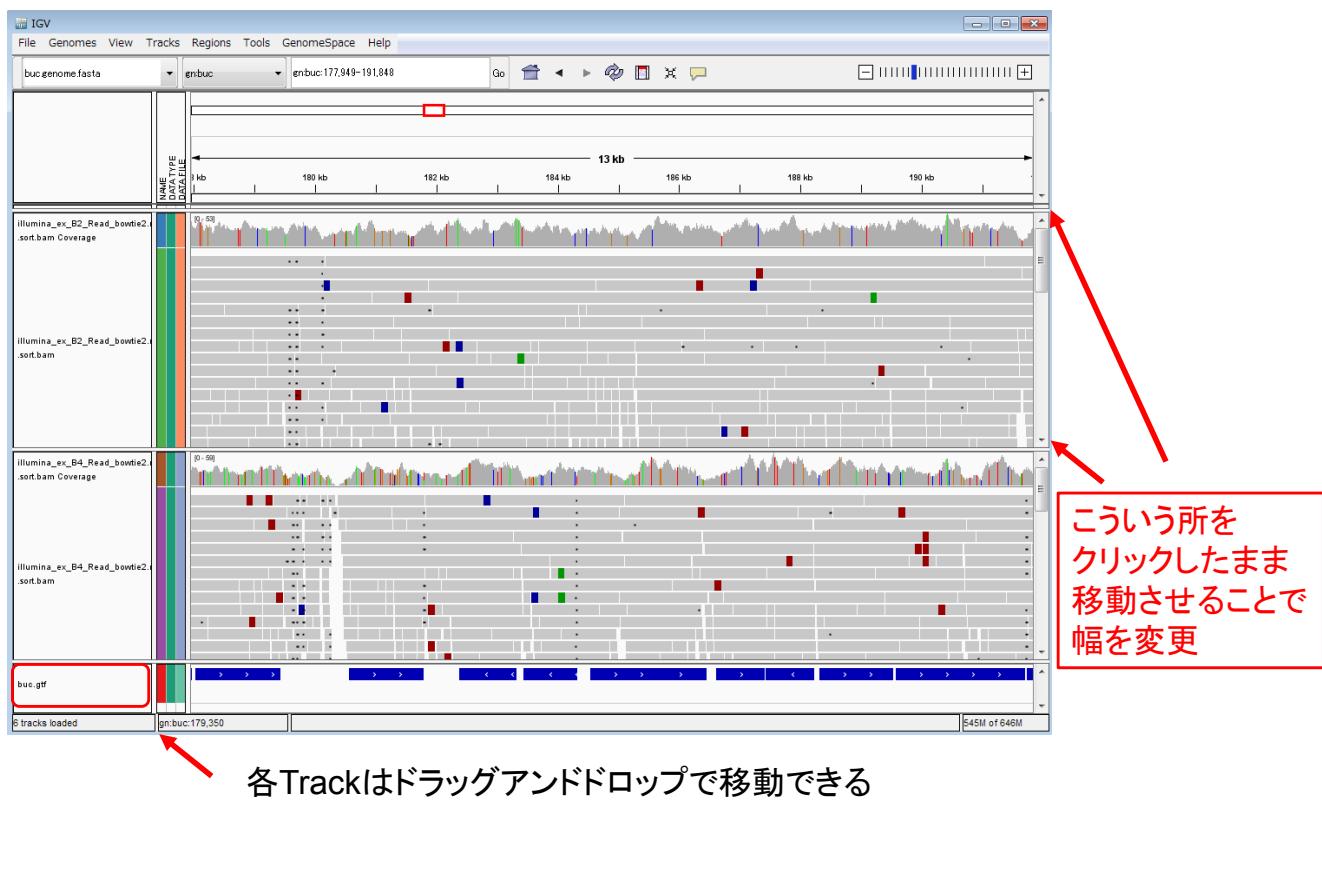
以下のファイルを確認

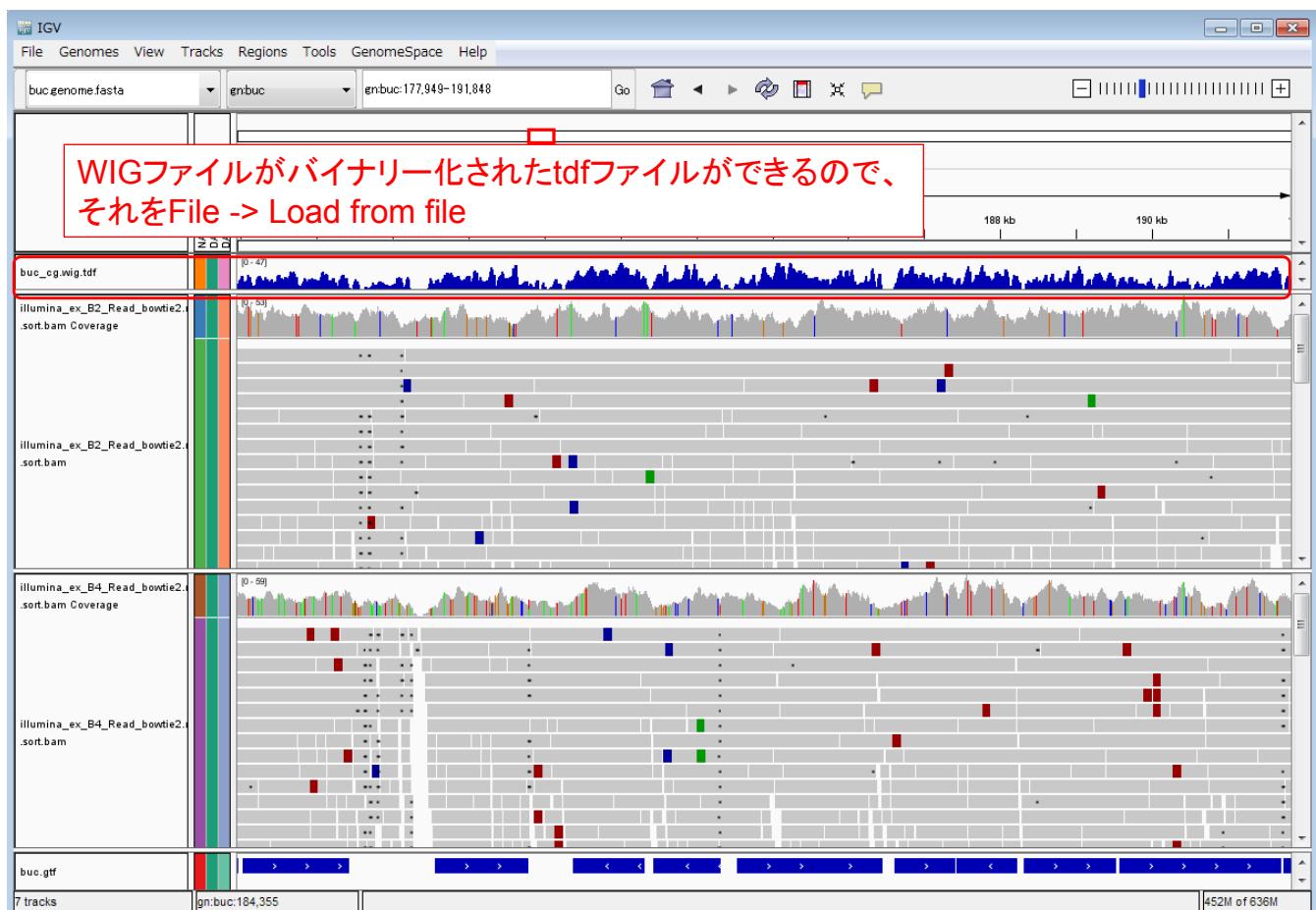
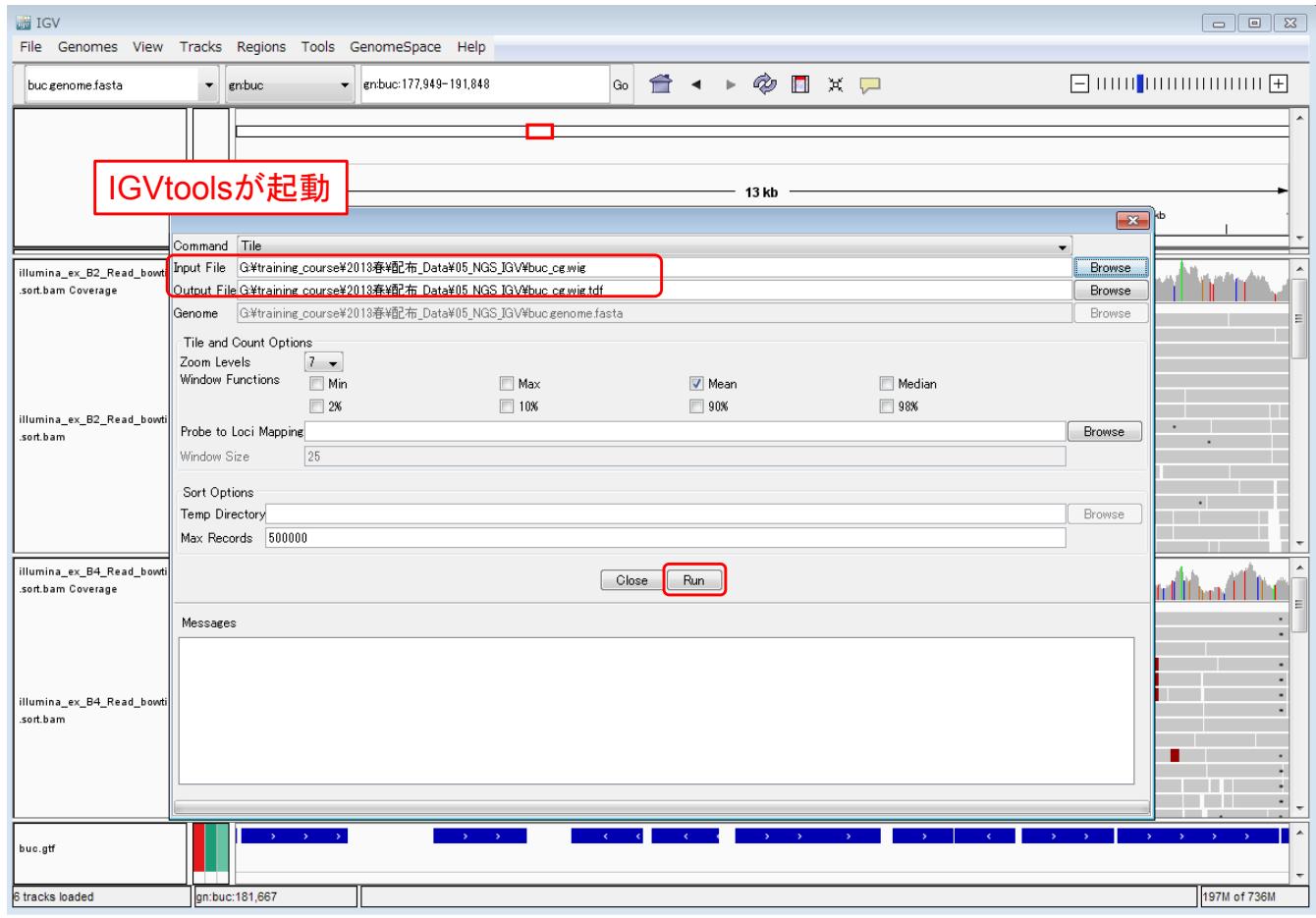
buc.genome.fasta  
 buc.gtf  
 buc\_cg.wig  
 illumina\_ex\_B2\_Read\_bowtie2.mate.sort.bam  
 illumina\_ex\_B2\_Read\_bowtie2.mate.sort.bam.bai  
 illumina\_ex\_B4\_Read\_bowtie2.mate.sort.bam  
 illumina\_ex\_B4\_Read\_bowtie2.mate.sort.bam.bai

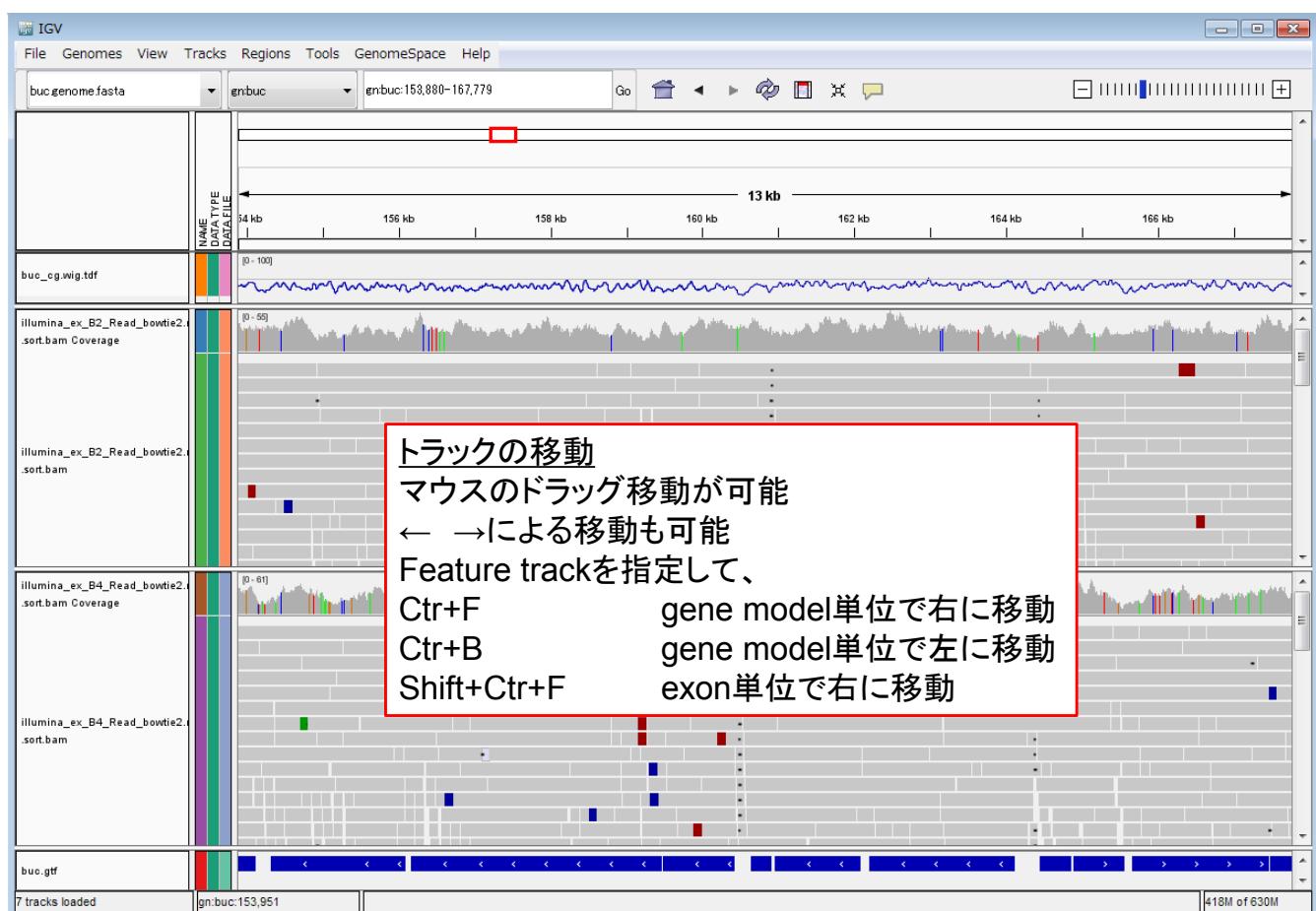
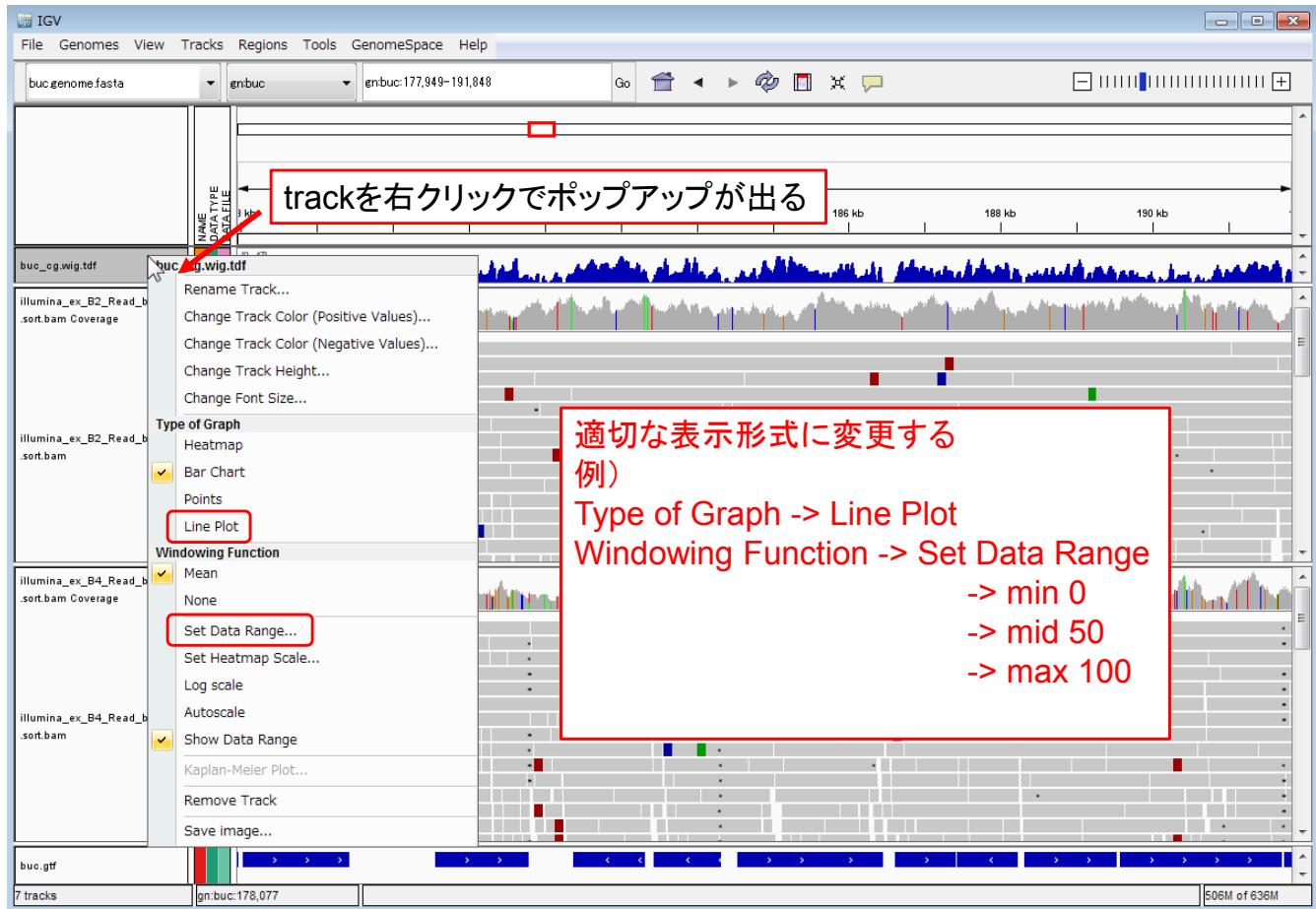


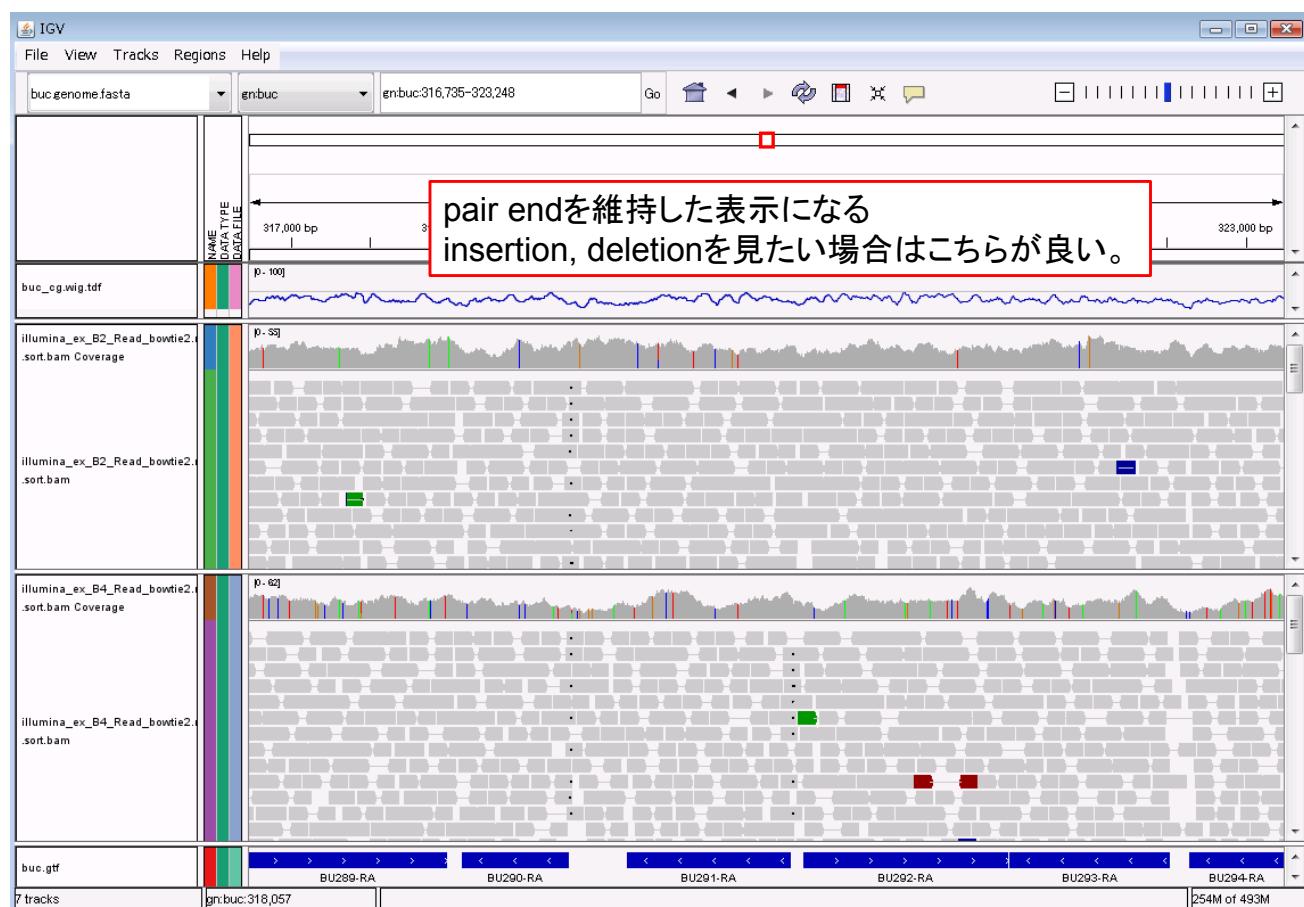
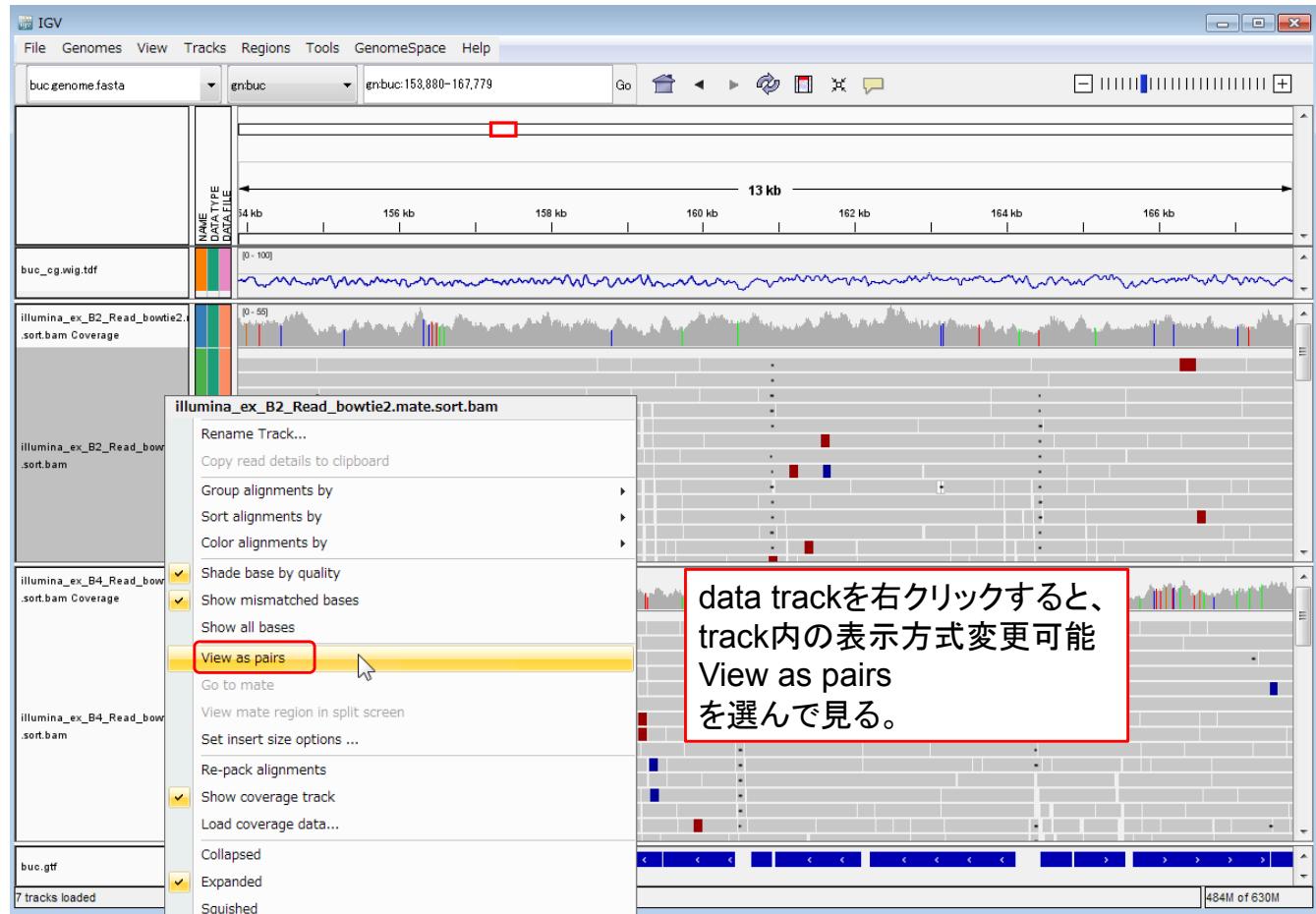


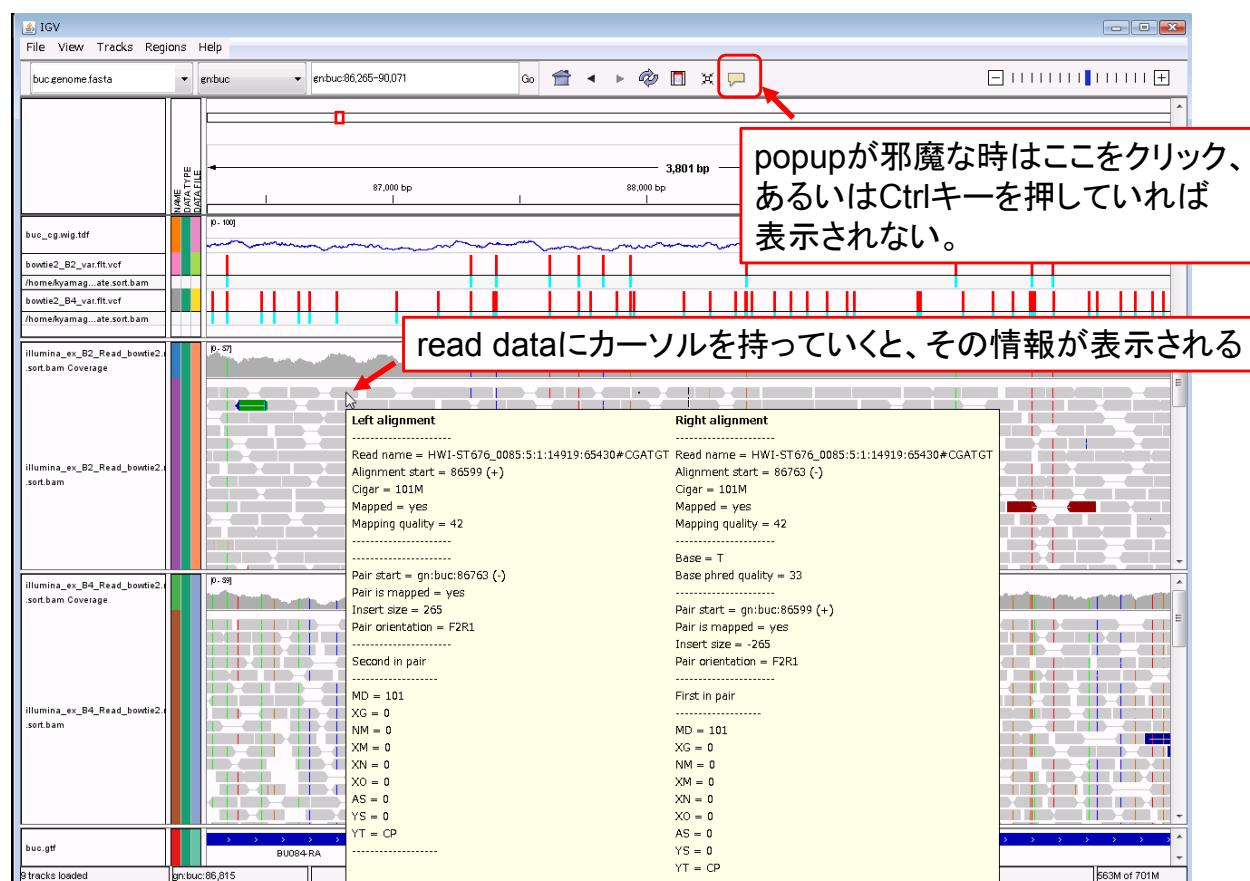












# IGV紹介のまとめ

可視化ツールとして十分な機能を持つ

- ・無料
- ・比較的簡単・お手軽
- ・自分で見るためにも良し、人に見せるためにも良し
- ・利用範囲は次世代DNAシーケンサーに限定しない  
広くゲノミクスの解析に有用

ごく一部のみの機能を紹介しました。  
ウェブサイトを見ながら復習をお勧めします。

# 統計学入門

北海道大学 農学研究院  
佐藤昌直

これらをこれから学習していくためには

- 汎用される統計の仕組みを知る
- 測定、実験計画を見直す
- 教科書を読めるように統計用語・表記に慣れる
- 道具を準備する - R

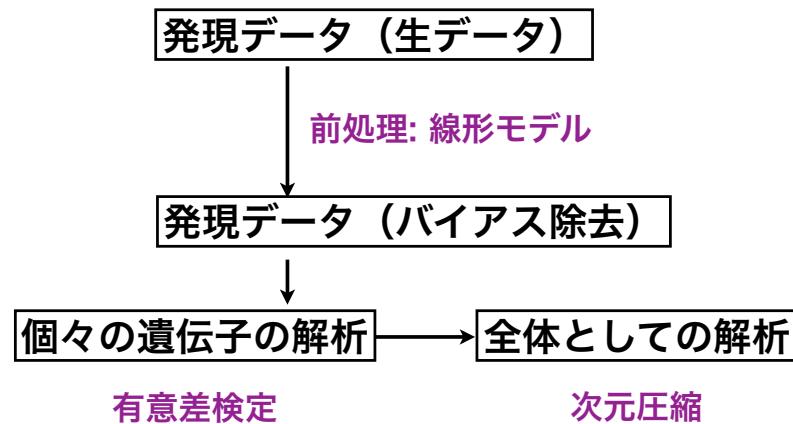
## 私が重視しているポイント

- 研究全体における統計の役割、実験と統計との連携を意識する
- 遺伝子発現解析に必要な統計の基礎概念を解説する
- “*statistical mind*”を養う

## 基本的な統計の用途

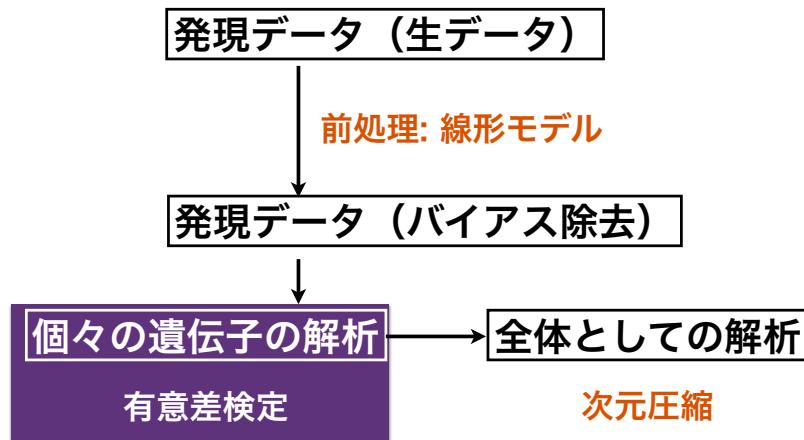
- 仮説検定
- 予測 (モデル構築)

## 遺伝子発現解析における統計の役割



## 仮説検定 - $t$ 検定を例に

## 遺伝子発現解析における統計の役割



ねらい

### $t$ 検定から検定の背景知識を得る:

- 検定の流れを知る
- 勉強のとっかかりを作る

### 用語の意味の整理

- 統計量、確率分布、自由度、 $p$ 値

## 統計における検定の手続き

1. 仮説を立てる
2. 統計量を求める
3. 求めた統計量を確率分布に照らし合わせる
4. 判定: 求めた確率と棄却限界値との比較

### 1. 仮説を立てる:

**帰無仮説**

*statistical  
mind*

**最終的に棄却される仮定:**

「AとBに差がある」かを検定する場合は  
「AとBには差がない」と仮定する

例1. 野生型と変異体Aの遺伝子xの発現量に違いがあるか？

例2. 野生型と変異体Aの遺伝子発現プロファイル間の相関係数は0.35だった。これらは有意に相関していると考えられるか？

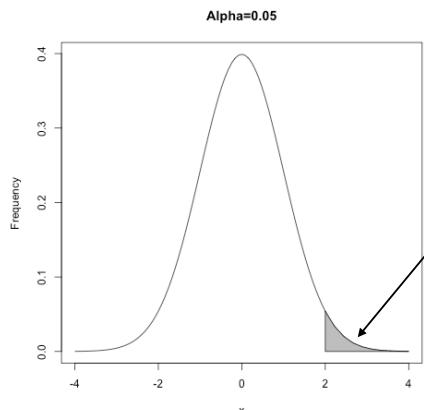
### 2. 統計量を求める:

**統計量:** データから導いた  
具体的な数値

↔ **母数:** 未知の数値

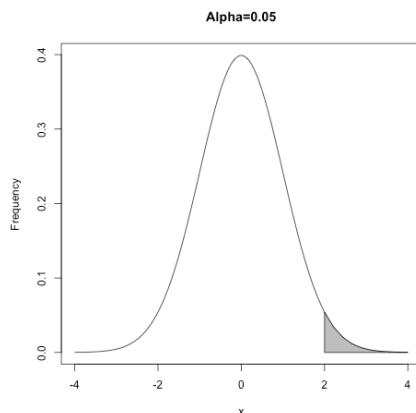
我々ができること: 少数の測定値（標本）から  
「母集団」を推定すること

### 3. 確率分布と照らし合わせる



棄却限界値によって  
規定される面積  
(通例: 全体の5%)

#### 4. 判定: 帰無仮説が棄却されるか?



最終的に棄却される仮定:

「AとBに差がある」かを検定する場合は「AとBには差がない」という仮定

#### 2. 統計量を求める:

**統計量:** データから導いた  
具体的な数値

↔ **母数:** 未知の数値

我々ができること: 少数の測定値（標本）から  
「母集団」を推定すること

## 代表値

**平均値:** 相加平均。すべてのデータを足して、データ数で割って得られる値

- (バー) は  
平均を表す  
^ (ハット) は  
推定を表す

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

**中央値:** データを小さいものから順に並べたときに中央にくる値

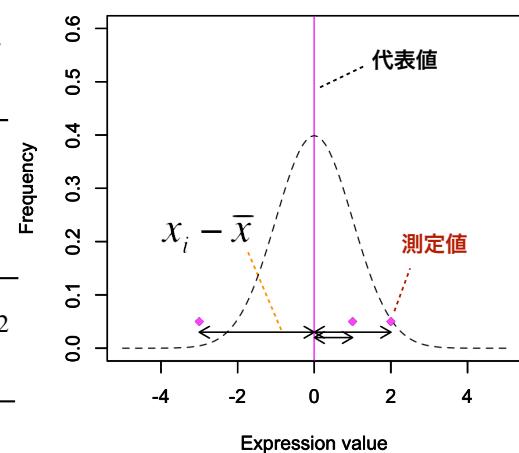
## ばらつき: 分散／偏差

**分散:**

$$\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$$

**標準偏差:**

$$\sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$$



**n-1?**

なぜ、平均を求める時と分散を求める時では分母が変わるので？

自由度: 統計量を求めるのに使うことができる「独立」な標本数

## 統計的検定の手続き

1. 仮説を立てる

2つのサンプル間で遺伝子発現量  
(平均値) の違いがある？

2. 統計量を求める

平均、標準誤差、自由度から  
t統計量を求める

3. 求めた統計量を確率  
分布に照らし合わせる

t分布からp値を求める

4. 判定: 求めた確率と  
棄却限界値との比較

有意差の判定

## t検定

## 母集団を推定する統計量

(標本)平均:

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

- (バー) は  
平均を表す

$$\text{標準偏差: } \hat{\sigma} = s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$$

^ (ハット) は  
推定を表す

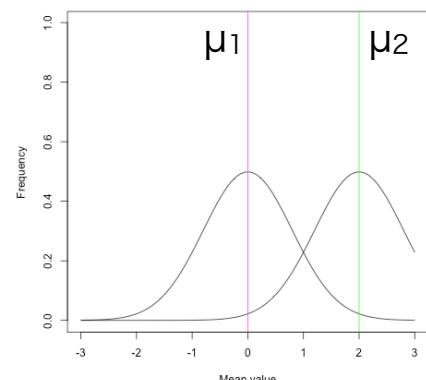
**t検定:**

ポイント

## 2サンプルの平均の検定

- 平均値 =  $\mu_1, \mu_2$
- データは正規分布

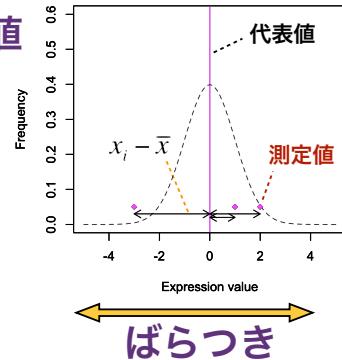
ほぼ全ての検定方法に  
前提がある



# 母集団を推定する統計量

## 1. (真の値に近い)代表値

## 2. ばらつきの範囲



# 統計量その1

**平均値:** 相加平均。すべてのデータを足して、データ数で割って得られる値

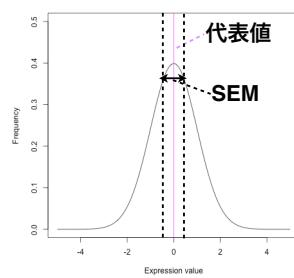
$$\bar{x} = \hat{x} = \frac{\sum_{i=1}^n x_i}{n}$$

# 統計量その2: *statistical mind*

## 平均値もあくまで推定値

(平均) 標準誤差:  
「統計量」の偏差

$$SEM = \frac{s}{\sqrt{n}}$$

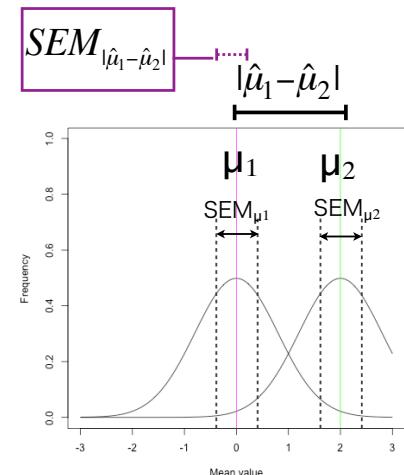


# 統計量その3: *statistical mind*

## 平均の差とその誤差

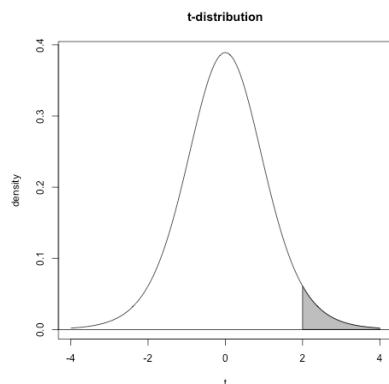
### t統計量

$$t = \frac{|\hat{\mu}_1 - \hat{\mu}_2|}{SEM_{|\hat{\mu}_1 - \hat{\mu}_2|}}$$



# 確率分布-t分布

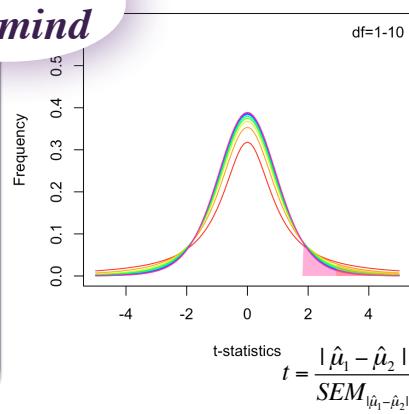
- 得られたt統計量がどのくらいの確率で起きうるか
- t分布（確率分布）を標本のt統計量と自由度を使って参照



【おさらい】自由度: 統計量を求めるのに使うことができる独立な標本数

## データの分布、仮説検定に即した確率分布を使う

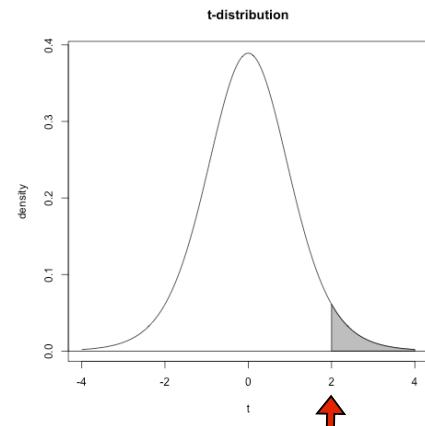
- 我々の測定では
- 母分散が**未知**
  - したがって確率密度は**自由度**によって変化
- 正規分布ではなく、t分布



例) 3つの観察で得られた平均値と100観察から得られた平均値はどちらが確からしいか

p値とは：

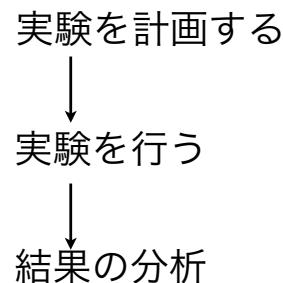
- 標本に基づいた統計量が帰無仮説の下、起きうる確率
- 多くの場合、0.05が危険率



## 統計における検定の手続き

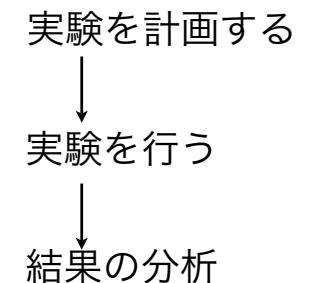
- 仮説を立てる
- 統計量を求める
- 求めた統計量を確率分布に照らし合わせる
- 判定: 求めた確率が棄却限界値より大きいか、小さいか

## 研究の手順（危険な例）



統計解析の結果によって  
結論を導く

**現実には：実験デザインはデータを取得する「前」に練ってある必要がある**



統計解析の結果によって  
結論を導く

ほぼ全ての検定方法に  
前提がある

## 多重検定の補正

+ 統計検定における重要な思考

*p*値とは：

- 標本に基づいた統計量が帰無仮説の下、起きうる確率
- 多くの場合、**0.05**が危険率

*p*値とは：

- ・ 標本に基づいた統計量が帰無仮説の下、起きうる確率
- ・ 多くの場合、**0.05**が危険率  
**= 100回に5回起きる**

多重検定の補正

- ・  $p = 0.05$ の検定を100回\*繰り返すと、  
**5回はランダムに間違い**

\*NGS解析では数万回以上繰り返すことになります

多重検定の補正

1. Bonferroniタイプ

2. False discovery rate (FDR):

- Benjamini-Hochberg
- Storey

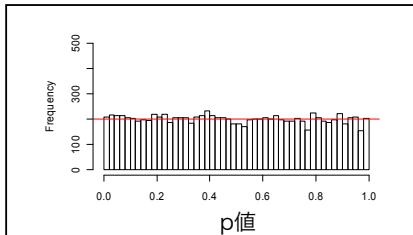
Bonferroniタイプの多重検定の補正

危険率を検定数で調整

$$\text{危険率} = \alpha/k$$

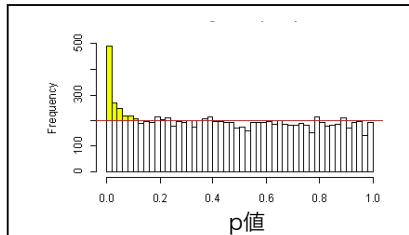
$\alpha$ : 元の危険率、  
k: 検定数

## False Discovery Rate (FDR)



帰無仮説

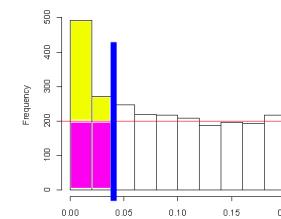
全ての範囲のp値が  
同等の頻度で観察される  
←どのp値を選んでも  
ランダムに選ぶのと同じ



観察

あるp値（閾値）以下のp  
値は有意な検定結果である  
→では、ランダムに生じて  
しまう各p値の頻度は？

## False Discovery Rate (FDR)



### q値:

補正されたp値。そのq値以  
下の検定のうち、どのくら  
いの割合でfalse positiveが  
含まれているか。

### ポイント

## $p$ 値、 $q$ 値の違い

$p$ 値の視点:  $\text{FP}/(\text{TN}+\text{FP})$

$q$ 値の視点:  $\text{TP}/(\text{TP}+\text{FP})$

### Statistical test

		positive	negative
Real	+	True positive	False negative
	-	False positive	True negative

## 復習／発展学習

- 検定の手順
  - 統計量
  - 確率分布
  - 自由度
  - $p$ 値
- 実験デザインの見直し、  
解析方法理解の基礎
- 統計解析の結果は確率に判断して得られたもの、  
トランскRIPTオーム解析ではそれを多数行う  
→ 多重検定の補正
- 検定方法、多重検定の補正における仮定  
例) 時系列データの比較にFDRは使えない

## データのばらつきと 実験デザイン・統計学的観点

### 我々の実験対象の例

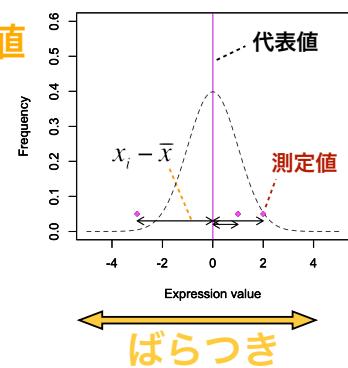
- ある遺伝子型の生物の
- ある環境での + 制御不能な実験要因
- ある遺伝子の発現量 + 生化学反応のノイズ

我々に実行できる事

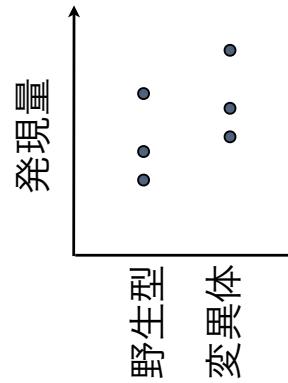
少数の測定値（標本）から  
「母集団」を推定すること

### 母集団を推定する統計量

1. (真の値に近い)代表値
2. ばらつきの範囲

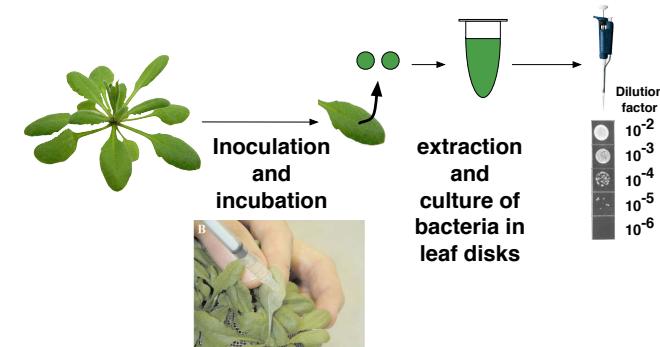


## 測定データはバラつく



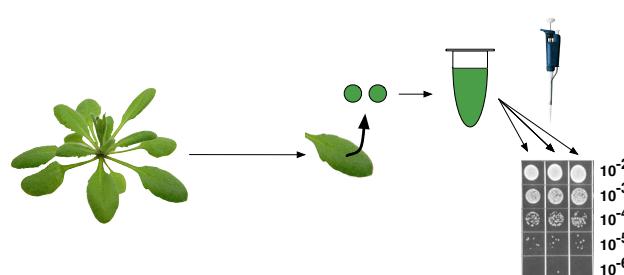
- 実験（測定）を反復する
- 何を「真」と考えるか
- 論文として発表できるデータには再現性が必要

## 例: バクテリア増殖定量

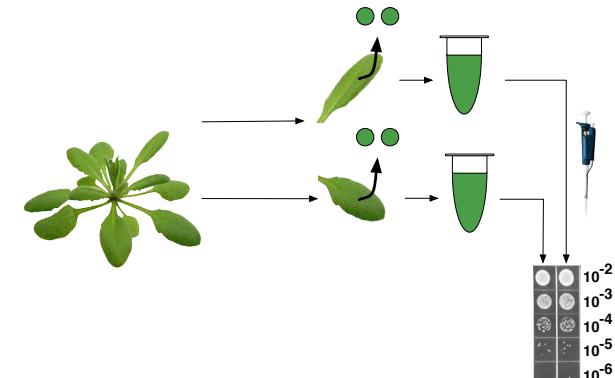


Katagiri, Thilmony R, and He S (2002) The *Arabidopsis Thaliana-Pseudomonas Syringae* Interaction. The *Arabidopsis* Book.

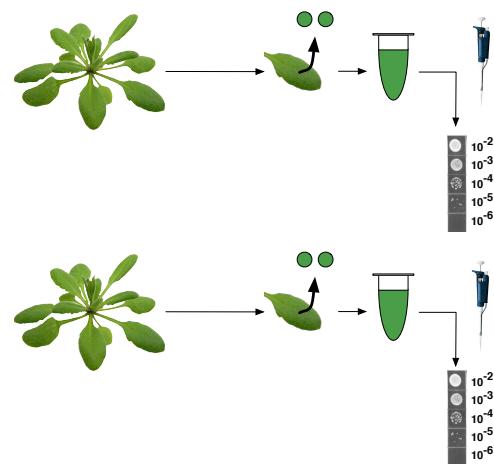
反復？



反復？



反復？



## 1データポイント: 潜在的に複数のばらつきを含む

- ・ 生物学的にはばらつきの中のある1点
- ・ 測定技術のばらつきの中のある1点

## 1データポイント: 潜在的に複数のばらつきを含む

- ・ 生物学的にはばらつきの中のある1点
- ・ 測定技術のばらつきの中のある1点

## 測定における2要因

- ・ Precision - 精度
- ・ Accuracy - 正確度

## 測定における2要因

- **Precision - 精度**

ある1測定を繰り返した際のばらつきの  
尺度

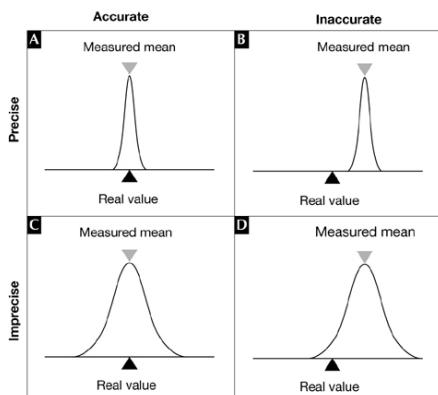
## 測定における2要因

- **Accuracy - 正確度**

ある測定値が「真の値」にどれだけ近い  
かの尺度

### ポイント

## 測定における2要因



ある測定を無限に繰り  
返した際の測定値ヒス  
トグラム

**Real value:** 真の値  
**Measured mean:**  
測定値から  
得られた平均

どちらが繰り返し測定する  
ことにより改善しうるか？

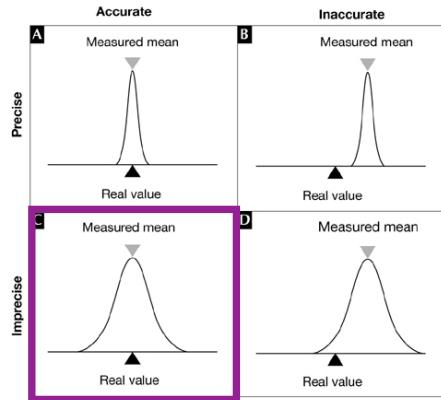
- **Precision - 精度**

ある1測定を繰り返した際の  
ばらつきの尺度

- **Accuracy - 正確度**

ある測定値が「真の値」に  
どれだけ近いかの尺度

## Technical replicates: 測定値の改善



Harm van Bakel & Frank C.P. Holstege (2004) *EMBO reports* (2004) 5, 964 - 969

我々にできる事

少数の測定値（標本）から  
「母集団」を推定すること

生体サンプルを繰り返し取る:  
biological replicates

同一サンプルを繰り返し測る:  
technical replicates

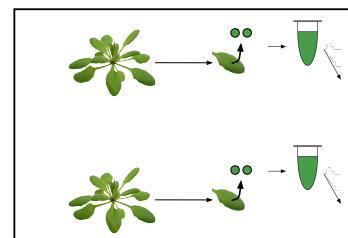
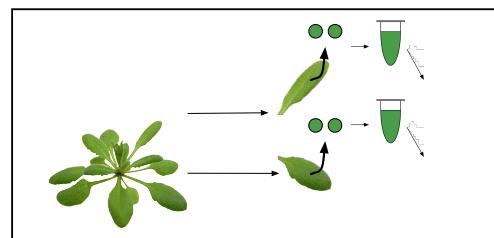
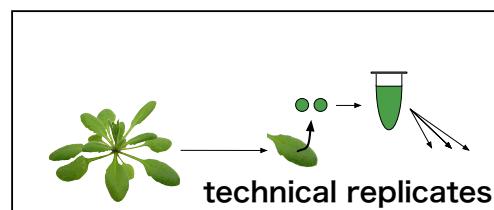
## 我々が1データポイントから 得ているもの

- ・ 生物学的にはらつきの中のある1点
- ・ 測定技術のはらつきの中のある1点

何を知るための実験か？

再現性のあるデータとは何か？

どのように反復を行うのが適切か？



## 定量的測定が可能且つ要求される時代の 再現性のあるデータとは何か？

- 何が再現されうるか？再現されたとするか？
- いつ行っても再現できる？
- どこで行っても再現できる？
- 誰が行っても再現できる？

## 定量的測定が可能且つ要求される時代の 再現性のあるデータとは何か？

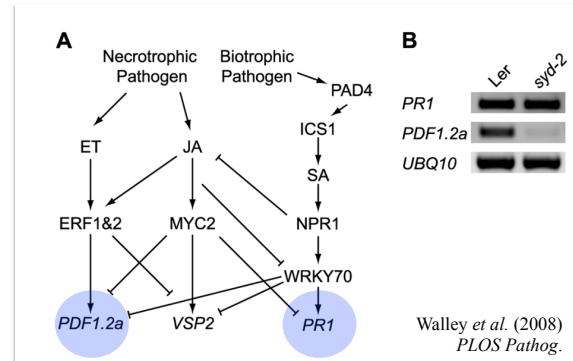
- 何が再現されうるか？再現されたとするか？

- いつ行っても再現できる？
- どこで行っても再現できる？
- 誰が行っても再現できる？

バラつきの  
定量と  
説明変数への  
割当て

## “マーカー遺伝子”測定

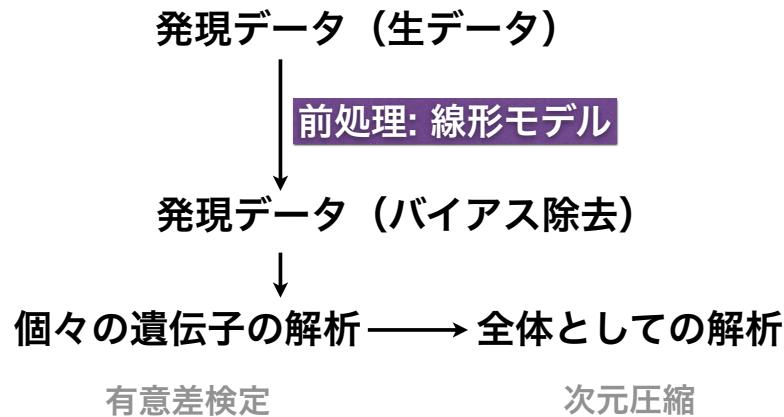
- 何が再現されうるか？再現されたとするか？



明瞭な違いを  
示す遺伝子:  
明瞭な再現性

分散分析・線形モデル:  
多変数データを系統立てて解析する  
- 実験デザインと統計の連携

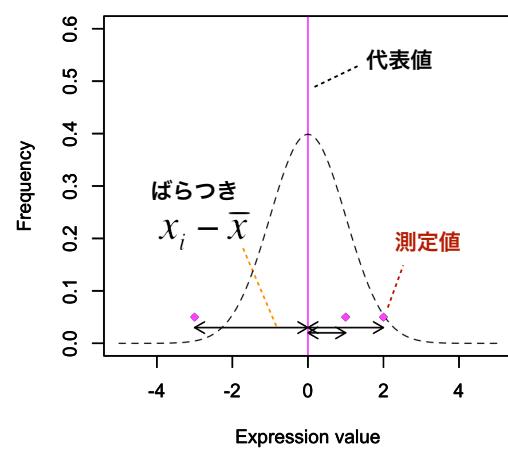
# 解析の流れ



# 目標

- 線形モデルの概念を掴む
- 実験デザインがどう統計に影響するかを考えるきっかけとする

リマインド:  
母集団を推定する統計量



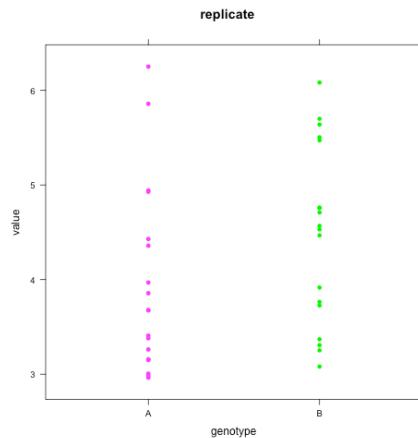
$t$ 検定: 平均値の検定

$$x_i = \bar{x} + (x_i - \bar{x})$$

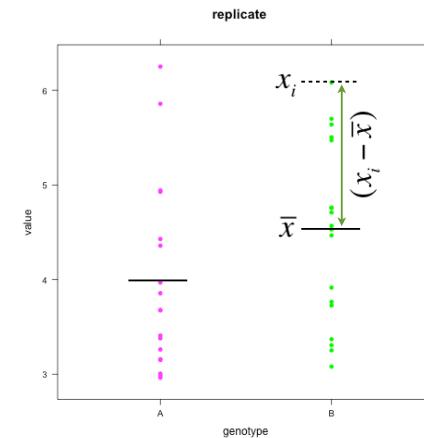
偏差: 平均値からのばらつき

## あるRT-qPCR実験

- genotype A, Bについて
- 6検体ずつ3回反復して計測



- genotype: A, B
- replicate: 1, 2, 3
- value:  
計18個/ genotype



$$x_i = \bar{x} + (x_i - \bar{x})$$

線形モデルの枠組みで考えてみる

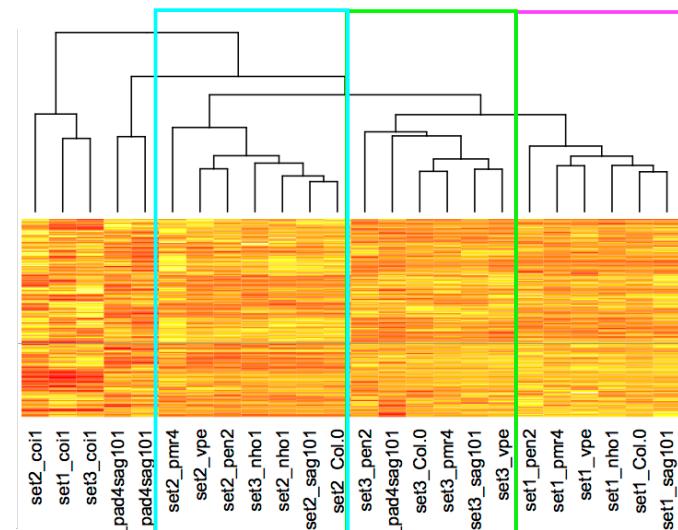
$$x_i = \bar{x} + (x_i - \bar{x})$$

$$x_i = \bar{x} + \varepsilon_i$$

残差 (観察値-推定値):  
想定要因では説明できない  
データの変動

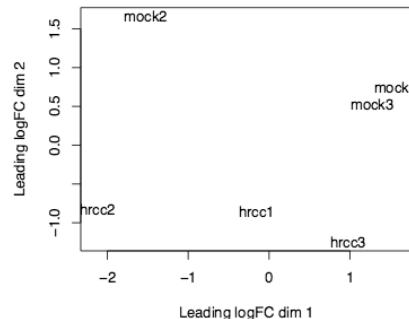
考慮するのは1要因で良いか?

ポイント



## “トランスクリプトーム”測定

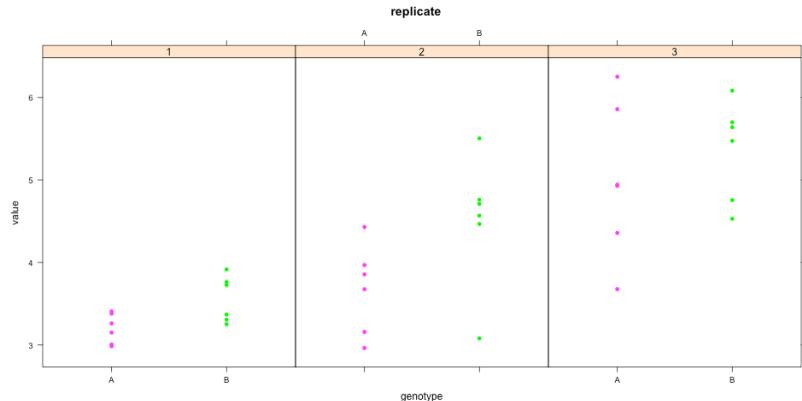
- 何が再現されうるか？再現されたとするか？



網羅的測定：  
再現性の  
再定義

Chen et al. (2015) edgeR User's Guide page 63

例：2遺伝子型の測定を3反復したデータ



観察値を複数要因の  
影響に起因するものとして分解

$$x_i = \bar{x} + (x_i - \bar{x})$$

$$x_i = \bar{x} + \varepsilon_i$$

*genotype*と*replicate*の  
影響を同時に  
考えられないか？

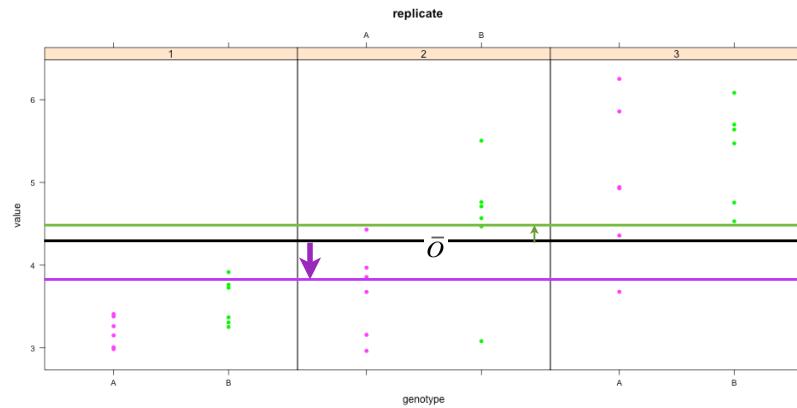
$$O_{ij} = x_i + y_j + \varepsilon_{ij}$$

線形モデルの仕組み

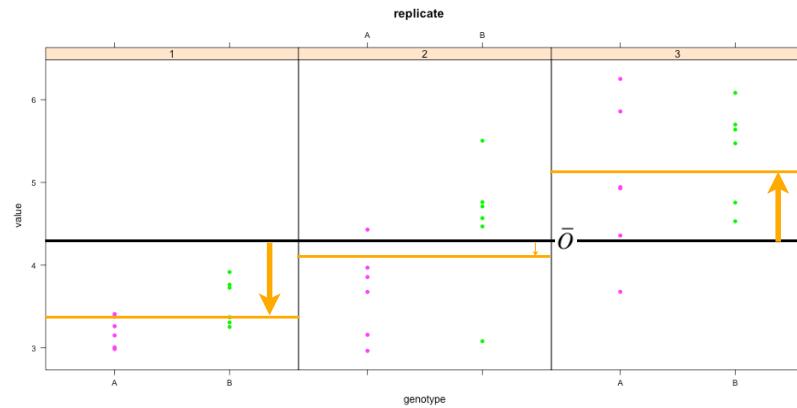
$$O_{ij} = x_i + y_j + \varepsilon_{ij}$$

$$O_{ij} = \bar{O} + (\bar{x}_{i\bullet} - \bar{O}) + (\bar{y}_{\bullet j} - \bar{O}) + \varepsilon_{ij}$$

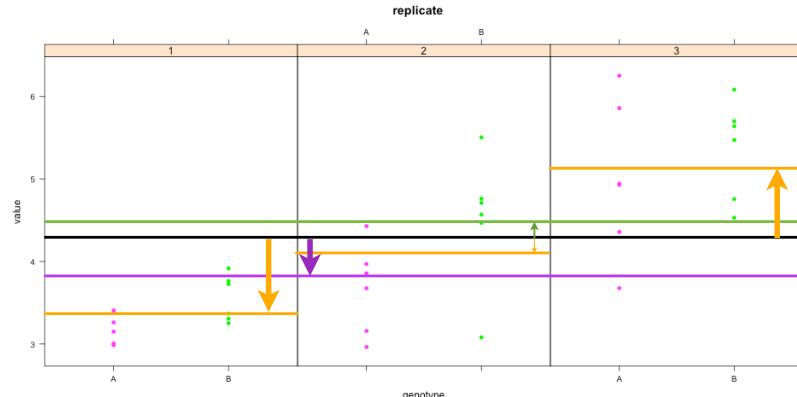
## $(\bar{x}_{i\bullet} - \bar{O})$ 遺伝子型による変動



## $(\bar{y}_{\bullet j} - \bar{O})$ 反復ごとの変動



各計測値は  $O_{ij} = \bar{O} + (\bar{x}_{i\bullet} - \bar{O}) + (\bar{y}_{\bullet j} - \bar{O}) + \varepsilon_{ij}$  と表せる



## 分散分析・線形モデルの枠組み

$$O_{ij} = x_i + y_j + \varepsilon_{ij}$$

$$O_{ij} = \bar{O} + (\bar{x}_{i\bullet} - \bar{O}) + (\bar{y}_{\bullet j} - \bar{O}) + \varepsilon_{ij}$$

↓ 教科書・論文での書き方

$$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

応答変数                  説明変数

# 線形モデルとは

応答変数  $\sim$  説明変数1 + 説明変数2 + ... + 誤差

と観察値を説明する（かもしれない）  
変数でそれらの関係性を書き下すこと

- 実際には: Rでlmなどの関数を使う

## 実験デザインの重要性

ポイント

- 線形モデルで推定・除去

$$O_{ij} = \alpha_i + \beta_j + \varepsilon_{ij}$$

$\alpha_i$ : 遺伝子型／処理など注目して  
いる効果の要因

$\beta_j$ : 反復（実験日時）／実験者  
などバイアス要因

- $\alpha_i$  の推定値、標準誤差のみを使う

## 実験デザインの重要性

- omicsデータは“batch effect”と呼ばれる  
体系的なバイアスが混入する。  
例: 実験時期、実験者、餌



Nature Reviews Genetics (2010) 11, 733-

- 線形モデルで推定・除去

## 定量的測定が可能かつ要求される時代の 再現性のあるデータとは何か？

- 何が再現されうるか？再現されたとするか？

- いつ行っても再現できる？
- どこで行っても再現できる？
- 誰が行っても再現できる？

系統的な  
バラつきの  
定量と  
説明変数への  
割当て

## R (edgeR) での実装

```
> x <- read.delim("TableOfCounts.txt",row.names="Symbol")
> group <- factor(c(1,1,2,2))
> y <- DGEList(counts=x,group=group)
> y <- calcNormFactors(y)
> design <- model.matrix(~group)
> y <- estimateDisp(y,design)
> fit <- glmFit(y,design)
> lrt <- glmLRT(fit,coef=2)
> topTags(lrt)
```

Chen, et al., edgeR User's Guide (December 26, 2017)

## ポイント

## model.matrixで生成される出力

```
group      <- factor(c(rep("M", 3), rep("H", 3)))
replicates <- factor(c(1:3, 1:3))
model.matrix(~group+replicates)

(Intercept) groupM replicates2 replicates3
1            1     1          0          0
2            1     1          1          0
3            1     1          0          1
4            1     0          0          0
5            1     0          1          0
6            1     0          0          1

attr(,"assign")
[1] 0 1 2 2
attr(,"contrasts")
attr(,"contrasts")$group
[1] "contr.treatment"
```

0と1の行列  
contrasts

## Rを用いた線形モデルにおける

### 実験デザイン指定: factor, model.matrix

```
> x <- read.delim("TableOfCounts.txt",row.names="Symbol")
> group <- factor(c(1,1,2,2))
> y <- DGEList(counts=x,group=group)
> y <- calcNormFactors(y)
> design <- model.matrix(~group)
> y <- estimateDisp(y,design)
> fit <- glmFit(y,design)
> lrt <- glmLRT(fit,coef=2)
> topTags(lrt)
```

Chen, et al., edgeR User's Guide (December 26, 2017)

## 分散分析・線形モデルの枠組み

$$O_{ij} = x_i + y_j + \varepsilon_{ij}$$

$$O_{ij} = \bar{O} + (\bar{x}_{i\bullet} - \bar{O}) + (\bar{y}_{\bullet j} - \bar{O}) + \varepsilon_{ij}$$

教科書・論文での書き方

$$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

応答変数 説明変数

## ポイント

## model.matrixで生成される出力

	$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$
	(Intercept) groupM replicates2 replicates3
1	1 1 0 0
2	1 1 1 0
3	1 1 0 1
4	1 0 0 0
5	1 0 1 0
6	1 0 0 1

## ポイント

## 線形モデルとmodel.matrixの関係

$$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

i, jを書き下すと

$$O_{M,1} \sim \mu + \alpha_M + \beta_1 + \varepsilon_{M,1}$$

$$O_{M,2} \sim \mu + \alpha_M + \beta_2 + \varepsilon_{M,2}$$

$$O_{M,3} \sim \mu + \alpha_M + \beta_3 + \varepsilon_{M,3}$$

$$O_{H,1} \sim \mu + \alpha_H + \beta_1 + \varepsilon_{H,1}$$

$$O_{H,2} \sim \mu + \alpha_H + \beta_2 + \varepsilon_{H,2}$$

$$O_{H,3} \sim \mu + \alpha_H + \beta_3 + \varepsilon_{H,3}$$

## ポイント

## 線形モデルとmodel.matrixの関係

$$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

$$O_{M,1} \sim \mu + \alpha_M + \beta_1 + \varepsilon_{M,1}$$

$$O_{M,1} \sim \mu + \alpha_M + \alpha_H + \beta_1 + \beta_2 + \beta_3 + \varepsilon_{M,1}$$

$$O_{M,1} \sim \mu \times 1 + \alpha_M \times 1 + \alpha_H + \beta_1 + \beta_2 \times 0 + \beta_3 \times 0 + \varepsilon_{M,1}$$

contrasts: 1番目の水準の係数を0として残りと比較

$$O_{M,1} \sim \mu \times 1 + \alpha_M \times 1 + \beta_2 \times 0 + \beta_3 \times 0 + \varepsilon_{M,1}$$

1 1 0 0

## ポイント

## model.matrix, contrasts, 実験デザインの関係

観察数: 6

推定する係数の数: 6

$$O_{M,1} \sim \mu + \alpha_M + \beta_1 + \varepsilon_{M,1}$$

$\mu, \alpha_M, \alpha_H, \beta_1, \beta_2, \beta_3$

$$O_{M,2} \sim \mu + \alpha_M + \beta_2 + \varepsilon_{M,2}$$

推定したい係数の数よりも  
観察数が多くなくてはならない

$$O_{M,3} \sim \mu + \alpha_M + \beta_3 + \varepsilon_{M,3}$$

contrasts: 1番目の水準の  
係数を0として残りと比較  
→係数の数を削減

$$O_{H,1} \sim \mu + \alpha_H + \beta_1 + \varepsilon_{H,1}$$

$$O_{H,2} \sim \mu + \alpha_H + \beta_2 + \varepsilon_{H,2}$$

$$O_{H,3} \sim \mu + \alpha_H + \beta_3 + \varepsilon_{H,3}$$

## ポイント

### model.matrixまとめ

$$O_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

(Intercept) groupM replicates2 replicates3

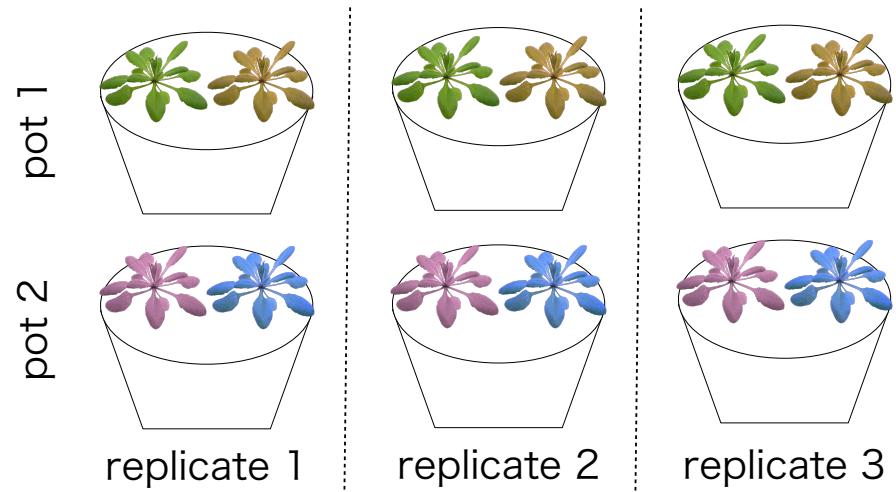
	1	1	0	0
1	1	1	1	0
2	1	1	0	1
3	1	0	0	0
4	1	0	1	0
5	1	0	0	1
6	1	0	0	1

#### ポイント

1. 0と1の意味
2. (この場合の) contrastsの概念:  $\mu$ =replicate1の処理Hの係数
3. 観察数、実験デザインとの関連

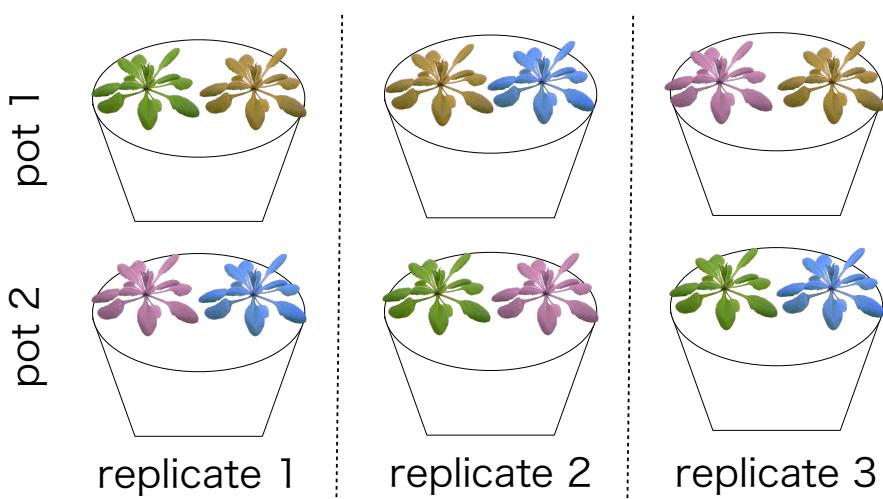
## 実験デザインの重要性:

genotype+replicate+potモデルを当てはめるには?



## 実験デザインの重要性:

genotype+replicate+potモデルを当てはめるには?



## ポイント

### 実験デザインの重要性

#### • 要因効果を推定するための実験デザイン

- 各実験要因を適切に反復させた実験デザイン  
(発展学習: 無作為化)

#### • 実験デザインとモデル

- 要因: データ取得「前」に想定しておくもの
- データの変動を説明しない要因を解析時に減らすことは可能。一方、実験デザイン時に計画しなかった要因を増せない。

# まとめ

- 計測データセットに影響を与える要因が一つではない場合、分散分析・線形モデルの枠組みが有効
- 理論を理解するのは難しいかもしれないが、実行はRで簡単に行える。理解に努める努力と実験デザインと連動したモデルを立てることが重要

## 復習／発展学習

- 回帰（最小二乗法）
- 実験計画法
- 交互作用
- Bioconductor: limma、edgeRパッケージ

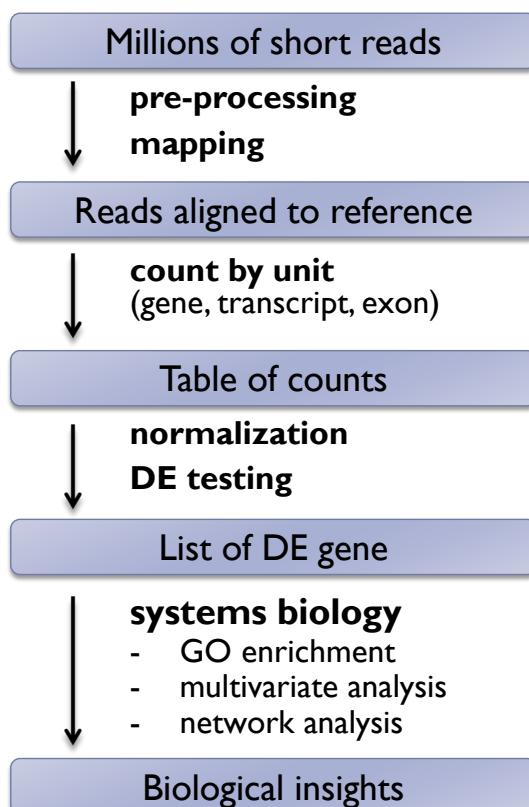
# RNA-seqの解析パイプライン：基礎

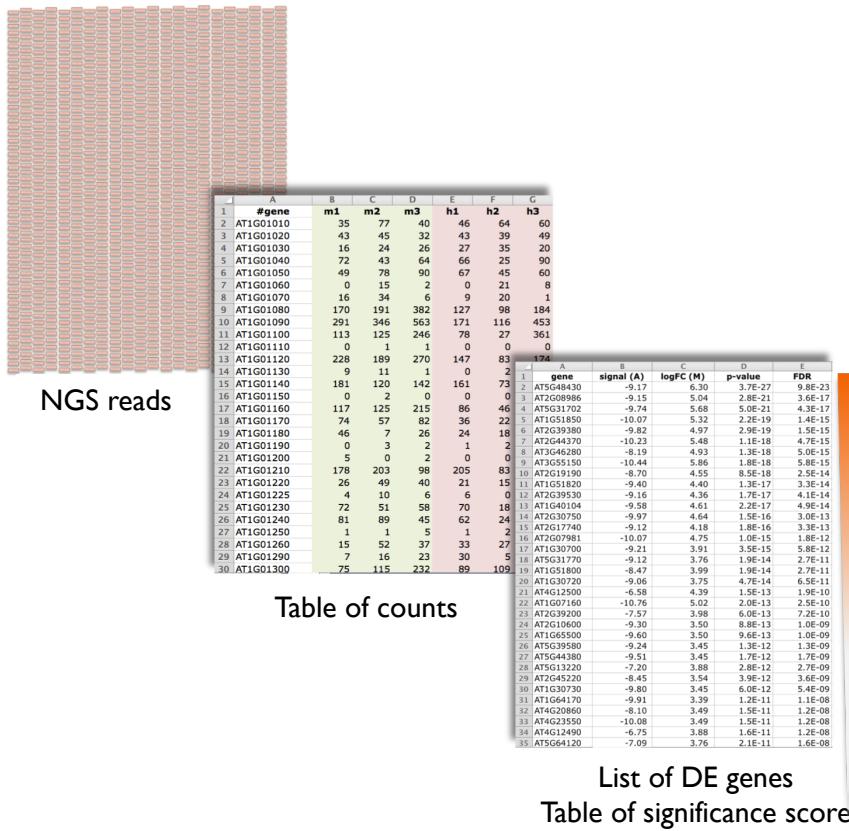
## RNA-seq Analysis Pipeline: basics

Shuji Shigenobu  
NIBB, Japan  
<shige@nibb.ac.jp>

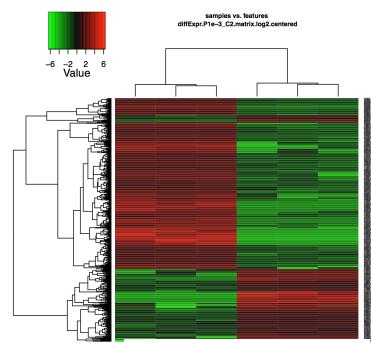
## RNA-seq analysis pipeline for DE

Differential Expression analysis



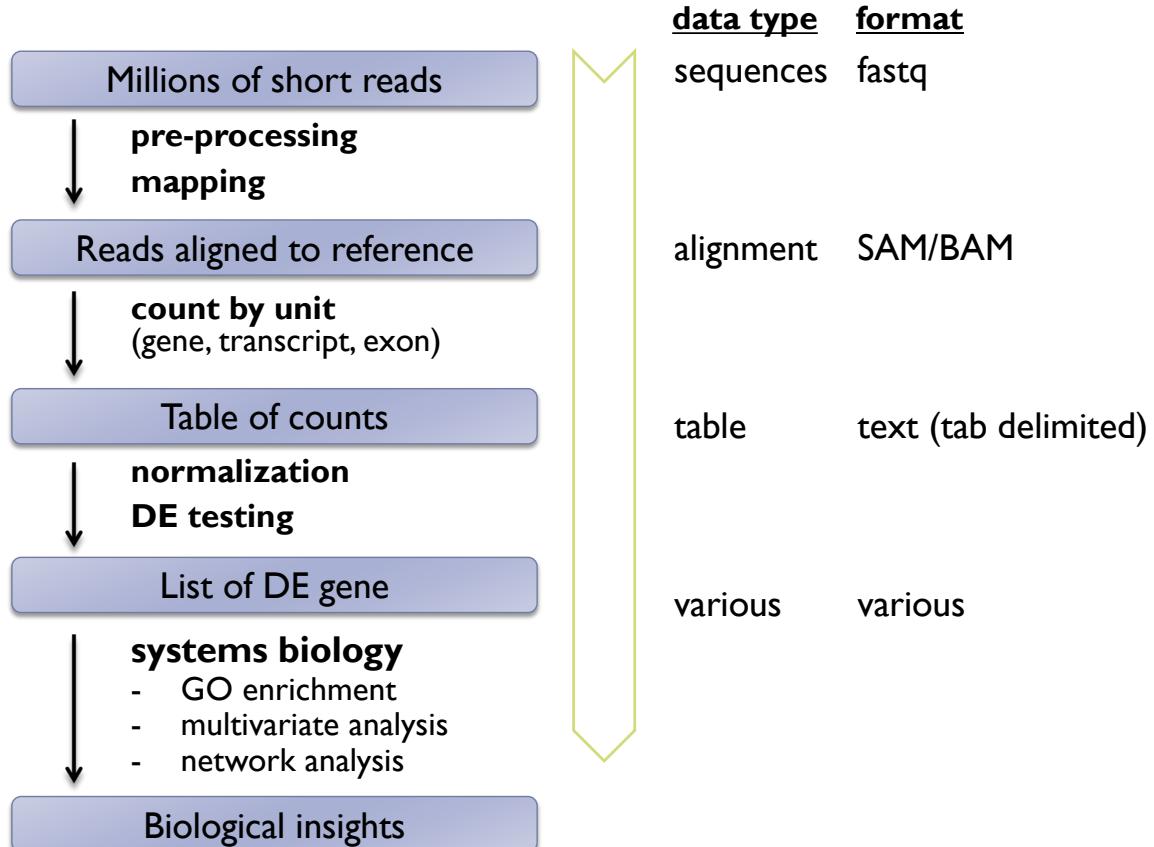


Biological insights



Systems biology:  
clustering and  
network analysis

## RNA-seq analysis pipeline for DE

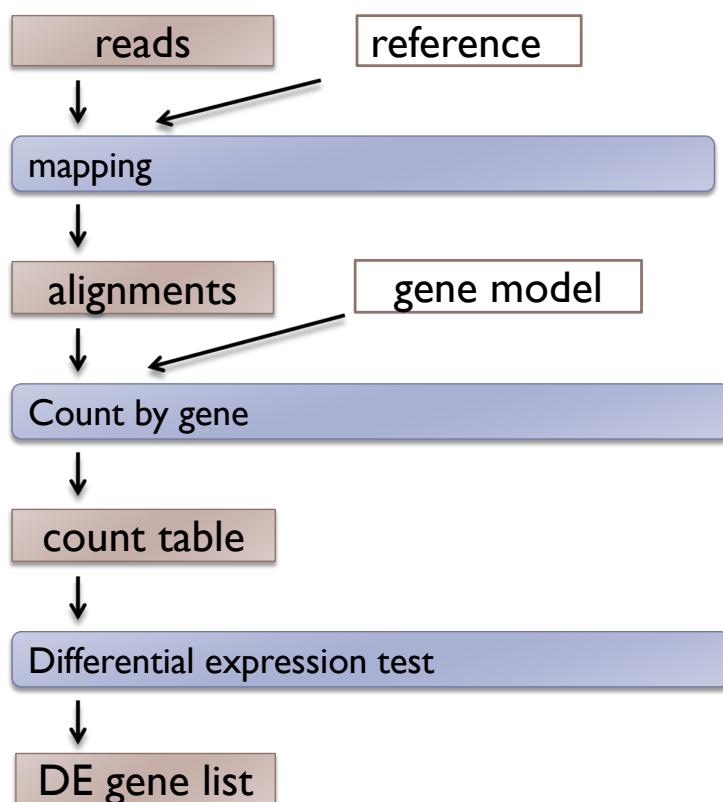


## Two Basic Pipelines

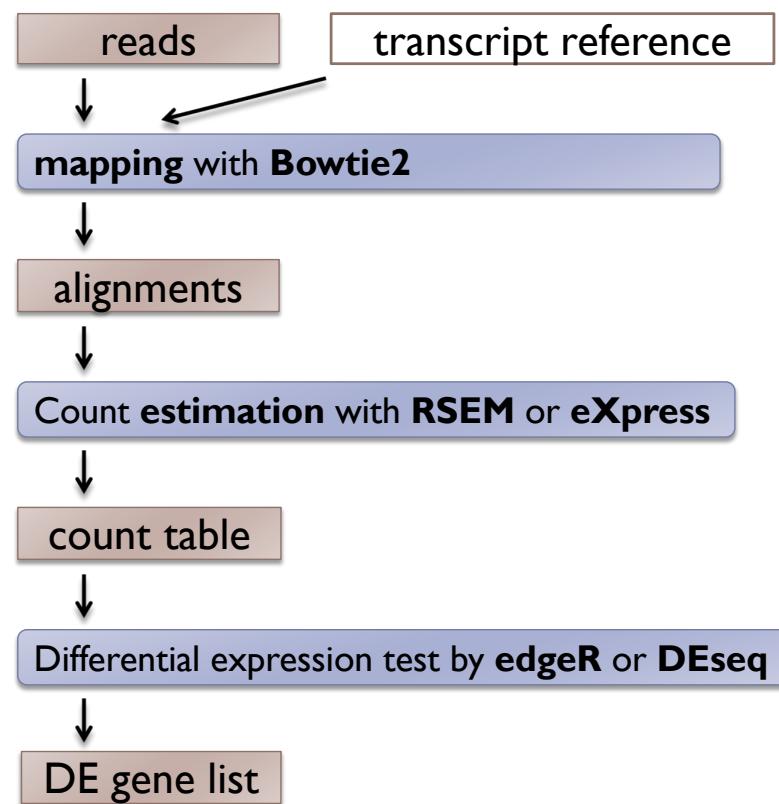
### ▶ Choice of reference

- ▶ **Genome** – standard for genome-known species
- ▶ **Transcript** – the only way for genome-unknown species
  - can be used for genome-known species

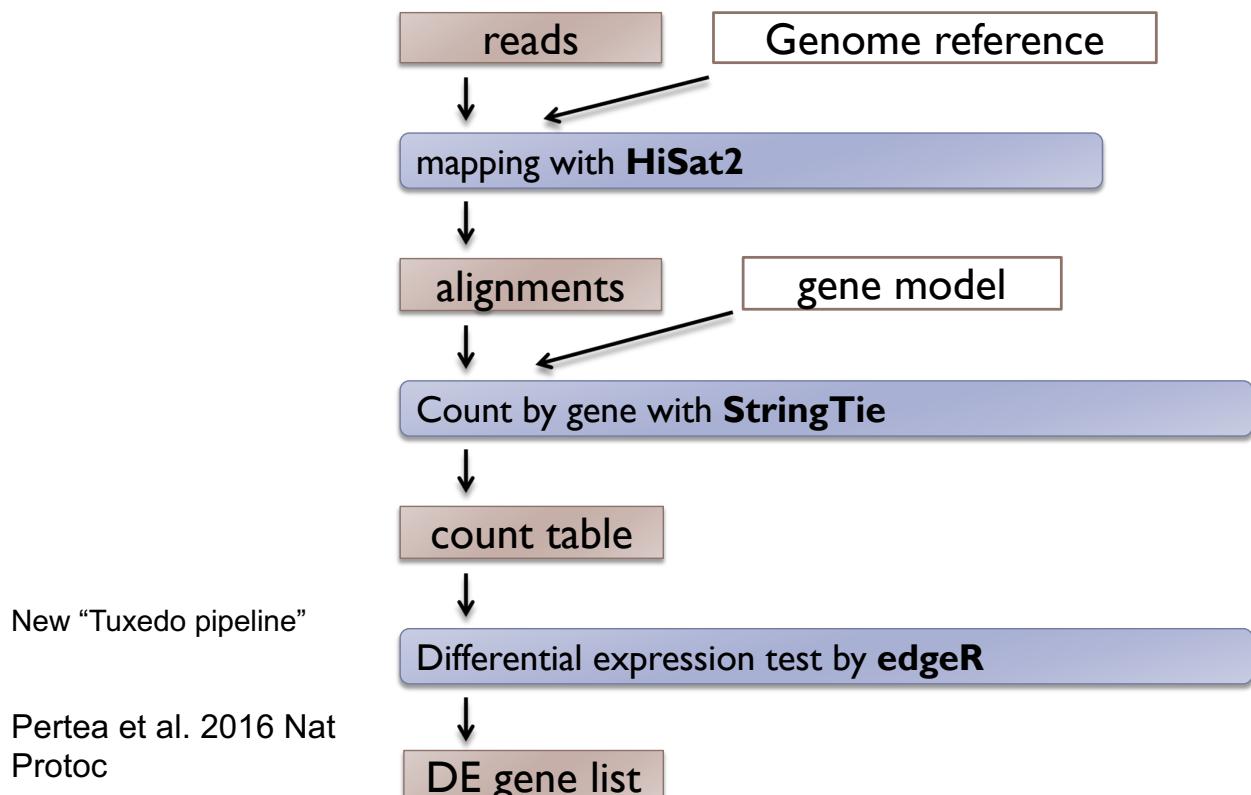
## Common workflow



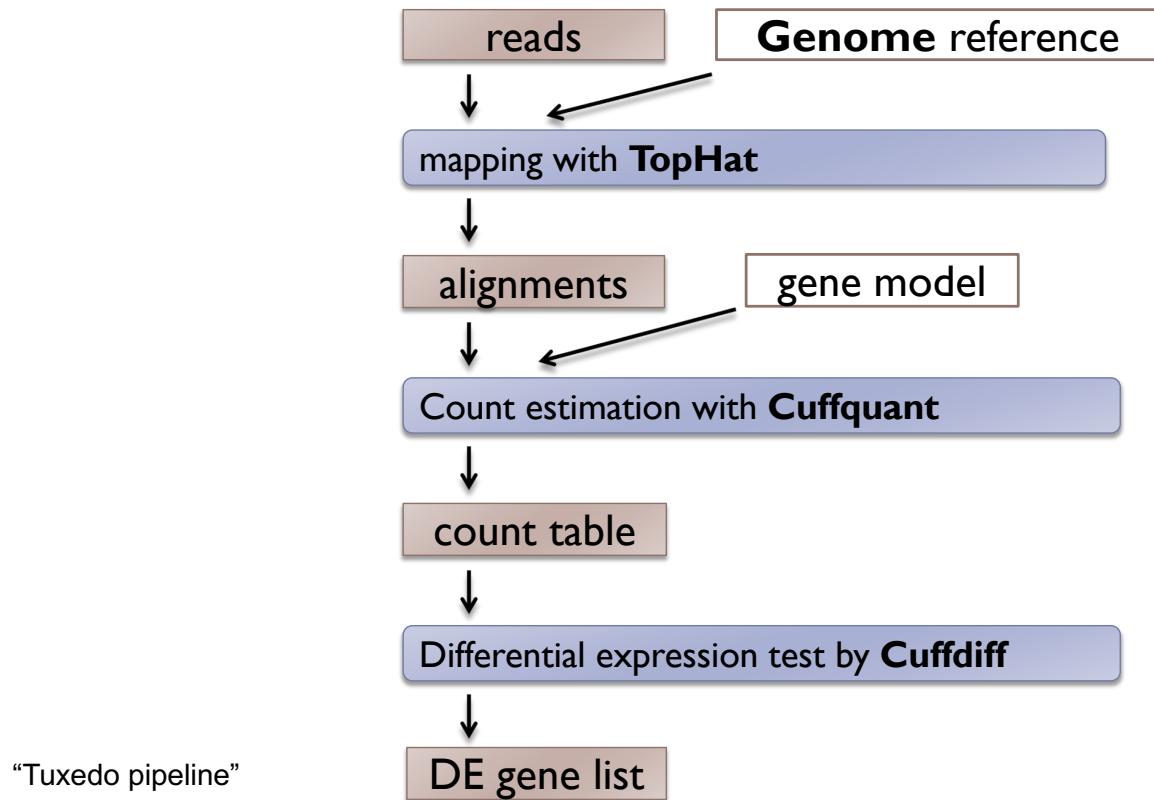
## A transcript-based pipeline



## A genome-based pipeline

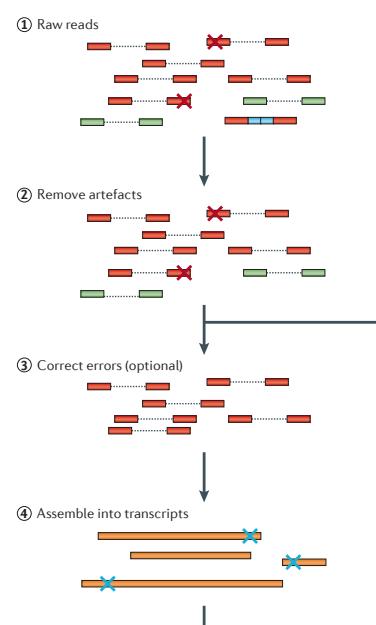


# A genome-based pipeline (old)



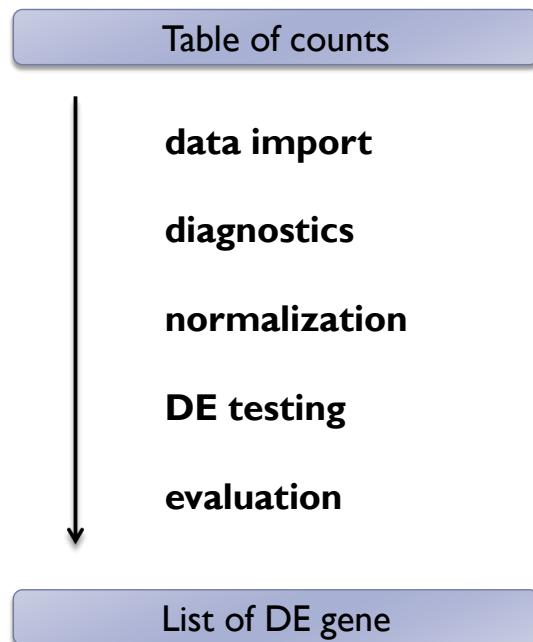
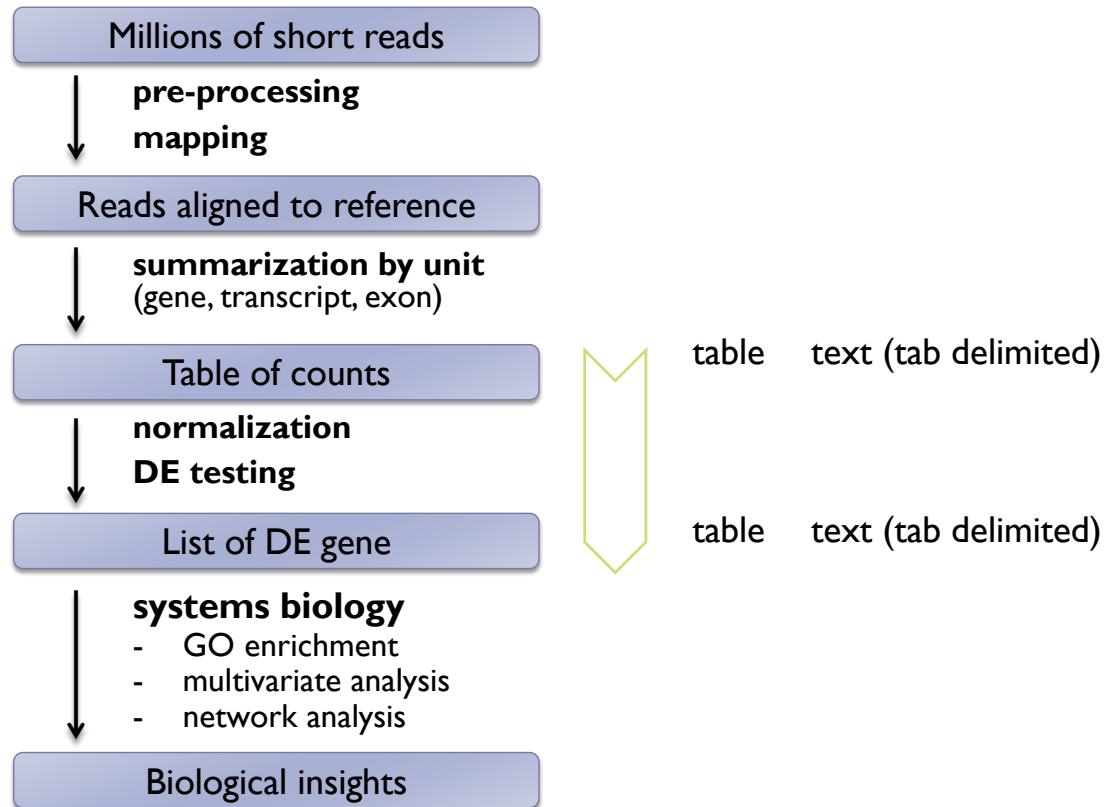
## Read QC and Pre-processing

- ▶ **Read QC**
  - ▶ Tools: FastQC etc.
- ▶ **Pre-processing**
  - ▶ Filter or trim by base quality
  - ▶ Remove artifacts
    - ▶ adaptors
    - ▶ low complexity reads
    - ▶ PCR duplications (optional)
  - ▶ Remove rRNA and other contaminations (optional)
  - ▶ Sequence error correction (optional)
  - ▶ Tools: cutadapt, trimmmatic



Martin et al (2011) Nat Rev Genet

# RNA-seq analysis pipeline for DE



## Input

	A	B	C	D	E	F	G
1	#gene	m1	m2	m3	h1	h2	h3
2	AT1G01010	35	77	40	46	64	60
3	AT1G01020	43	45	32	43	39	49
4	AT1G01030	16	24	26	27	35	20
5	AT1G01040	72	43	64	66	25	90
6	AT1G01050	49	78	90	67	45	60
7	AT1G01060	0	15	2	0	21	8
8	AT1G01070	16	34	6	9	20	1
9	AT1G01080	170	191	382	127	98	184
10	AT1G01090	291	346	563	171	116	453
11	AT1G01100	113	125	246	78	27	361
12	AT1G01110	0	1	1	0	0	0
13	AT1G01120	228	189	270	147	83	174
14	AT1G01130	9	11	1	0	2	9
15	AT1G01140	181	120	142	161	73	134
16	AT1G01150	0	2	0	0	0	0
17	AT1G01160	117	125	215	86	46	212
18	AT1G01170	74	57	82	36	22	29
19	AT1G01180	46	7	26	24	18	58
20	AT1G01190	0	3	2	1	2	2
21	AT1G01200	5	0	2	0	0	0
22	AT1G01210	178	203	98	205	83	143
23	AT1G01220	26	49	40	21	15	34
24	AT1G01225	4	10	6	6	0	3
25	AT1G01230	72	51	58	70	18	77
26	AT1G01240	81	89	45	62	24	33
27	AT1G01250	1	1	5	1	2	2
28	AT1G01260	15	52	37	33	27	54
29	AT1G01290	7	16	23	30	5	19
30	AT1G01300	75	115	232	89	109	224



## Output

	A	B	C	D	E
1	gene	signal (A)	logFC (M)	p-value	FDR
2	AT5G48430	-9.17	6.30	3.7E-27	9.8E-23
3	AT2G08986	-9.15	5.04	2.8E-21	3.6E-17
4	AT5G31702	-9.74	5.68	5.0E-21	4.3E-17
5	AT1G51850	-10.07	5.32	2.2E-19	1.4E-15
6	AT2G39380	-9.82	4.97	2.9E-19	1.5E-15
7	AT2G44370	-10.23	5.48	1.1E-18	4.7E-15
8	AT3G46280	-8.19	4.93	1.3E-18	5.0E-15
9	AT3G55150	-10.44	5.86	1.8E-18	5.8E-15
10	AT2G19190	-8.70	4.55	8.5E-18	2.5E-14
11	AT1G51820	-9.40	4.40	1.3E-17	3.3E-14
12	AT2G39530	-9.16	4.36	1.7E-17	4.1E-14
13	AT1G40104	-9.58	4.61	2.2E-17	4.9E-14
14	AT2G30750	-9.97	4.64	1.5E-16	3.0E-13
15	AT2G17740	-9.12	4.18	1.8E-16	3.3E-13
16	AT2G07981	-10.07	4.75	1.0E-15	1.8E-12
17	AT1G30700	-9.21	3.91	3.5E-15	5.8E-12
18	AT5G31770	-9.12	3.76	1.9E-14	2.7E-11
19	AT1G51800	-8.47	3.99	1.9E-14	2.7E-11
20	AT1G30720	-9.06	3.75	4.7E-14	6.5E-11
21	AT4G12500	-6.58	4.39	1.5E-13	1.9E-10
22	AT1G07160	-10.76	5.02	2.0E-13	2.5E-10
23	AT2G39200	-7.57	3.98	6.0E-13	7.2E-10
24	AT2G10600	-9.30	3.50	8.8E-13	1.0E-09
25	AT1G65500	-9.60	3.50	9.6E-13	1.0E-09
26	AT5G39580	-9.24	3.45	1.3E-12	1.3E-09
27	AT5G44380	-9.51	3.45	1.7E-12	1.7E-09
28	AT5G13220	-7.20	3.88	2.8E-12	2.7E-09
29	AT2G45220	-8.45	3.54	3.9E-12	3.6E-09
30	AT1G30730	-9.80	3.45	6.0E-12	5.4E-09
31	AT1G64170	-9.91	3.39	1.2E-11	1.1E-08
32	AT4G20860	-8.10	3.49	1.5E-11	1.2E-08
33	AT4G23550	-10.08	3.49	1.5E-11	1.2E-08
34	AT4G12490	-6.75	3.88	1.6E-11	1.2E-08
35	AT5G64120	-7.09	3.76	2.1E-11	1.6E-08



Table of counts

List of DE genes

Table of significance score

## Identify differentially expressed genes (DEG)

Question: Which are differentially expressed genes (DEG)?

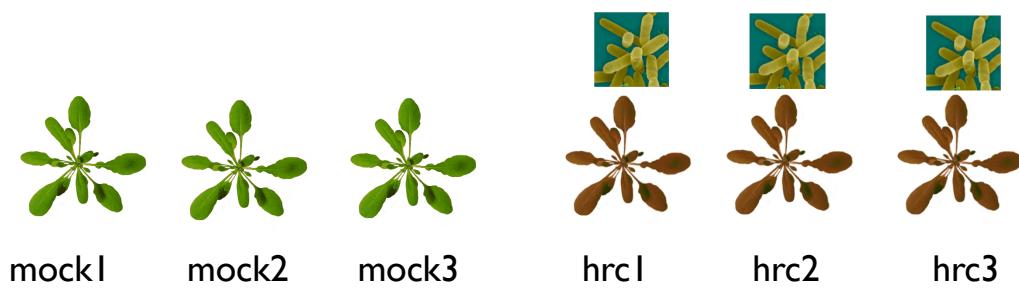
[simple examples (pairwise comparison)]

- mutant v.s. WT
- tissue A v.s. tissue B
- developmental time point A (ex. Early) v.s. B (ex. Late)

Goal:

- Find DE genes
- Rank by significance

# Example: Arabidopsis RNA-seq



mock inoculation (treated w/  
10mM MgCl<sub>2</sub>)

Challenged by defense-eliciting delta-hrcC mutant of *Pseudomonas syringae* pathovar *tmato* DC3000.

- 6 libraries = 2 groups x 3 biological replicates

Di, Y. et al. *Stat Appl Genet Mol* (2011).  
Cumbie, J. S. et al. *PLoS ONE* (2011).

## Input

- Typical primary data = matrix of #genes x #samples

column x number of samples (libraries)

row x number of genes (probes)

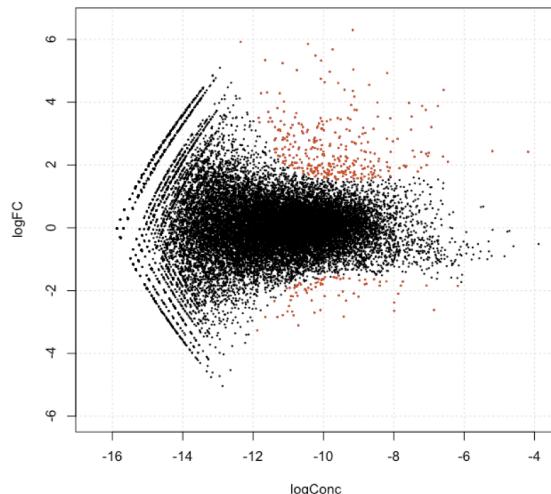
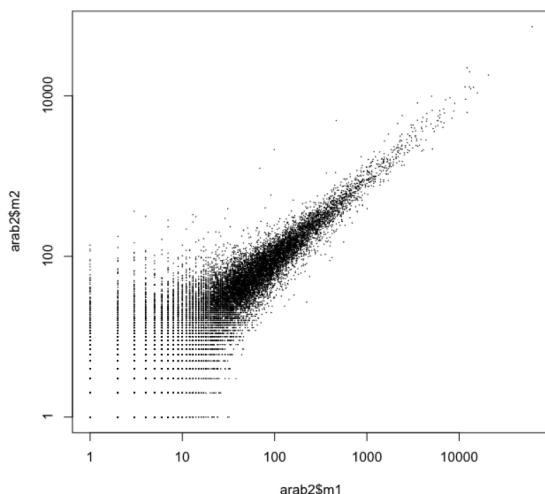
	A	B	C	D	E	F	G
1	#gene	m1	m2	m3	h1	h2	h3
2	AT1G01010	35	77	40	46	64	60
3	AT1G01020	43	45	32	43	39	49
4	AT1G01030	16	24	26	27	35	20
5	AT1G01040	72	43	64	66	25	90
6	AT1G01050	49	78	90	67	45	60
7	AT1G01060	0	15	2	0	21	8
8	AT1G01070	16	34	6	9	20	1
9	AT1G01080	170	191	382	127	98	184
10	AT1G01090	291	346	563	171	116	453
11	AT1G01100	113	125	246	78	27	361
12	AT1G01110	0	1	1	0	0	0
13	AT1G01120	228	189	270	147	83	174
14	AT1G01130	9	11	1	0	2	9
15	AT1G01140	181	120	142	161	73	134
16	AT1G01150	0	2	0	0	0	0
17	AT1G01160	117	125	215	86	46	212
18	AT1G01170	74	57	82	36	22	29
19	AT1G01180	46	7	26	24	18	58
20	AT1G01190	0	3	2	1	2	2
21	AT1G01200	5	0	2	0	0	0
22	AT1G01210	178	203	98	205	83	143
23	AT1G01220	26	49	40	21	15	34
24	AT1G01225	4	10	6	6	0	3
25	AT1G01230	72	51	58	70	18	77
26	AT1G01240	81	89	45	62	24	33
27	AT1G01250	1	1	5	1	2	2
28	AT1G01260	15	52	37	33	27	54
29	AT1G01290	7	16	23	30	5	19
30	AT1G01300	75	115	232	89	109	224

# Import count table / diagnostics

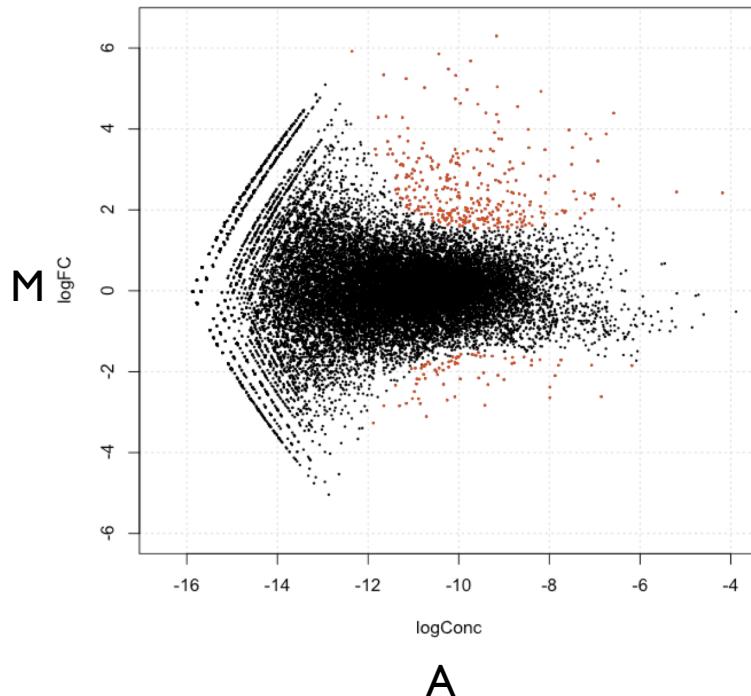
Look into the input data first.

- ▶ Quick view of the table (tools: R, MS Excel etc.)
  - ▶ Check: Format, data structure, data size etc.
- ▶ Scatter plot, MA plot (tools: R, MS Excel etc.)

## Diagnostics: Scatter plot & MA plot



# MA plot



**M:** log fold-change  
**A:** log intensity average

$$M = \log_2(R/G) = \log_2(R) - \log_2(G)$$
$$A = \frac{1}{2}\log_2(RG) = \frac{1}{2}(\log_2(R) + \log_2(G))$$

R: expression level of sample 1  
G: expression level of sample 2

## Let's try: data import and quick check

```
> dat <- read.delim("~/data/SS/arab2.txt", row.names=1)
> head(arab2)                                     # look at the first several lines
                                                # for checking
AT1G01010 35 77 40 46 64 60
AT1G01020 43 45 32 43 39 49
AT1G01030 16 24 26 27 35 20
AT1G01040 72 43 64 66 25 90
AT1G01050 49 78 90 67 45 60
AT1G01060 0 15 2 0 21 8

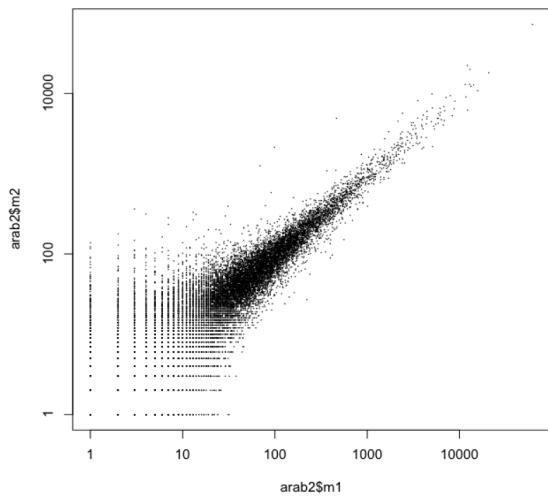
> dim(dat)                                         # get numbers of rows and columns
[1] 26221      6

> colSums(dat)                                    # get column sums
      m1      m2      m3      h1      h2      h3
1902032 1934029 3259705 2129854 1295304 3526579
```

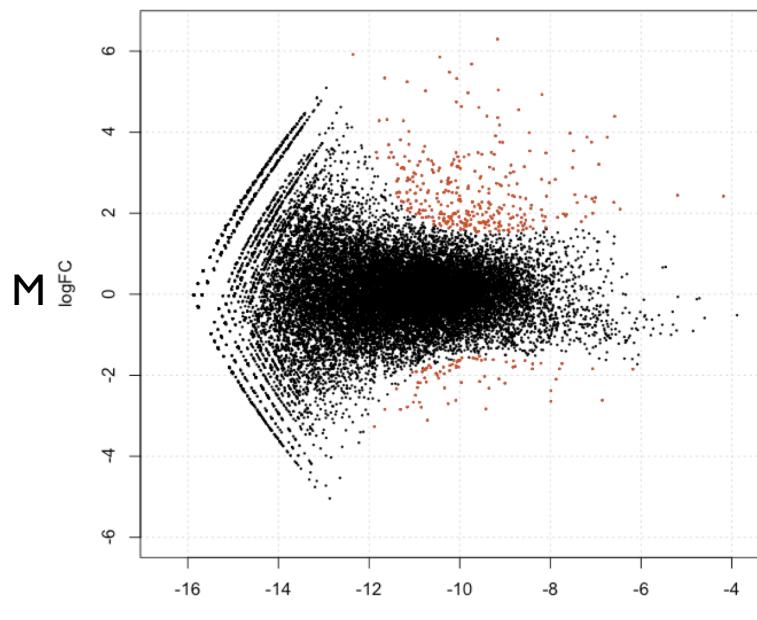
演習問題 ex3

## Let's try: Scatter plot

```
> plot(dat$m1 + 1, dat$m2 + 1, log="xy")
```



## MA plot

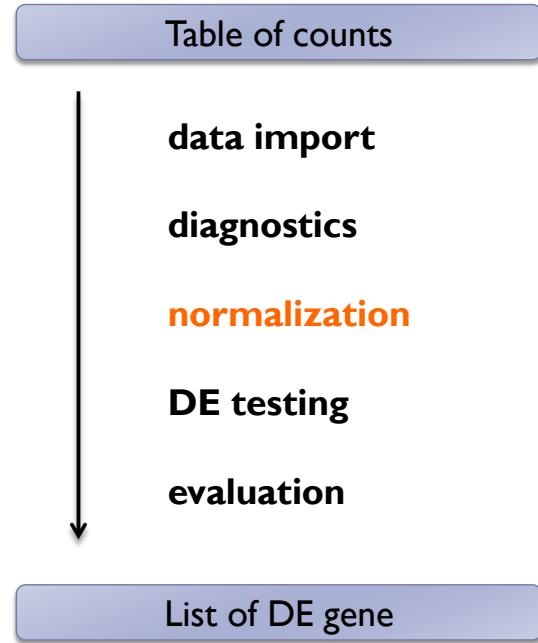


**M:** log fold-change  
**A:** log intensity average

$$M = \log_2(R/G) = \log_2(R) - \log_2(G)$$
$$A = \frac{1}{2} \log_2(RG) = \frac{1}{2}(\log_2(R) + \log_2(G))$$

R: expression level of sample 1  
G: expression level of sample 2

演習問題 ex4



## Normalization

- ▶ What is normalization? Why it is required?
- ▶ Types of normalization.
- ▶ RNAseq specific issue.

# Normalization

## What is normalization? Why it is required?

- ▶ Normalization means to adjust transcriptome data for effects which arise from variation in the technology rather than from biological differences between the RNA samples or between genes.
- ▶ Normalization is an essential step in the analysis of DE from RNA-seq data to make them really comparable.

## Normalization: two types

- ▶ Between-libraries
  - ▶ Comparing expression (counts) of genes between libraries
- ▶ Within-library
  - ▶ Comparing expression (counts) of genes within a library (possible with NGS – in contrast to microarray)

# Normalization

- ▶ **Between-library:**  
gene vs gene **between** libraries/sample

Adjust by the total number of reads

- ▶ CPM (Counts Per Million mapped reads)

$$\text{CPM}_i = \frac{X_i}{N} = \frac{X_i}{N} \cdot \frac{10^6}{10^6}$$

**X<sub>i</sub>:** count of gene  
**N:** number of fragments sequenced

# Normalization

- ▶ **Within-library:**  
gene vs gene **within** sample

Longer transcripts gets higher counts. => Normalized by length

- ▶ RPKM/FPKM (Reads/Fragments Per Kb per Million mapped reads)

$$\text{FPKM}_i = \frac{X_i}{\left(\frac{\tilde{l}_i}{10^3}\right) \left(\frac{N}{10^6}\right)} = \frac{X_i}{\tilde{l}_i N} \cdot 10^9$$

- ▶ TPM (Transcript per million)

$$\text{TPM}_i = \frac{X_i}{\tilde{l}_i} \cdot \left( \frac{1}{\sum_j \frac{X_j}{\tilde{l}_j}} \right) \cdot 10^6$$

**l<sub>i</sub>:** effective length of gene  
**N:** number of fragments sequenced

## ► Relationship between TPM and FPKM

$$\text{TPM}_i = \left( \frac{\text{FPKM}_i}{\sum_j \text{FPKM}_j} \right) \cdot 10^6$$

```

countToTpm <- function(counts, effLen){
  rate <- log(counts) - log(effLen)
  denom <- log(sum(exp(rate)))
  exp(rate - denom + log(1e6))
}

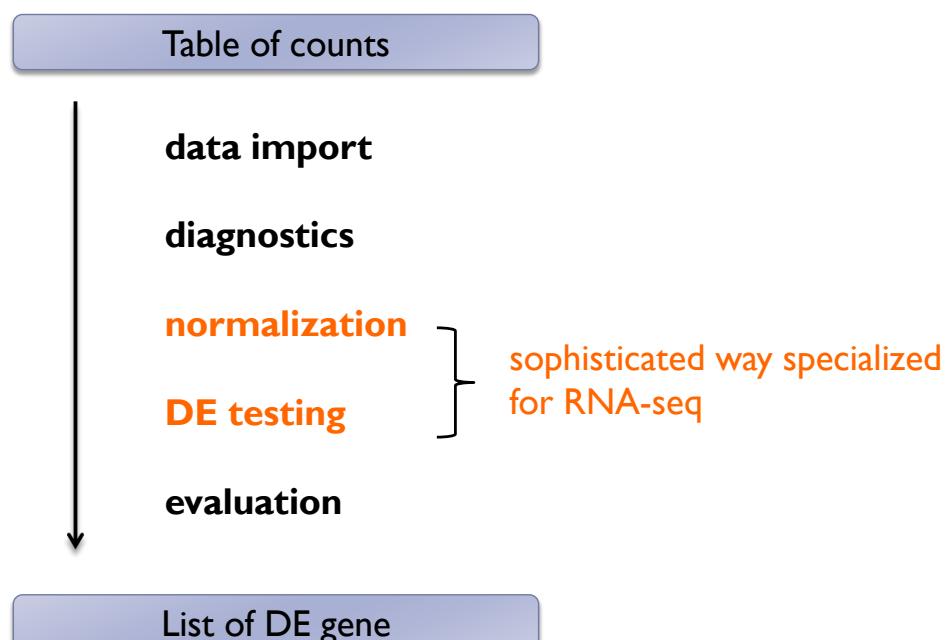
countToFpkm <- function(counts, effLen){
  N <- sum(counts)
  exp( log(counts) + log(1e9) - log(effLen) - log(N) )
}

fpkmToTpm <- function(fpkm){
  exp(log(fpkm) - log(sum(fpkm)) + log(1e6))
}

countToEffCounts <- function(counts, len, effLen){
  counts * (len / effLen)
}

```

<https://haroldpimentel.wordpress.com/2014/05/08/what-the-fpkm-a-review-rna-seq-expression-units/>



## DEG: RNA-seq specific issues

- ▶ RNA-seq count data is Non-Gaussian
- ▶ Normalization: composition effects
- ▶  $N$  (biological replicates) is so small
- ▶ Multiple comparisons (多重検定の問題)

## RNA-seq data is Non-Gaussian

- ▶ RNA-seq data
  - ▶ Discrete-valued data (離散値)
  - ▶ Not normally distributed random variables
  - ▶ **Poisson distribution** for technical replicates
  - ▶ **Negative binomial distribution** for biological replicates.  
(負の二項分布)

## RNA-seq issue: Normalization

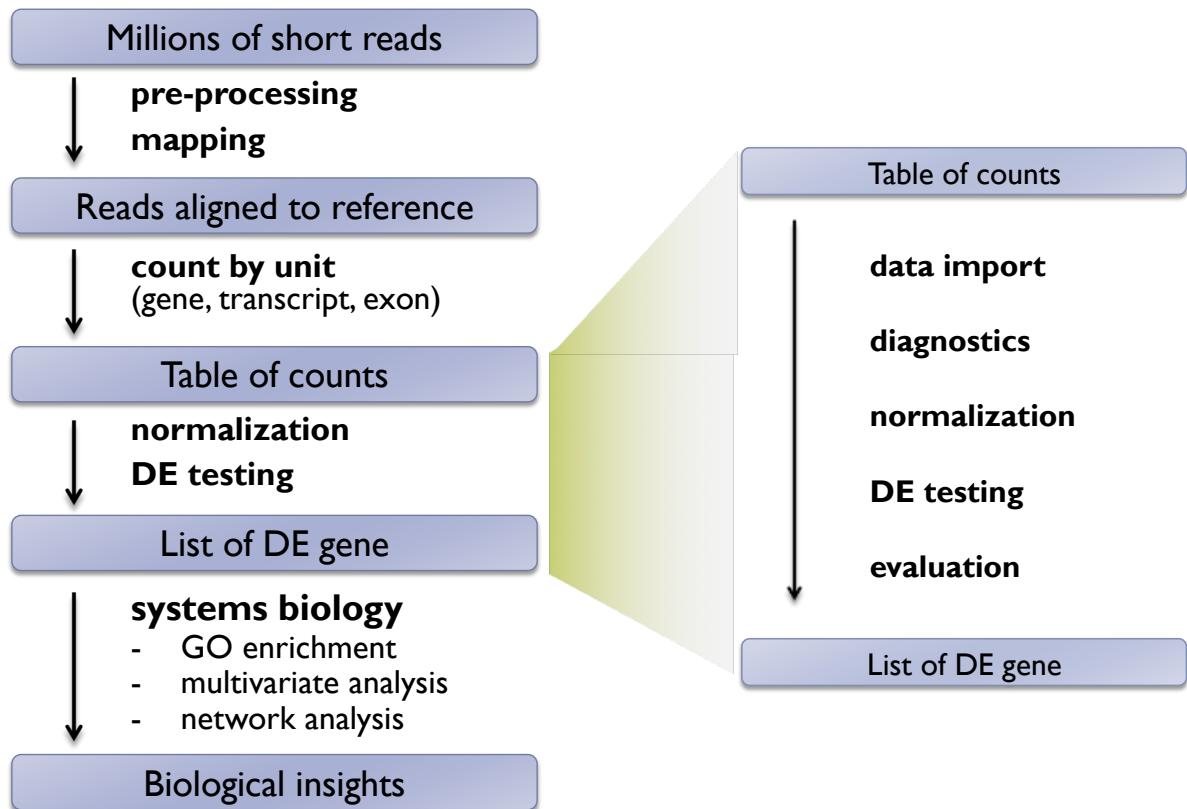
- ▶ Simple normalization
  - ▶ CPM or RPKM/FPKM works well, but not best
- ▶ Composition effects
  - ▶ A small number of highly expressed genes can consume a significant amount of the total sequence.
  - ▶ Strategies
    - ▶ estimate scaling factors from data and statistical models
    - ▶ quantile normalization
    - ▶ ...

## Implementation examples

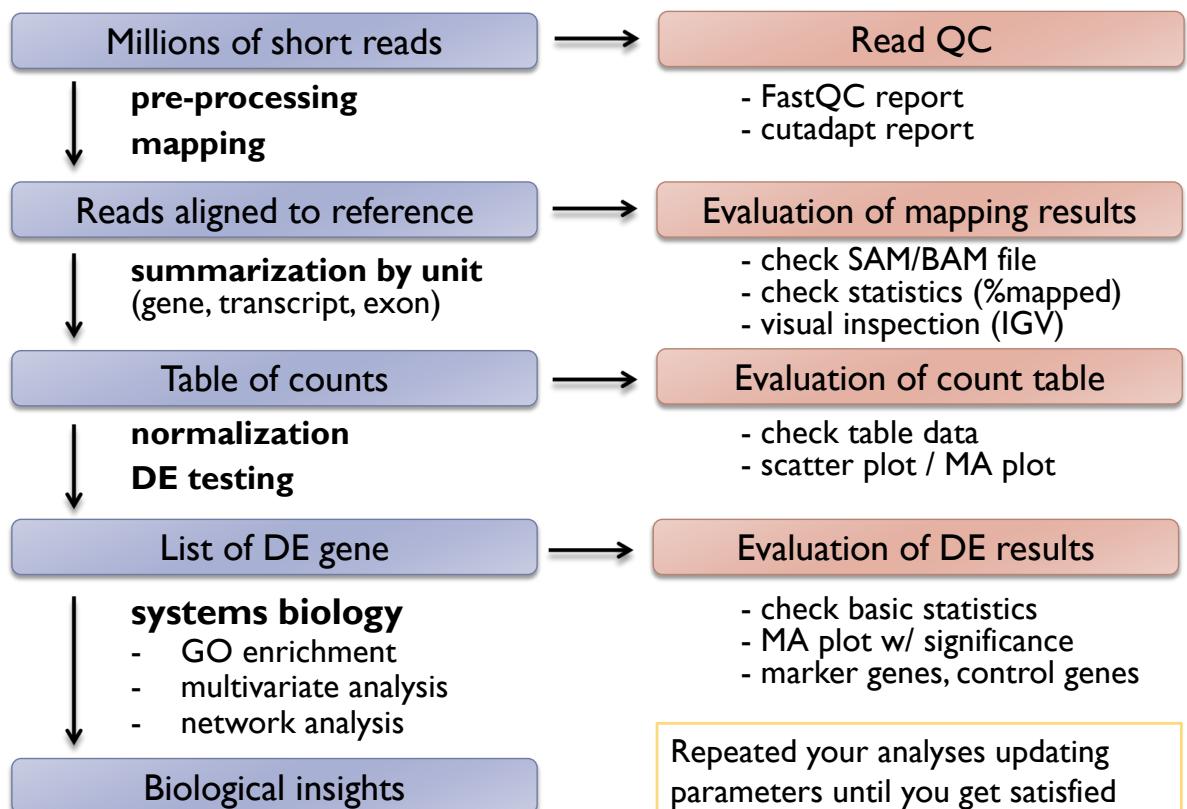
### edgeR

- ▶ **Model:** An over dispersed Poisson model, **negative binomial (NB) model** is used
- ▶ **Normalization:** **TMM method** (trimmed mean of M values; Robinson et al., 2010), **RLE** (Anders et al., 2010) and **upperquantile** (Bullard et al., 2010)

# RNA-seq analysis pipeline for DE



## Check Points



# RNA-seq解析パイプライン： Transcript-based

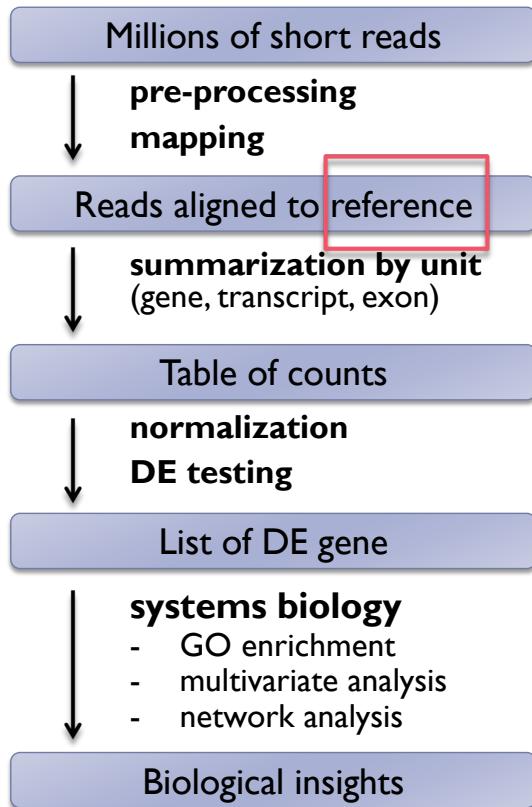
Shuji Shigenobu  
重信 秀治

基礎生物学研究所  
生物機能解析センター

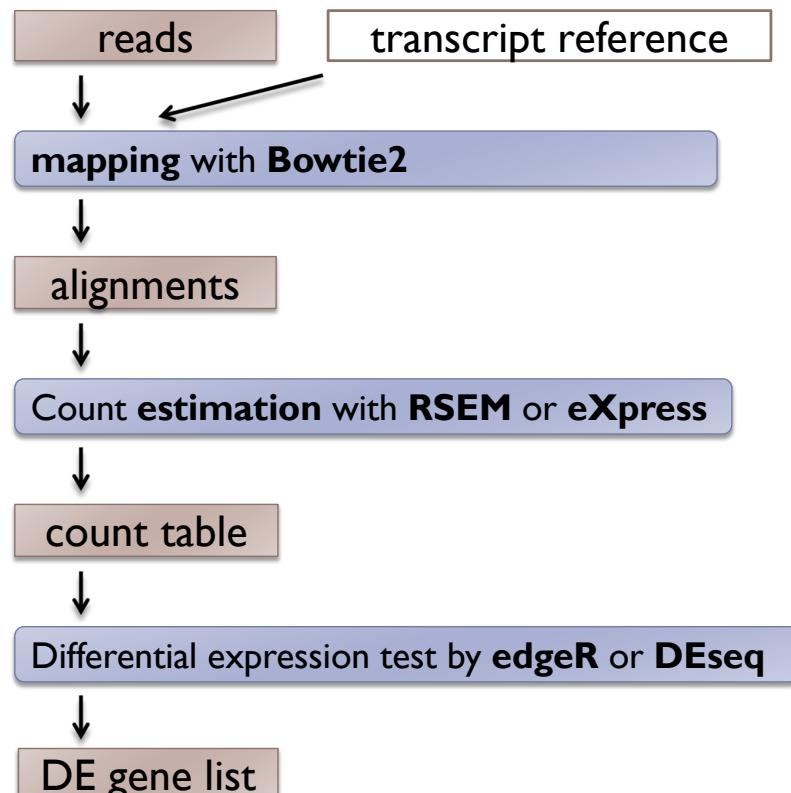


## Two Basic Pipelines

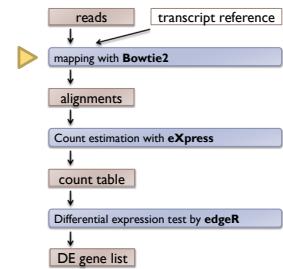
- ▶ Choice of reference
  - ▶ **Genome** – standard for genome-known species
  - ▶ **Transcript** – the only way for genome-unknown species
    - can be used for genome-known species



## A Pipeline: Transcript-based



# Mapping – alignment software



- ▶ For mapping reads onto transcript reference  
*short read mapper (unspliced read aligner)* is used

- ▶ **Bowtie2**

<http://bowtie-bio.sourceforge.net/bowtie2/index.shtml>

## bowtie2

*Bowtie is an ultrafast, memory-efficient short read aligner.*

<http://bowtie-bio.sourceforge.net/bowtie2/index.shtml>

(example)

```
$ bowtie2 -x transcript.fa -U reads.fq -a -S out.sam
```

- ▶ **Input**

- ▶ Reads (fastq) and reference (bowtie2-db)

- ▶ **Output**

- ▶ Alignment in SAM format : **out.sam**

# Let's Try Bowtie2

Align 75-bp Illumina reads with a transcript reference using Bowtie2.

## Prepare reads and reference genome

Sequences for this exercise are stored in `~/data/ss/`.

```
IlluminaReads1.fq - Illumina reads in fastq format  
minimouse_mRNA.fa - a set of transcript sequences
```

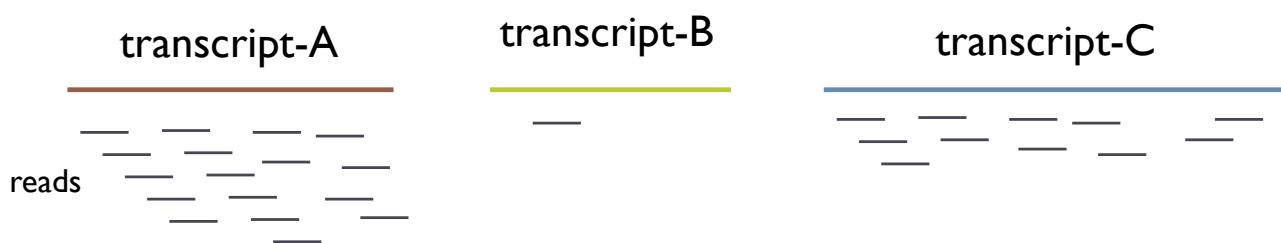
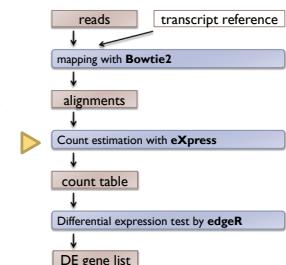
## Build index of reference sequence

```
$bowtie2-build minimouse_mRNA.fa myref
```

## Align reads with reference

```
$bowtie2 -x myref -U IlluminaReads1.fq -a -S out.sam
```

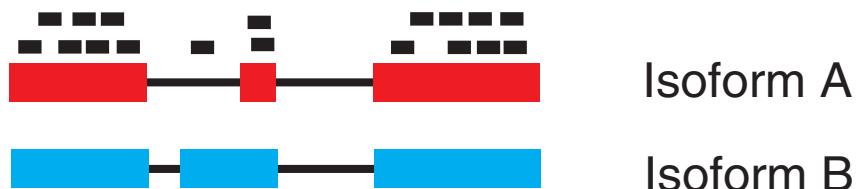
## Count Reads by Transcript/gene



- ▶ The simplest way: just count reads by contig.  
But...
- ▶ Mapping ambiguity should be taken into consideration.

# Estimate Abundance

- ▶ **Multimapping issues**
  - ▶ Isoforms
  - ▶ Very similar paralogs
  - ▶ Repetitive sequences
  - ▶ => cannot align reads uniquely
- ▶ Mapping ambiguity should be taken into consideration.



- ▶ Critical for RNA-seq de novo analysis
- ▶ Software: RSEM and eXpress (EM algorithm)

## eXpress

eXpress is a streaming tool for quantifying the abundances of a set of target sequences from sampled subsequences.

<http://bio.math.berkeley.edu/eXpress/>

(example)

```
$ express transcripts.fasta hits.bam
```

- ▶ **Input**
  - ▶ alignment (bam|sam) and reference (fasta)
- ▶ **Output**
  - ▶ Count estimation table: **results.xprs**



# eXpress

Streaming quantification for high-throughput sequencing



Google Custom Search

[Home](#) [About](#) [Download](#) [Getting Started](#) [Source](#) [Manual](#) [FAQ](#)

## Home

**News**

**02.09.2014 • eXpress Mac 1.5.1 binary updated.**

The previous 1.5.1 binary for OSX was linked with a dynamic Protobuf library that caused the binary to fail on systems without the library installed. The binary has now been updated.

There is no need to update if you were not using this binary or the binary was working for you previously.

**12.08.2013 • eXpress now available in the cloud with eXpress-D!**

Thanks in major part to the amazing work of [Harvey Feng](#), a distributed, batch version of eXpress can now be run on a cluster using [Apache Spark](#) to provide a scalable solution for fragment assignment and abundance estimation. Since eXpress-D uses the full batch EM algorithm, it provides the most accurate estimates according to our tests.

For more details, please read our [manuscript](#) in BMC Bioinformatics and check out the [wiki](#) on GitHub.

**Download**

**Current Release**  
 eXpress 1.5.0  
 → [Mac OS X \(64-bit\) Binary](#)  
 → [Linux \(64-bit\) Binary](#)  
 → [Windows \(64-bit\) Binary](#)  
 → [Source](#)

**Previous Versions**  
 → [View All](#)

**Support**

Email your questions to [ask.xprs@gmail.com](mailto:ask.xprs@gmail.com)

<http://bio.math.berkeley.edu/eXpress/index.html>

# Let's Try eXpress

## Prepare alignments and reference genome

Sequences for this exercise are stored in `~/data/ss/`.

```
IlluminaReads1.fq – Illumina reads in fastq format
out.sam – this file should be generated in the previous bowtie practice
```

## Run eXpress

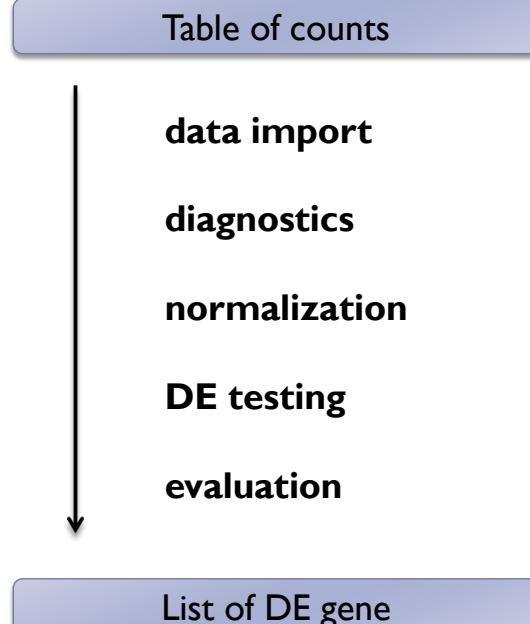
```
$ express minimouse_mRNA.fa out.sam
```

```
Output : results.xprs, params.xprs
```

# eXpress: output

results.xprs

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	bundle_id	target_id	length	eff_length	tot_counts	uniq_counts	est_counts	eff_counts	ambig_distr_alpha	ambig_distr_beta	fpm	fpm_conf_low	fpm_conf_high	solvable
2	1	m.245853	621	398.1	807	15	86.2	134.4	9.83E+01	9.96E+02	2.34E+01	1.88E+01	2.80E+01	T
3	1	m.245856	660	442.0	991	199	919.8	1373.4	5.53E+01	5.46E+00	2.25E+02	2.12E+02	2.38E+02	T
4	2	m.42076	1959	1591.7	156	156	156.0	192.0	0.00E+00	0.00E+00	1.06E+01	1.06E+01	1.06E+01	T
5	3	m.60782	291	83.0	12	12	12.0	42.1	0.00E+00	0.00E+00	1.57E+01	1.57E+01	1.57E+01	T
6	4	m.158451	282	64.5	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
7	5	m.337734	219	39.4	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
8	6	m.338934	261	82.3	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
9	7	m.5973	822	719.9	4	4	4.0	4.6	0.00E+00	0.00E+00	6.01E-01	6.01E-01	6.01E-01	T
10	8	m.337793	219	38.7	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
11	9	m.340910	210	40.5	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
12	10	m.289784	3177	2521.4	350	350	350.0	441.0	0.00E+00	0.00E+00	1.50E+01	1.50E+01	1.50E+01	T
13	11	m.248666	240	61.8	1	1	1.0	3.9	0.00E+00	0.00E+00	1.75E+00	1.75E+00	1.75E+00	T
14	12	m.90727	240	55.7	13	13	13.0	56.1	0.00E+00	0.00E+00	2.53E+01	2.53E+01	2.53E+01	T
15	13	m.338727	216	48.1	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
16	14	m.123519	225	43.2	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
17	15	m.328661	251	50.8	1	1	1.0	4.9	0.00E+00	0.00E+00	2.13E+00	2.13E+00	2.13E+00	T
18	16	m.26062	642	356.1	1	1	1.0	1.8	0.00E+00	0.00E+00	3.04E-01	3.04E-01	3.04E-01	T
19	17	m.1295	240	53.6	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
20	18	m.307626	201	220.2	4	3	3.0	2.7	8.33E+00	4.07E+04	1.47E+00	1.46E+00	1.49E+00	T
21	19	m.307625	204	35.7	301	300	301.0	1718.3	1.02E+01	2.10E-03	9.12E+02	9.05E+02	9.18E+02	T
22	20	m.33508	162	151.3	1	1	1.0	1.1	0.00E+00	0.00E+00	7.15E-01	7.15E-01	7.15E-01	T
23	21	m.109341	183	286.3	2	2	2.0	1.3	0.00E+00	0.00E+00	7.56E-01	7.56E-01	7.56E-01	T
24	22	m.331919	564	277.3	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
25	23	m.23766	303	98.5	3	3	3.0	9.2	0.00E+00	0.00E+00	3.30E+00	3.30E+00	3.30E+00	T
26	24	m.246777	1149	1152.1	631	29	202.5	202.0	1.58E+02	3.90E+02	1.90E+01	1.65E+01	2.15E+01	T
27	25	m.246852	1323	1315.4	761	156	588.8	592.2	1.22E+02	4.85E+01	4.84E+01	4.50E+01	5.19E+01	T
28	26	m.246633	207	31.8	10	4	5.7	37.1	1.29E+04	3.27E+04	1.94E+01	1.05E+01	2.82E+01	T
29	27	m.246662	192	200.4	6	3	3.0	2.9	1.20E+01	3.22E+03	1.63E+00	1.51E+00	1.74E+00	T
30	28	m.99743	1641	1387.9	470	470	470.0	555.7	0.00E+00	0.00E+00	3.66E+01	3.66E+01	3.66E+01	T
31	29	m.335620	234	58.9	0	0	0.0	0.0	0	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
32	30	m.16882	528	297.5	14	14	14.0	24.9	0.00E+00	0.00E+00	5.09E+00	5.09E+00	5.09E+00	T
33	31	m.77438	255	81.4	9	9	9.0	28.2	0.00E+00	0.00E+00	1.20E+01	1.20E+01	1.20E+01	T
34	32	m.131505	450	263.2	18	11	15.8	27.1	8.87E+00	3.95E+00	6.51E+00	4.68E+00	8.35E+00	T
35	33	m.131517	170	195.9	6	0	1.8	1.5	8.17E+00	1.96E+01	9.74E-01	0.00E+00	2.46E+00	T
36	34	m.131504	705	528.2	15	14	14.4	19.2	6.53E+01	1.01E+02	2.95E+00	2.69E+00	3.21E+00	T
37	35	m.131504	705	528.2	15	14	14.4	19.2	6.53E+01	1.01E+02	2.95E+00	2.69E+00	3.21E+00	T



## edgeR

- ▶ A Bioconductor package for differential expression analysis of digital gene expression data
- ▶ **Model:** An over dispersed Poisson model, negative binomial (NB) model, is used
- ▶ **Normalization:** TMM method (trimmed mean of M values) to deal with composition effects
- ▶ **DE test:** exact test and generalized linear models (GLM)

## edgeR

- ▶ **input:** count data (not RPKM or TPM)
- ▶ **output:** gene table with DE significance statistics (FDR)

(example)

```
$ R
> library(edgeR)                      #load edgeR library
> dat <- read.delim("count_data.txt", ...) #import count table to R
> group <- c(rep("M", 3), rep("H", 3))  #assign groups
> D <- DGEList(dat, group=group)        #import data to edgeR
> D <- calcNormFactors(D)              #normalization (TMM)
> D <- estimateCommonDisp(D)          #estimate common dispersion
> D <- estimateTagwiseDisp(D)         #estimate tagwise dispersion
> de <- exactTest(D, pair=c("M", "H")) #DE test
> topTags(de)
Comparison of groups: H-M
      logConc    logFC     P.Value       FDR
AT5G48430 -15.36821 6.255498 9.919041e-12 2.600872e-07
AT5G31702 -15.88641 5.662522 3.637593e-10 4.083773e-06
AT3G55150 -17.01537 5.870635 4.672331e-10 4.083773e-06
...
```

Let's try edgeR

## ex5

Shuji Shigenobu edited this page 6 hours ago · 2 revisions

# ex5: Differential expression analysis with edgeR

arab2データの遺伝子発現の2群間比較を、edgeRで行う。

arab2.txt は、ex3,4でもあつかった、シロイヌナズナRNA-seqのデータである。細菌の感染と非感染でトランск립トームを比較している。実験デザインに関する記述を原著論文から引用する。

Bacteria were grown in King's B media and infiltrated into plants as previously described. Briefly, we used a syringe lacking a needle to infiltrate the abaxial side of leaves of six-week old *Arabidopsis* plants. Plants were infected with either the  $\Delta$ hrcC mutant of *Pseudomonas syringae* pv. tomato DC3000 (PtoDC3000) or mock inoculated with 10 mM MgCl<sub>2</sub> 7 hpi. Each treatment was done as biological triplicates with each pair of replicates done at separate times and derived from independently grown plants and bacteria. -- Cumbie et al., 2011 PLOS ONE

つまり、6 libraries (2 groups x 3 biological replicates) のデータ。すでにマッピング済みで遺伝子毎のリードカウントがタブ区切りテキストとして提供されている。

## Import library

```
> library(edgeR)
```

## Import data

```
> dat <- read.delim("arab2.txt", row.names=1)
```

2グループ、各3繰り返し実験、という実験デザインを定義する。

```
> grp <- c("M", "M", "M", "H", "H", "H")
> grp
[1] "M" "M" "M" "H" "H" "H"
```

edgeRのDGEList関数でカウントデータを読み込む。

```
> D <- DGEList(dat, group=grp)
```

## Normalization

TMM法で、ノーマライズする。calcNormFactorsを使う。

```
> D <- calcNormFactors(D, method="TMM")
```

計算結果の確認

```
> D$samples
   group lib.size norm.factors
m1      M  1902032    1.0399197
m2      M  1934029    1.0611305
m3      M  3259705    0.8841923
h1      H  2129854    1.0266944
h2      H  1295304    1.1412144
h3      H  3526579    0.8747345
```

▼ Pages 17

Find a Page...

[Home](#)

[case1](#)

[case2](#)

[ex1](#)

[ex10](#)

[ex11](#)

[ex12](#)

[ex2](#)

[ex3](#)

[ex4](#)

[ex5](#)

[ex6](#)

[ex8](#)

[ex9](#)

[exercises](#)

Show 2 more pages...

Clone this wiki locally

<https://github.com/nibb-gitc>

[Clone in Desktop](#)

## DE testing

estimate dispersion

```
> D <- estimateCommonDisp(D)
> D$common.dispersion
[1] 0.342609

> D <- estimateTagwiseDisp(D)
> summary(D>tagwise.dispersion)
   Min. 1st Qu. Median Mean 3rd Qu. Max.
0.1173 0.1834 0.4728 1.0540 1.7400 3.7390
```

DE test

```
> de.tagwise <- exactTest(D, pair=c("M", "H"))
```

Multiple comparison correction and View results

```
> topTags(de.tagwise)
Comparison of groups: H-M
  logFC      logCPM      PValue      FDR
AT5G48430 6.233066 6.706315 3.281461e-21 8.604319e-17
AT3G46280 5.078716 8.120404 1.110955e-19 1.456517e-15
AT2G19190 4.620707 7.381817 1.710816e-19 1.495310e-15
AT4G12500 4.334870 10.435847 4.689616e-19 3.074161e-15
AT2G44370 5.514376 5.178263 9.902189e-18 5.192906e-14
AT2G39380 5.012163 5.765848 2.010501e-17 8.786223e-14
AT3G55150 5.809677 4.871425 3.065826e-17 1.148414e-13
AT4G12490 3.901996 10.198755 8.068822e-17 2.455369e-13
AT1G51820 4.476647 6.369685 8.490613e-17 2.455369e-13
AT2G39530 4.366709 6.710299 9.364131e-17 2.455369e-13
```

Dump the table into a text file

```
> write.table(de.tagwise$table, "de.tagwise.txt", sep="\t", quote=F)
```

もしくは、

```
> tmp <- topTags(de.tagwise, n=nrow(de.tagwise$table))
> write.table(tmp$table, "de.tagwise2.txt", sep="\t", quote=F)
```

後者はFDRの値も出力される。

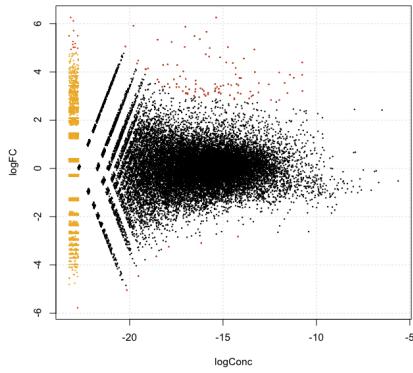
MA plot

edgeR提供のplotSmear関数を使うと便利。

```
plotSmear(D)
```

有意に発現差のある遺伝子を赤色でハイライトすることもできる。

```
> de.names <- row.names(dat[decideTestsDGE(de.tagwise, p.value=0.05) !=0, ])
> plotSmear(D, de.tags=de.names)
```



## Inspect DE result

example: fold-change > 10を抽出・カウント

```
> detab <- tmp$table
# get fold-change > 10
> detab[detab$logFC > log2(10),]
> nrow(detab[detab$logFC > log2(10),])
```

example: FC > 5 AND FDR < 0.01

```
> detab[(detab$logFC > log2(2) & detab$FDR < 0.05), ]
```

edgeRに組み込まれている"decideTestDGE"関数も便利。

```
> summary(decideTestsDGE(de.tagwise, p.value=0.05))
      M+H
Down      65
NotSig 25870
Up       286
```

dumpしたタブ区切りテキストをMS Excelで読み込んで、フィルタ機能やソート機能を使ってデータを探査するのも良いだろう。

## Links

- <http://www.bioconductor.org/packages/release/bioc/html/edgeR.html> | edgeR
- <http://www.bioconductor.org/packages/release/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf> | edgeR User's Guide

## ex12

Shuji Shigenobu edited this page 6 hours ago · 3 revisions

# ex12-1: Differential expression analysis with edgeR (GLM)

arab2データの遺伝子発現の2群間比較を、edgeRで行う。[ex5]と同じデータセットを使い、ここでは、一般化線形モデル(GLM)で、感染サンプルと対照区(mock)の間でdifferential expressionを示す遺伝子(DEG)を同定する。

## Import library

```
> library(edgeR)
```

## Import data

arab2.txt データを読み込む。arab2.txtは data/SS/arab2.txt にある。

```
> dat <- read.delim("arab2.txt", row.names=1)  
# ... dat中身の確認作業 ...
```

2グループ、各3繰り返し実験、という実験デザインを定義する。

```
treat <- factor(c("M", "M", "M", "H", "H", "H"))  
treat <- relevel(treat, ref="M")
```

edgeRのDGEList関数でカウントデータを読み込む。

```
> y <- DGEList(dat, group=treat)
```

## Normalization

TMM法で、ノーマライズする。calcNormFactorsを使う。

```
> y <- calcNormFactors(y, method="TMM")
```

計算結果の確認

```
> y$samples  
  group lib.size norm.factors  
m1      M  1902032   1.0399197  
m2      M  1934029   1.0611305  
m3      M  3259705   0.8841923  
h1      H  2129854   1.0266944  
h2      H  1295304   1.1412144  
h3      H  3526579   0.8747345
```

## Design matrix for GLM

GLMにフィットさせる前に、デザインマトリックスを定義する。

▼ Pages 17

Find a Page...

[Home](#)

[case1](#)

[case2](#)

[ex1](#)

[ex10](#)

[ex11](#)

[ex12](#)

[ex2](#)

[ex3](#)

[ex4](#)

[ex5](#)

[ex6](#)

[ex8](#)

[ex9](#)

[exercises](#)

[Show 2 more pages...](#)

[Clone this wiki locally](#)

<https://github.com/nibb-gitc>

[Clone in Desktop](#)

```

design <- model.matrix(~treat)

> rownames(design) <- colnames(y)
> design
  (Intercept) treatH
m1           1     0
m2           1     0
m3           1     0
h1           1     1
h2           1     1
h3           1     1
attr("assign")
[1] 0 1
attr("contrasts")
attr("contrasts")$treat
[1] "contr.treatment"

```

## DE testing

---

GLMにフィットさせ、係数を確認する。

```

>y <- estimateDisp(y, design)
>y$common.dispersion
[1] 0.3508238

>fit <- glmFit(y, design)
>head(fit$coefficients)
  (Intercept)      treatH
AT1G01010 -10.68003  0.18653587
AT1G01020 -10.91423  0.12030392
AT1G01030 -11.54861  0.36889215
AT1G01040 -10.53809 -0.03178354
AT1G01050 -10.36875 -0.16096270
AT1G01060 -12.81986  0.72272916

```

Rでの線形モデルあてはめ結果を解釈する際の注意点: (Intercept)はこの場合、treatMの値であり、  
treatHはtreatMとの違いを表示している。このような結果の表示は"contrast"をRが設定して線形モ  
デルを解くことに由来する。上で treat <- relevel(treat, ref="M") というように説明変数に「水準  
(levels) ]を設定している。

それでは、いよいよDEGのstatistical testである。glmFit関数を使う。

```

>lrt <- glmLRT(fit, coef=2)
> topTags(lrt)
Coefficient: treatH
  logFC    logCPM      LR      PValue      FDR
AT2G19190 4.548601 7.389099 98.88445 2.676723e-23 7.018636e-19
AT2G44370 5.435579 5.212357 90.98708 1.446132e-21 1.895952e-17
AT3G55150 5.778102 4.915724 88.87656 4.202332e-21 3.672978e-17
AT4G12500 4.391107 10.436950 86.43435 1.444542e-20 9.302502e-17
AT3G46280 4.925035 8.124269 85.73574 2.056619e-20 9.302502e-17
AT1G51820 4.388004 6.384948 85.66768 2.128638e-20 9.302502e-17
AT2G39380 4.949054 5.787376 84.02191 4.893221e-20 1.832931e-16
AT5G64120 3.755076 9.700968 81.02047 2.233916e-19 7.321938e-16
AT2G39530 4.349719 6.722165 80.31178 3.197579e-19 9.315968e-16
AT4G12490 3.877100 10.199985 79.66823 4.428640e-19 1.161234e-15

```

Dump the table to a text file

```

> tab <- topTags(lrt, n=nrow(lrt$table))
> write.table(tab$table, "de.treat.txt", sep="\t", quote=F)

```

## Inspect DE result

---

edgeRに組み込まれている"decideTestDGE"で DEGの数を集計

```

> summary(decideTestsDGE(lrt))
      treatH
Down      96
NotSig  25787
Up       338

```

## ex12-2: Differential expression analysis with edgeR (GLM; considering batch effect)

arab2データの遺伝子発現の2群間比較を、edgeRで行う。[ex5]と同じデータセットを使い、ここでは、一般化線形モデル(GLM)で、感染サンプルと退職の間でdifferential expressionを示す遺伝子(DEG)を同定する。ex12-1からさらに発展させ、ここではbatch effectを考慮する。

上のex12-1では、実験群 vs 対照群のペアワイズ比較を行った。それぞれ繰り返しが3つある実験であるが、原著論文を読むと、実験1, 実験2, 実験3は異なる時期に取得されており、バッチエフェクトが存在する可能性がある。ex12-2では、バッチエフェクトをコントロールしながらDEGを同定する。

```

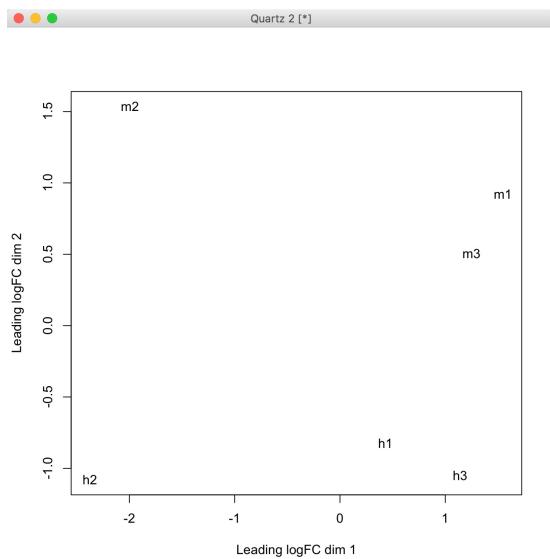
> library(edgeR)
> dat <- read.delim("arab2.txt", row.names=1)
> treat <- factor(c("M", "M", "M", "H", "H", "H"))
> treat <- relevel(treat, ref="M")
> Time <- factor(c(1,2,3,1,2,3))
> y2 <- DGEList(dat, group=treat)
> y2 <- calcNormFactors(y2, method="TMM")

```

## MDS plot

MDS plot はサンプル間の発現の類似性の概略を把握するのに便利である。

```
plotMDS(y2)
```



dim1 (X軸) に沿って、左側にバッチ2が寄っており、バッチエフェクトがありそうだと見当がつく。

## Design matrix for GLM

GLMにフィットさせる前に、デザインマトリックスを定義する。

```

> design2 <- model.matrix(~Time+ treat)
> rownames(design2) <- colnames(y2)
> design2
design2
  (Intercept) Time2 Time3 treatH
1            1     0     0      0
2            1     1     0      0

```

```

3      1      0      1      0
4      1      0      0      1
5      1      1      0      1
6      1      0      1      1
attr("assign")
[1] 0 1 1 2
attr("contrasts")
attr("contrasts")$Time
[1] "contr.treatment"

attr("contrasts")$treat
[1] "contr.treatment"

```

## DE testing

estimate dispersion

```

>y2 <- estimateDisp(y2, design2)
> y2$common.dispersion
[1] 0.07757116

```

model fitting

```

>fit2 <- glmFit(y2, design2)
>head(fit2$coefficients)
(Intercept)      Time2      Time3      treatH
AT1G01010 -10.96162  0.7369950 -0.1554951  0.21596161
AT1G01020 -10.85267  0.1467467 -0.4366442  0.14534329
AT1G01030 -11.63777  0.5221485 -0.2430494  0.27373713
AT1G01040 -10.29519 -0.5574841 -0.2542314 -0.03036512
AT1G01050 -10.39803  0.1959288 -0.1145022 -0.16614196
AT1G01060 -17.25874  5.3363641  3.4094136  0.86928320

```

今回の線形モデルでは:

- (Intercept): Time1
- treatM: 0 (treatHの数値の基準) となっている。

coefficientの大きさは説明変数（及び各水準）の大きさである。次のプロットで横軸にTime2による影響の大きさ、縦軸にhrcCの大きさの違いを遺伝子毎に可視化してみる。

```

> plot(fit2$coefficients[,2], fit2$coefficients[,4], cex=0.1, pch=16,
       xlab="Time2", ylab="hrcC")

```

統計的に有意なバッチエフェクトを持つ遺伝子があるかどうかチェック。

```

> lrt.time <- glmLRT(fit2, coef=2:3)
> sum(p.adjust(lrt.time$table$PValue) < 0.05)
[1] 943

```

発現量の違いがバッチの違いで説明できる遺伝子が、943個もあるので、バッチエフェクトがあると判定できる。従ってバッチエフェクトを考慮してDEGを同定すべきであるといえる。それではいよいよ、DEGの同定である。

```

> lrt.treat2 <- glmLRT(fit2, coef=4)
> topTags(lrt.treat2)
Coefficient: treatH
  logFC      logCPM      LR      PValue      FDR
AT5G48430 6.351014  6.712737 272.5565 3.145880e-61 8.248811e-57
AT3G46280 4.826138  8.125943 225.0412 7.191456e-51 9.428359e-47
AT2G19190 4.540899  7.389687 204.8646 1.812742e-46 1.584397e-42
AT4G12500 4.363887 10.436555 197.3463 7.924090e-45 5.194439e-41
AT2G39530 4.381421  6.722612 182.5254 1.361593e-41 7.140464e-38
AT2G39380 4.986071  5.791255 175.5897 4.450777e-40 1.729617e-36
AT2G17740 4.245929  6.780218 175.5166 4.617412e-40 1.729617e-36
AT1G51800 4.012461  7.723948 174.1586 9.139875e-40 2.995708e-36

```

```
AT1G51820 4.377955 6.384974 172.8187 1.792941e-39 5.223634e-36
AT1G51850 5.372491 5.442450 169.2718 1.067128e-38 2.798116e-35
```

Dump the table to a text file

```
> tab2 <- topTags(lrt.treat2, n=nrow(lrt.treat2$table))
> write.table(tab2$table, "de.treat2.txt", sep="\t", quote=F)
```

## Inspect DE result

---

edgeRに組み込まれている"decideTestDGE"で DEGの数を集計

```
> summary(decideTestsDGE(lrt.treat2))
[1]
-1   914
0   24088
1   1219
```

バッヂエフェクトを考慮しないex12-1 よりも、より多くのDEGが同定された。

## Links

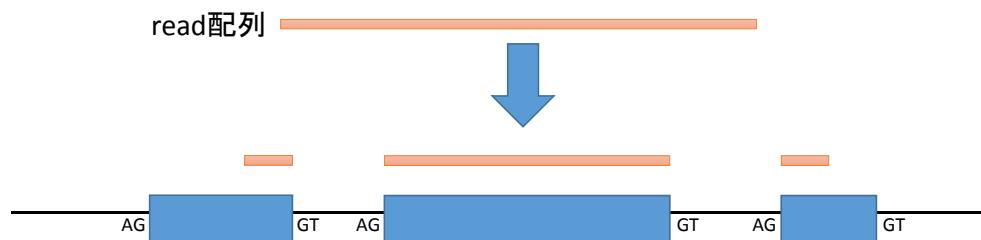
- <http://www.bioconductor.org/packages/release/bioc/html/edgeR.html> | edgeR
- <http://www.bioconductor.org/packages/release/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf> | edgeR User's Guide

# RNA-Seqパイプライン ゲノムベースの解析法

基礎生物学研究所  
生物機能解析センター  
山口勝司

## genomeをレファレンスとする場合

レファレンスがゲノム配列の場合、  
イントロン配列のスプライシングを考慮した  
アライメントを行う必要がある。  
今回はHISATを用いる  
他 Tophat, Blat, SpliceMap, MapSplice, GSMAP, QPALMA



# 実際こんな感じにアラインされる



**TopHat**  
A spliced read mapper for RNA-Seq

JOHNS HOPKINS UNIVERSITY  
CENTER FOR COMPUTATIONAL BIOLOGY  
**CCB**

TopHat is a fast splice junction mapper for RNA-Seq reads. It aligns RNA-Seq reads to mammalian-sized genomes using the ultra high-throughput short read aligner Bowtie, and then analyzes the mapping results to identify splice junctions between exons.

TopHat is a collaborative effort among Daehwan Kim and Steven Salzberg in the Center for Computational Biology at Johns Hopkins University, and Cole Trapnell in the Genome Sciences Department at the University of Washington. TopHat was originally developed by Cole Trapnell at the Center for Bioinformatics and Computational Biology at the University of Maryland, College Park.

» **TopHat 2.1.1 release 2/23/2016**  
Please note that TopHat has entered a low maintenance, low support stage as it is now largely superseded by [HISAT2](#) which provides the same core functionality (i.e. spliced alignment of RNA-Seq reads), in a more accurate and **much more efficient way**.  
Version 2.1.1 is a maintenance release which includes the following changes, some of them thanks to [GitHub](#) contributors:

**OSI certified**

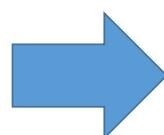
**Site Map**  
[Home](#)  
[Getting started](#)  
[Manual](#)  
[Index and annotation downloads](#)  
[FAQ](#)

Traditional 'Tuxedo' package

TopHat2



Cufflinks



New 'Tuxedo' package

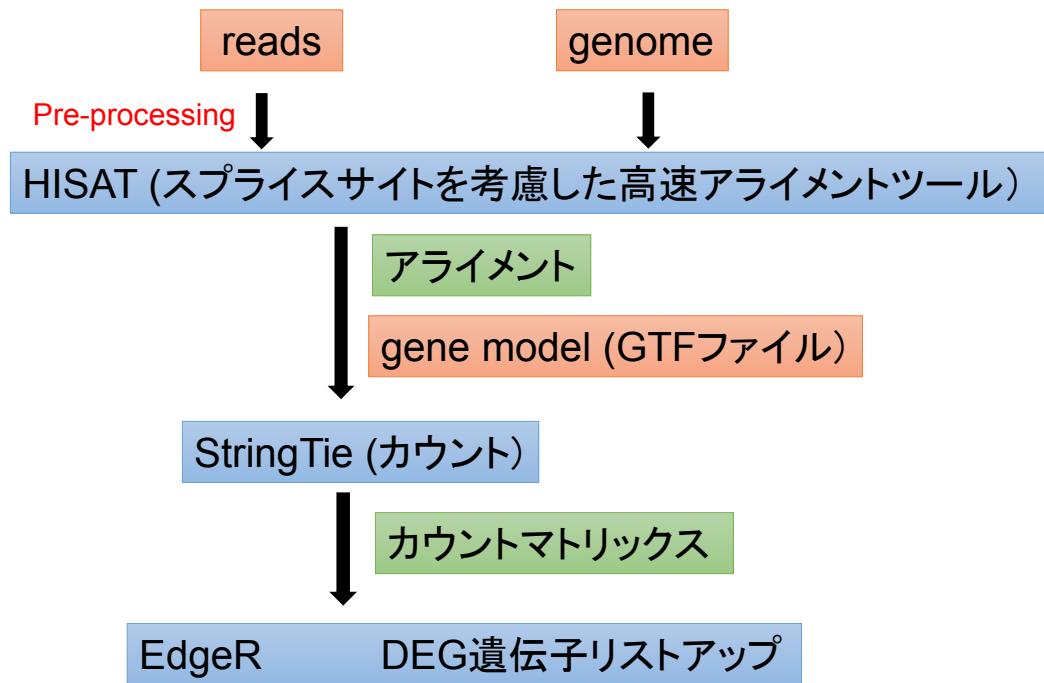
HISAT



StringTie  
Ballgown

劇的に解析速度が速くなった

# 本トレーニングコースでの流れ



## HISAT

**HISAT2**  
graph-based alignment of next generation sequencing reads to a population of genomes

**JOHNS HOPKINS UNIVERSITY**  
CENTER FOR COMPUTATIONAL BIOLOGY  
**CCB**

**OSI certified**

**HISAT2** is a fast and sensitive alignment program for mapping next-generation sequencing reads (both DNA and RNA) to a population of human genomes (as well as to a single reference genome). Based on an extension of BWT for graphs [Sirén et al. 2014], we designed and implemented a graph FM index (GFM), an original approach and its first implementation to the best of our knowledge. In addition to using one global GFM index that represents a population of human genomes, HISAT2 uses a large set of small GFM indexes that collectively cover the whole genome (each index representing a genomic region of 56 Kbp, with 55,000 indexes needed to cover the human population). These small indexes (called local indexes), combined with several alignment strategies, enable rapid and accurate alignment of sequencing reads. This new indexing scheme is called a Hierarchical Graph FM index (HGFM).

**TopHat2と比較して  
とにかく速い**

**HISAT2 2.1.0 release 6/8/2017**

- This major version includes the first release of HISAT-genotype, which currently performs HLA typing, DNA fingerprinting analysis, and CYP typing on whole genome sequencing (WGS) reads. We plan to extend the system so that it can analyze not just a few genes, but a whole human genome. Please refer to the [HISAT-genotype website](#) for more details.
- HISAT2 can be directly compiled and executed on Windows system using Visual Studio, thanks to [Nigel Dyer](#).
- Implemented --new-summary option to output a new style of alignment summary, which is easier to parse for programming purposes.
- Implemented --summary-file option to output alignment summary to a file in addition to the terminal (e.g. stderr).
- Fixed discrepancy in HISAT2's alignment summary.
- Implemented --no-template-len-adjustment option to disable automatic template length adjustment for RNA-seq reads.

**Site Map**  
**Home**  
**Manual** (highlighted)  
**FAQ**

**News and Updates**  
New releases and related tools will be announced through the Bowtie [mailing list](#).

**Getting Help**  
Please use [hisat2.genomics@gmail.com](mailto:hisat2.genomics@gmail.com) for private communications only. Please do not email technical questions to HISAT2 contributors directly.

# HISAT2

graph-based alignment of next generation sequencing reads to a population of genomes



## Table of Contents

Introduction	
What is HISAT2?	
Obtaining HISAT2	
Building from source	
Running HISAT2	
Adding to PATH	
Reporting	
Distinct alignments map a read to different places	
Default mode: search for one or more alignments, report each	
Alignment summary	
Wrapper	
Small and large indexes	
Performance tuning	
Command Line	
Setting function options	
Usage	
Main arguments	
Options	
SAM output	
The <code>hisat2-build</code> indexer	
Command Line	
Notes	
Main arguments	
Options	
The <code>hisat2-inspect</code> index inspector	
Command Line	
Main arguments	
Options	
Getting started with HISAT2	
Indexing a reference genome	
Aligning example reads	
Paired-end example	
Using SAMtools/BCFtools downstream	

パラメータの意味など  
詳しく知るためにには、  
必ずManualを見る

メジャーなモデル生物なら  
indexが用意されている

### Site Map

- [Home](#)
- [Manual](#)
- [FAQ](#)

### News and Updates

New releases and related tools will be announced through the Bowtie [mailing list](#).

### Getting Help

Please use [hisat2.genomics@gmail.com](mailto:hisat2.genomics@gmail.com) for private communications only. Please do not email technical questions to HISAT2 contributors directly.

### Releases

version 2.1.0 6/8/2017

- [Source code](#)
- [Linux x86\\_64 binary](#)
- [Mac OS X x86\\_64 binary](#)
- [Windows binary](#)

### Please cite:

Kim D, Langmead B and Salzberg SL. **HISAT: a fast spliced aligner with low memory requirements.** *Nature Methods* 2015

### Indexes (see note)

<i>H. sapiens</i> , GRCh38	
<i>genome</i>	3.9 GB
<i>genome_snp</i>	4.6 GB
<i>genome_tran</i>	4.1 GB
<i>genome.snp.tran</i>	4.6 GB

## Introduction

### What is HISAT2?

HISAT2 is a fast and sensitive alignment program for mapping next-generation sequencing reads (whole-genome, transcriptome, and exome sequencing data) against the general human population (as well as against a single reference genome). Based on [GCSA](#) (an extension of [BWT](#) for a graph), we designed and implemented a graph FM index (GFM), an original approach and its first implementation to the best of our knowledge. In addition to using one global GFM index that represents general population, HISAT2 uses a large set of small GFM indexes that collectively cover the whole genome (each index representing a genomic region of 56 Kbp, with 55,000 indexes needed to cover human population). These small indexes (called local indexes) combined with several alignment strategies enable effective alignment of sequencing reads. This new indexing scheme is called Hierarchical Graph FM index (HGFM). We have developed HISAT 2 based on the [HISAT](#) and [Bowtie2](#) implementations. HISAT2 outputs alignments in [SAM](#) format, enabling interoperation with a large number of other tools (e.g. [SAMtools](#), [GATK](#)) that use SAM. HISAT2 is distributed under the [GPLv3 license](#), and it runs on the command line under Linux, Mac OS X and Windows.

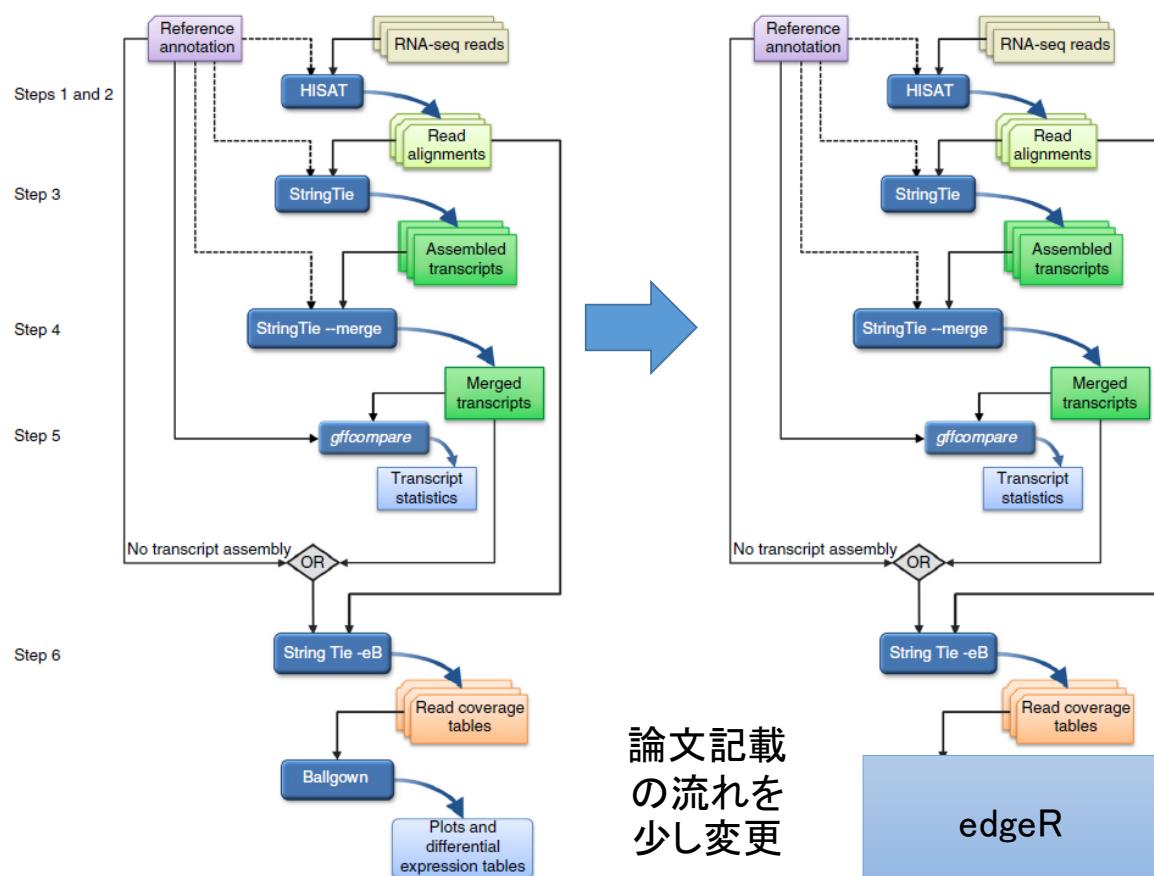
## PROTOCOL

# Transcript-level expression analysis of RNA-seq experiments with HISAT, StringTie and Ballgown

Mihaela Pertea<sup>1,2</sup>, Daehwan Kim<sup>1</sup>, Geo M Pertea<sup>1</sup>, Jeffrey T Leek<sup>3</sup> & Steven L Salzberg<sup>1–4</sup>

<sup>1</sup>Center for Computational Biology, McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins School of Medicine, Baltimore, Maryland, USA. <sup>2</sup>Department of Computer Science, Whiting School of Engineering, Johns Hopkins University, Baltimore, Maryland, USA. <sup>3</sup>Department of Biostatistics, Bloomberg School of Public Health, Johns Hopkins University, Baltimore, Maryland, USA. <sup>4</sup>Department of Biomedical Engineering, Johns Hopkins University, Baltimore, Maryland, USA. Correspondence should be addressed to S.L.S. (salzberg@jhu.edu).

Published online 11 August 2016; doi:10.1038/nprot.2016.095



# hisat2-buildでリファレンスのインデックスを作る

```
$ hisat2-build -h
HISAT2 version 2.0.5 by Daehwan Kim (infphilo@gmail.com, http://www.ccb.jhu.edu/people/infphilo)
Usage: hisat2-build [options]* <reference_in> <bt2_index_base>
      reference_in          comma-separated list of files with ref sequences
      hisat2_index_base     write ht2 data to files with this dir/basename
Options:
  -c                      reference sequences given on cmd line (as
                        <reference_in>)
  --large-index            force generated index to be 'large', even if ref
                        has fewer than 4 billion nucleotides
  -a/--noauto              disable automatic -p/--bmax/--dcv memory-fitting
  -p                      number of threads
  --bmax <int>             max bucket sz for blockwise suffix-array builder
  --bmaxdivn <int>         max bucket sz as divisor of ref len (default: 4)
  --dcv <int>               diff-cover period for blockwise (default: 1024)
  --nodec                 disable diff-cover (algorithm becomes quadratic)
  -r/--noref               don't build .3/.4.bt2 (packed reference) portion
  -3/--justref              just build .3/.4.bt2 (packed reference) portion
  -o/--offrate <int>        SA is sampled every 2^offRate BWT chars (default: 5)
  -t/--ftabchars <int>      # of chars consumed in initial lookup (default: 10)
  --localoffrate <int>      SA (local) is sampled every 2^offRate BWT chars (default: 3)
  --localftabchars <int>    # of chars consumed in initial lookup in a local index (default: 6)
  --snp <path>              SNP file name
  --haplotype <path>        haplotype file name
  --ss <path>                Splice site file name
  --exon <path>              Exon file name
  --seed <int>               seed for random number generator
  -q/--quiet                verbose output (for debugging)
  -h/--help                  print detailed description of tool and its options
  --usage                   print this usage message
  --version                 print version information and quit
```

ヒト・マウス等一部を除き、リファレンス配列のインデックスを作る必要がある

## 実習1 hisat2-build

genome.faはArabidopsis thaliana (シロイヌナズナ) のレファレンスゲノム配列である。

中身を閲覧、query名およびreads数を確認せよ。

```
$ less genome.fa
$ grep '>' genome.fa
$ grep '>' genome.fa | wc
```

indexを作製せよ。

```
$ hisat2-build genome.fa genome
```

新たに作製されたファイルを確認せよ。

```
$ ls
```

# HISAT基本コマンド

```
$ hisat2 -h
HISAT2 version 2.0.5 by Daehwan Kim (infphilo@gmail.com,
www.ccb.jhu.edu/people/infphilo)
Usage:
  hisat2 [options]* -x <ht2-idx> {-1 <m1> -2 <m2> | -U <r> | --sra-acc <SRA accession
  number>} [-S <sam>]

  <ht2-idx>  Index filename prefix (minus trailing .X.ht2).
  <m1>        Files with #1 mates, paired with files in <m2>.
              Could be gzip'ed (extension: .gz) or bzip2'ed (extension: .bz2).
  <m2>        Files with #2 mates, paired with files in <m1>.
              Could be gzip'ed (extension: .gz) or bzip2'ed (extension: .bz2).
  <r>         Files with unpaired reads.
              Could be gzip'ed (extension: .gz) or bzip2'ed (extension: .bz2).
  <SRA accession number>    Comma-separated list of SRA accession numbers, e.g. --
  sra-acc SRR353653,SRR353654.
  <sam>       File for SAM output (default: stdout)

  <m1>, <m2>, <r> can be comma-separated lists (no whitespace) and can be
  specified many times. E.g. '-U file1.fq,file2.fq -U file3.fq'.
```

結果はsamファイルで出力される

## 実習2 hisat2

### read結果

2D2L\_rep1\_R1.fastq

2D2L\_rep1\_R2.fastq

を先にindexを作製したリファレンスにmapさせよ。

```
$ hisat2 -p 4 --dta \
-x genome \
-1 2D2L_rep1_R1.fastq \
-2 2D2L_rep1_R2.fastq \
-S 2D2L_rep1.sam
```

samファイルの内容を確認しよう

```
$ less 2D2L_rep1.sam
```



**StringTie**  
Transcript assembly and quantification for RNA-Seq

JOHNS HOPKINS UNIVERSITY  
CENTER FOR COMPUTATIONAL BIOLOGY  
**CCB**

Home | Manual | FAQ | CCB » Software » StringTie

- Overview
- News
- Obtaining and installing StringTie
- Licensing and contact information
- Publications

### Overview

**StringTie** is a fast and highly efficient assembler of RNA-Seq alignments into potential transcripts. It uses a novel network flow algorithm as well as an optional *de novo* assembly step to assemble and quantitate full-length transcripts representing multiple splice variants for each gene locus. Its input can include not only the alignments of raw reads used by other transcript assemblers, but also alignments longer sequences that have been assembled from those reads. In order to identify differentially expressed genes between experiments, StringTie's output can be processed by specialized software like **Ballgown**, **Cuffdiff** or other programs (DESeq2, edgeR, etc.).

### News

▷ 2/15/2017 - v1.3.3 release

- fixed the computation of the TPM value that was reported incorrectly sometimes in the presence of guides (-G option)
- few other minor fixes

## StringTieを用いてアラインされたreadを数える

StringTieの解析の方向性として大きく2つある

- ・GTFファイルに記載された遺伝子モデルのみを数える
- ・新規な遺伝子モデルを見出し、それも数える  
新規な遺伝子モデルはサンプルによって異なりうるので、  
個々のモデルをStringTieのmerge modeでmergeし、  
それを含めた、新しい遺伝子モデルを作製できる

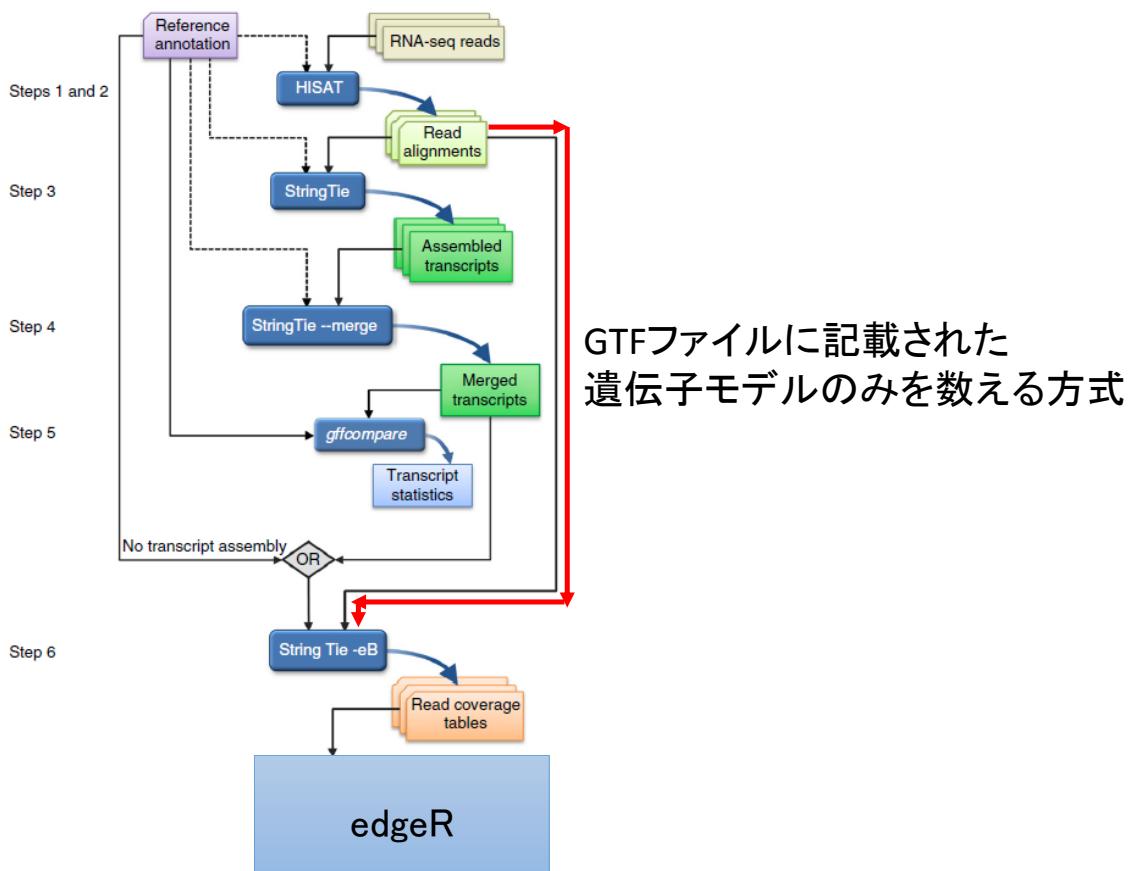
# StringTie基本コマンド

```
$ stringtie -h
StringTie v1.3.3 usage:
  stringtie <input.bam ...> [-G <guide_gff>] [-l <label>] [-o <out_gtf>] [-p
  <cpus>]
    [-v] [-a <min_anchor_len>] [-m <min_tlen>] [-j <min_anchor_cov>] [-f
  <min_iso>]
    [-C <coverage_file_name>] [-c <min_bundle_cov>] [-g <bdist>] [-u]
    [-e] [-x <seqid,...>] [-A <gene_abund.out>] [-h] {-B | -b <dir_path>}
Assemble RNA-Seq alignments into potential transcripts.

:
:
:

Transcript merge usage mode:
  stringtie --merge [Options] { gtf_list | strgl.gtf ...}
With this option StringTie will assemble transcripts from multiple
input files generating a unified non-redundant set of isoforms. In this mode
the following options are available:
  -G <guide_gff>    reference annotation to include in the merging (GTF/GFF3)
  -o <out_gtf>      output file name for the merged transcripts GTF
                    (default: stdout)

:
:
:
```



**StringTie**  
Transcript assembly and quantification for RNA-Seq

JOHNS HOPKINS UNIVERSITY  
CENTER FOR COMPUTATIONAL BIOLOGY  
**CCB**

Home **Manual** FAQ CCB » Software » StringTie

- Running StringTie
- Input files
- Output files
- Evaluating transcript assemblies
- Differential expression analysis
  - Using StringTie with DESeq2 and edgeR
- Assembling super-reads

### Running StringTie

Run `stringtie` from the command line like this:

```
stringtie <aligned_reads.bam> [options]*
```

The main input of the program is a BAM file with RNA-Seq read mappings which must be sorted by their genomic location (for example the `accepted_hits.bam` file produced by `TopHat` or the output of `HISAT2` after sorting and converting it using `samtools` as explained below).

The following optional parameters can be specified when running `stringtie`:

<code>-h/--help</code>	Prints help message and exits.
<code>-v</code>	Turns on verbose mode, printing bundle processing details.
<code>-o [&lt;path/&gt;]&lt;out.gtf&gt;</code>	Sets the name of the output GTF file where StringTie will write the assembled transcripts. This can be specified as a full path, in which case directories will be created as needed. By default StringTie writes the GTF at standard output.
<code>-p &lt;int&gt;</code>	Specify the number of processing threads (CPUs) to use for transcript assembly. The default is 1.
<code>-G &lt;ref_anno.gff&gt;</code>	Use the reference annotation file (in GTF or GFF3 format) to guide the assembly process. The output will include expressed reference transcripts as well as any novel transcripts that are assembled. This option is required by options -B, -b, -e, -C (see below).

```
$ stringtie \
-e \
-B \
-p 4 \
-G genes.gtf \
-o count_genes.gtf \
hoge.sort.bam
```

-G reference annotation to use for guiding the assembly process (GTF/GFF3)  
-e only estimate the abundance of given reference transcripts (requires -G)  
-B enable output of Ballgown table files which will be created in the same directory as the output GTF (requires -G, -o recommended)  
-p number of threads (CPUs) to use (default: 1)  
-o output path/file name for the assembled transcripts GTF (default: stdout)

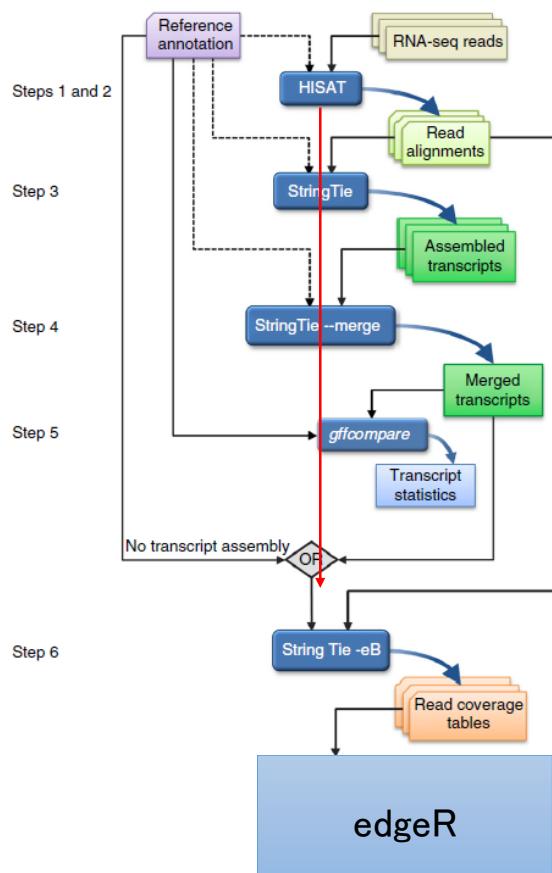
個々のサンプルごと行う

### 実習3 stringtie

HISATで作製したsamをsort.bamにし、StringTieにかける  
hisat結果 2D2L\_rep1.sam

```
$ samtools sort \  
-@ 4 \  
-o 2D2L_rep1.sort.bam \  
2D2L_rep1.sam  
  
$ stringtie -e -B -p 4 \  
-G genes.gtf \  
-o count_2D2L_rep1.gtf \  
2D2L_rep1.sort.bam
```

samtools v1.3以降は  
samファイルのsort,bam化  
を同時にできる



新規な遺伝子モデルを見出し、  
それも数える

```
$ stringtie \
-p 4 \
-G genes.gtf \
-o count_genes.gtf \
hoge.sort.bam
```

-G reference annotation to use for guiding the assembly process (GTF/GFF3)

-p number of threads (CPUs) to use (default: 1)

-o output path/file name for the assembled transcripts GTF (default: stdout)

-e -Bの指定はなし

個々のサンプルごと行う

StringTieのmerge modeでmerged\_gtfファイルを作製する

```
$ stringtie \
--merge \
-p 4 \
-G genes.gtf \
-o stringtie_merged.gtf \
sample.list
```

sample\_lst.txt gtfファイルの場所を指定

```
./ballgown/2D_rep1/2D_rep1.gtf
./ballgown/2D_rep2/2D_rep2.gtf
./ballgown/2D_rep3/2D_rep3.gtf
./ballgown/2D_rep1/2D_rep1.gtf
./ballgown/2D_rep2/2D_rep2.gtf
./ballgown/2D_rep3/2D_rep3.gtf
./ballgown/2D_rep4/2D_rep4.gtf
./ballgown/4D_rep1/4D_rep1.gtf
./ballgown/4D_rep2/4D_rep2.gtf
./ballgown/4D_rep3/4D_rep3.gtf
./ballgown/4D_rep4/4D_rep4.gtf
```

mergeしたgtfファイルを-Gで指定して、先と同様-e -Bを指定し、個々のbamからカウントデータを得る

```
$ stringtie \
-e \
-B \
-p 4 \
-G stringtie_merged.gtf \
-o count_genes.gtf \
hoge.sort.bam
```

-G reference annotation to use for guiding the assembly process (GTF/GFF3)  
-e only estimate the abundance of given reference transcripts (requires -G)  
-B enable output of Ballgown table files which will be created in the same directory as the output GTF (requires -G, -o recommended)  
-p number of threads (CPUs) to use (default: 1)  
-o output path/file name for the assembled transcripts GTF (default: stdout)

個々のサンプルでおこなう

## gtfファイルを比較するツール

### The gffcompare utility

The program `gffcompare` can be used to compare, merge, annotate and estimate accuracy of one or more GFF files (the "query" files), when compared with a reference annotation (also provided as GFF/GTF). A more detailed documentation for the program and its output files can be found here ([gffcompare documentation page](#))

<https://ccb.jhu.edu/software/stringtie/gff.shtml#gffcompare>

```
gffcompare \
-r \
gene.gtf \
-G \
-o merged \
stringtie_merged.gtf
```

gene.gtf <-既知model  
stringtie\_merged.gtf <-含新規model  
この両者を比較できる

```
# gffcompare v0.10.4 | Command line was:
#gffcompare -r genes.gtf -G -o merged stringtie_merged.gtf
#
#= Summary for dataset: stringtie_merged.gtf
#   Query mRNAs : 42241 in 33367 loci (30667 multi-exon transcripts)
#                   (6233 multi-transcript loci, ~1.3 transcripts per locus)
# Reference mRNAs : 41607 in 33350 loci (30127 multi-exon)
# Super-loci w/ reference transcripts: 33240
#-----| Sensitivity | Precision |
# Base level: 100.0 | 99.8 |
# Exon level: 100.0 | 99.4 |
# Intron level: 100.0 | 99.8 |
# Intron chain level: 100.0 | 98.2 |
# Transcript level: 100.0 | 98.5 |
# Locus level: 100.0 | 99.9 |

Matching intron chains: 30127
Matching transcripts: 41607
Matching loci: 33350

Missed exons:      0/169264 ( 0.0%)
Novel exons:       102/170581 ( 0.1%)
Missed introns:    0/127896 ( 0.0%)
Novel introns:     55/128111 ( 0.0%)
Missed loci:        0/33350 ( 0.0%)
Novel loci:         37/33367 ( 0.1%)
```

# Differential expression analysis^

## Differential expression analysis

Together with HISAT and Ballgown, StringTie can be used for estimating differential expression across multiple RNA-Seq samples and generating plots and differential expression tables as described in our protocol paper.

### Using StringTie with DESeq2 and edgeR

DESeq2 and edgeR are two popular Bioconductor packages for analyzing differential expression, which take as input a matrix of read counts mapped to particular genomic features (e.g., genes). We provide a Python script ([prepDE.py](#)) to extract this read count information directly from the files generated by StringTie (run with the -e parameter).

# カウントマトリックス作製

```
$ python prepDE.py -h
Usage: prepDE.py [options]

Generates two CSV files containing the count matrices for genes and
transcripts, using the coverage values found in the output of `stringtie -e`


Options:
  -h, --help            show this help message and exit
  -i INPUT, --input=INPUT, --in=INPUT
                        the parent directory of the sample sub-directories or
                        a textfile listing the paths to GTF files [default:
                        ballgown]
  -g G                  where to output the gene count matrix [default:
                        gene_count_matrix.csv]
  -t T                  where to output the transcript count matrix [default:
                        transcript_count_matrix.csv]
  -l LENGTH, --length=LENGTH
                        the average read length [default: 75]
  -p PATTERN, --pattern=PATTERN
                        a regular expression that selects the sample
                        subdirectories
  -c, --cluster         whether to cluster genes that overlap with different
                        gene IDs, ignoring ones with geneID pattern (see
                        below)
  -s STRING, --string=STRING
                        if a different prefix is used for geneIDs assigned by
                        StringTie [default: MSTRG]
  -k KEY, --key=KEY     if clustering, what prefix to use for geneIDs assigned
                        by this script [default: prepG]
  --legend=LEGEND       if clustering, where to output the legend file mapping
                        transcripts to assigned geneIDs [default: legend.csv]
```

```
$ python prepDE.py
```

defaultではballgownフォルダ一下にあるgtfファイルのカウントマトリックスファイルが作成される。

gene\_count\_matrix.csv  
transcript\_count\_matrix.csv

確認してみよう

```
less gene_count_matrix.csv
```

```
gene_id,2D2L_rep1,2D2L_rep2,2D2L_rep3,2D2L_rep4,2D_rep1,2D_rep2,2D_rep3,4D_rep1,4  
D_rep2,4D_rep3,4D_rep4  
AT4G22890,295,204,203,154,20,22,17,35,26,17,22  
AT1G38440,0,0,0,0,0,0,0,0,0,0,0  
AT3G27910,0,0,0,0,0,0,0,0,0,0,0  
AT1G06620,3,0,6,0,0,3,4,9,0,3,0  
AT5G54067,0,0,0,0,0,0,0,0,0,0,0  
AT2G34630,52,13,10,18,9,0,3,11,7,12,11  
AT2G46660,0,0,0,3,4,0,0,16,23,3,6  
AT2G25590,13,7,7,12,3,4,7,21,15,13,15  
AT1G43171,0,0,0,0,0,0,0,0,0,0,0  
AT5G25130,3,5,3,5,0,0,0,0,0,0,0  
AT2G32280,6,0,7,0,5,0,15,0,5,6,0  
AT3G15020,5,0,4,7,40,9,23,9,18,10,0  
AT5G61100,0,0,0,0,0,0,0,0,0,0,0  
AT5G01650,42,15,27,13,35,19,33,0,23,10,18  
AT5G05570,6,8,4,4,3,5,3,0,11,9,3  
AT3G09770,47,30,25,10,3,14,14,38,46,13,26  
AT3G10210,9,0,5,12,0,7,12,20,9,9,3  
AT5G06000,0,0,0,5,7,0,5,0,0,0,0  
AT5G64620,40,31,20,31,64,35,41,21,37,41,36  
AT1G75280,36,45,36,44,8,11,14,16,10,4,11
```

このカウントマトリックスファイルをedgeRへのinputとして、transcript base解析で扱った同一の方法で解析を進める。

# edgeRでの解析

このケースでは ,が区切りのテキストとして得られているので、  
read.csvを用いる。

```
$ R  
> library(edgeR)  
> dat<-read.csv("gene_count_matrix.csv",row.names=1)  
> group <- c(rep("2D2L",4),rep("2D",3),rep("4D",4))  
> D<-DGEList(dat,group=group)  
> D<-calcNormFactors(D)  
> D<-estimateCommonDisp(D)  
> D<-estimateTagwiseDisp(D)
```

## 2D vs 2D2Lの比較

```
> de_2D_2D2L <- exactTest(D,pair=c("2D","2D2L"))  
> tmp <- topTags(de_2D_2D2L, n=nrow(de_2D_2D2L$table))  
> write.table(tmp$table, "de.tagwise2.txt", sep="\t", quote=F)
```

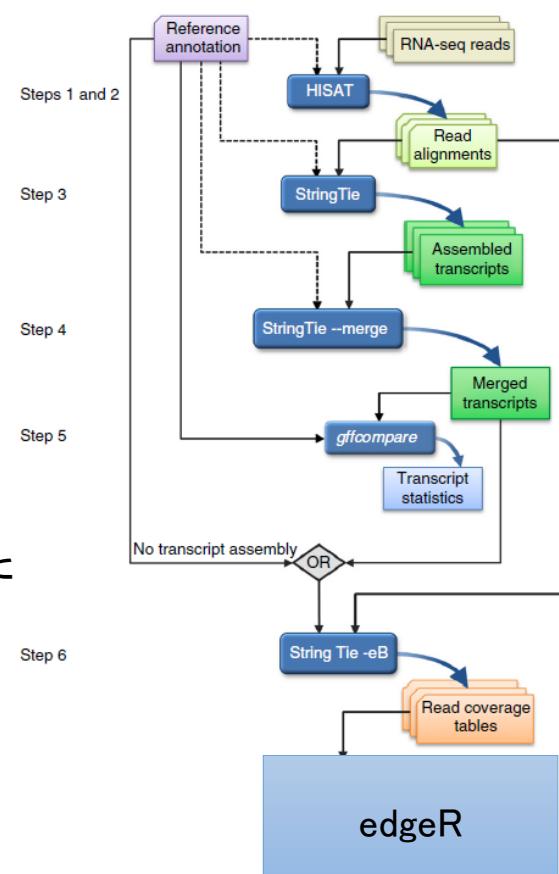
# まとめ

HISAT

StringTie

edgeR

の流れを基盤にした、  
genome baseのDEG解析を紹介した



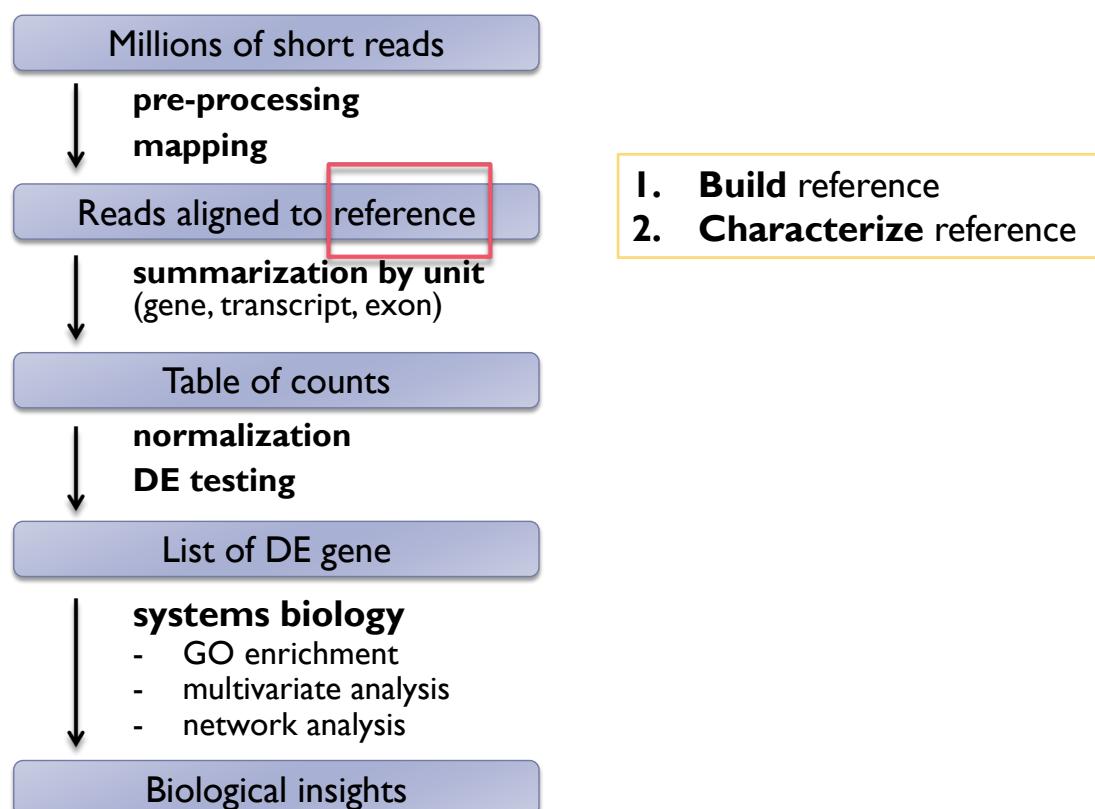
# RNA-seq解析パイプライン： *de novo*

Shuji Shigenobu  
重信 秀治

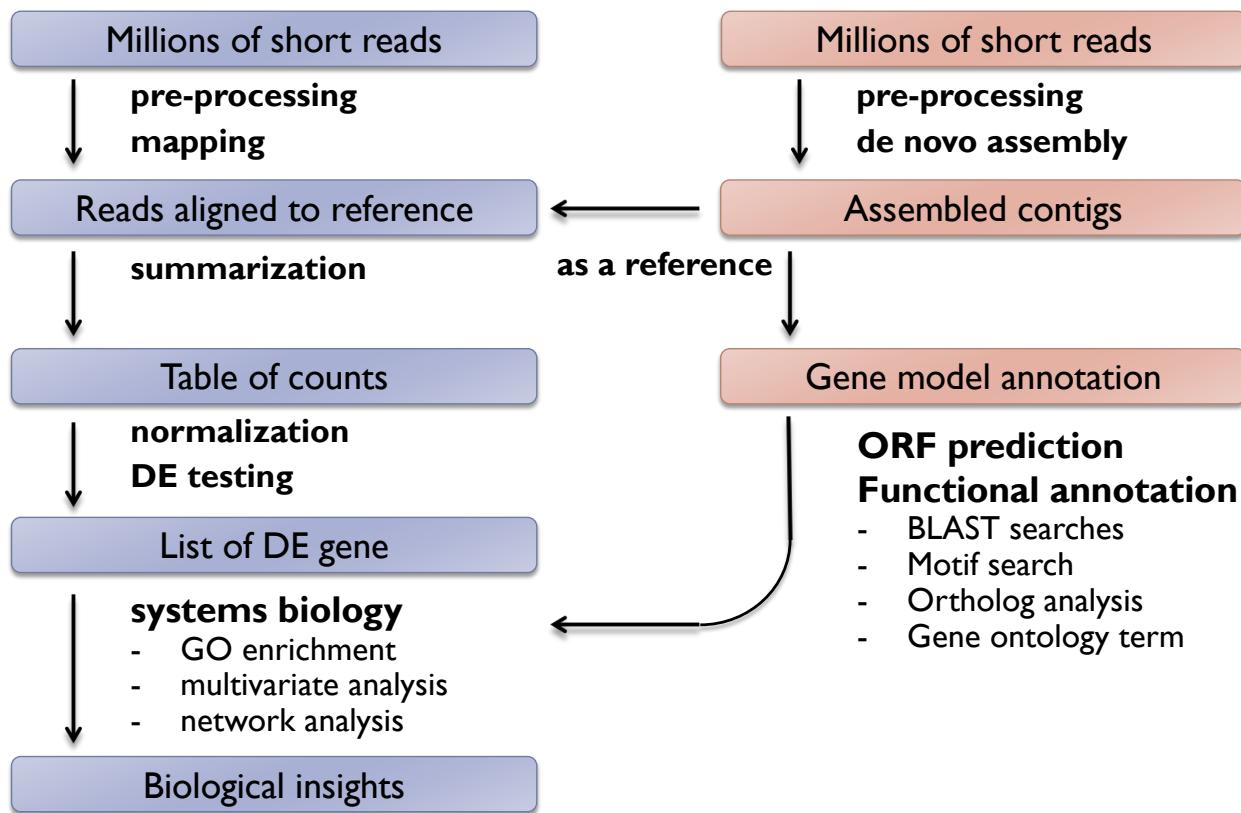
基礎生物学研究所  
生物機能解析センター



## *de novo* RNA-seq



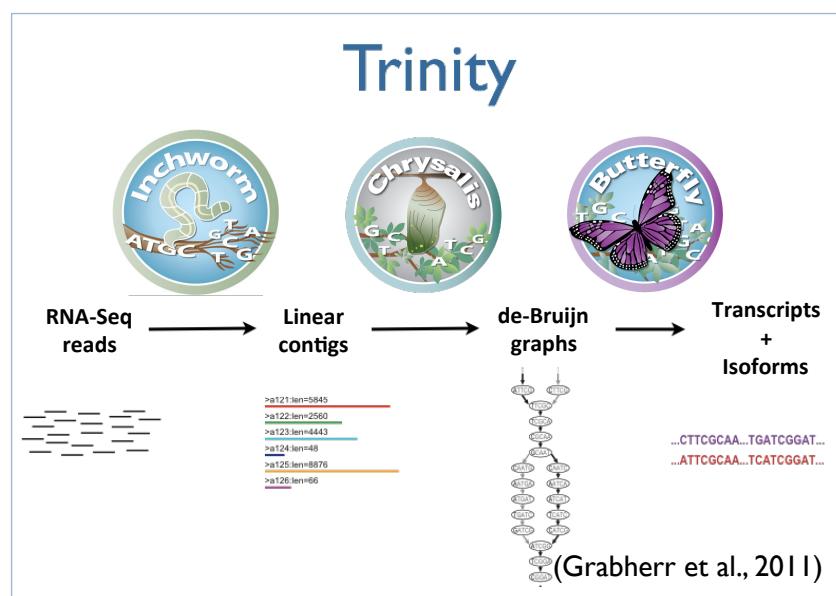
# RNA-seq analysis pipeline (*de novo* strategy)



## *de novo* assemblers of RNA-seq

De novo assemblers use reads to assemble transcripts directly, which does not depend on a reference genome.

- ▶ Trinity
- ▶ Oases
- ▶ TransAbyss
- ▶ EBARDenovo
- ▶ ...



<http://trinityrnaseq.sourceforge.net/>

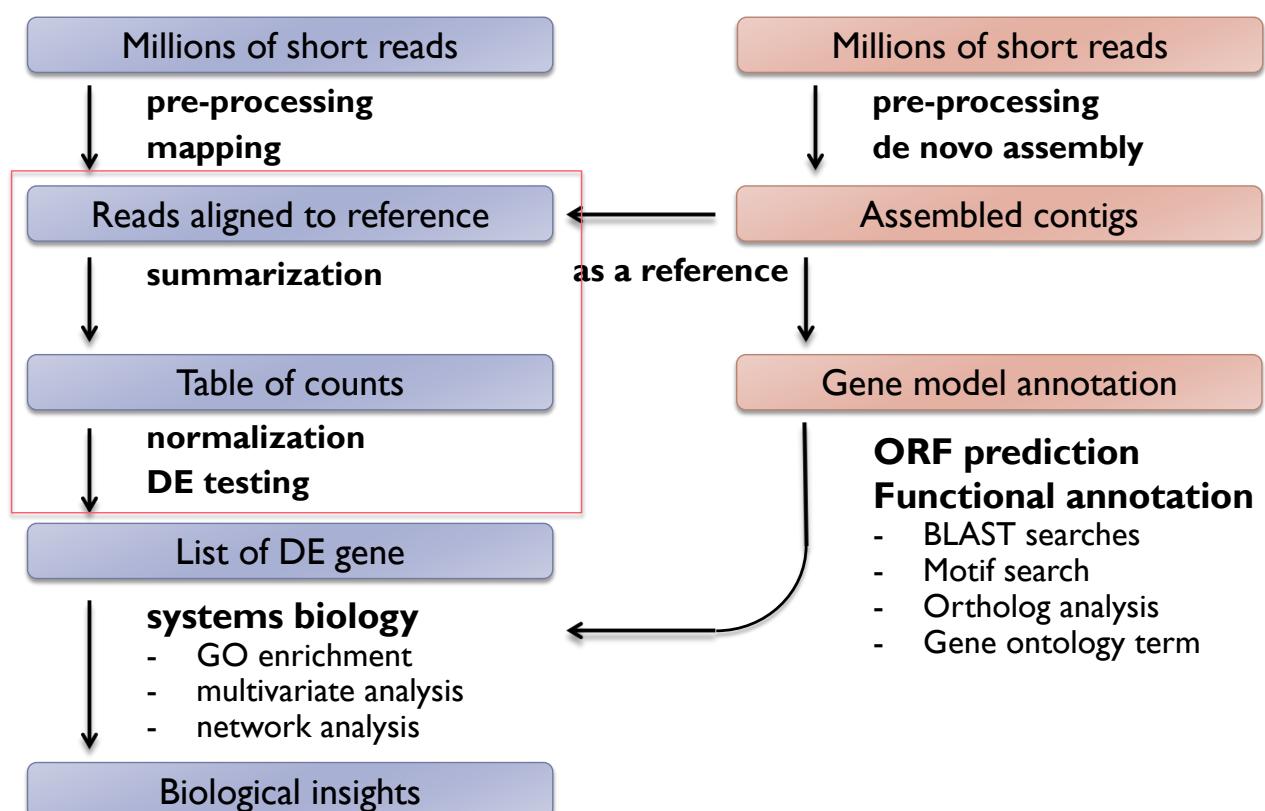
# Trinity example

- ▶ Input: Illumina short reads in FASTQ | FASTA format
- ▶ Output: assembled contigs in FASTA format

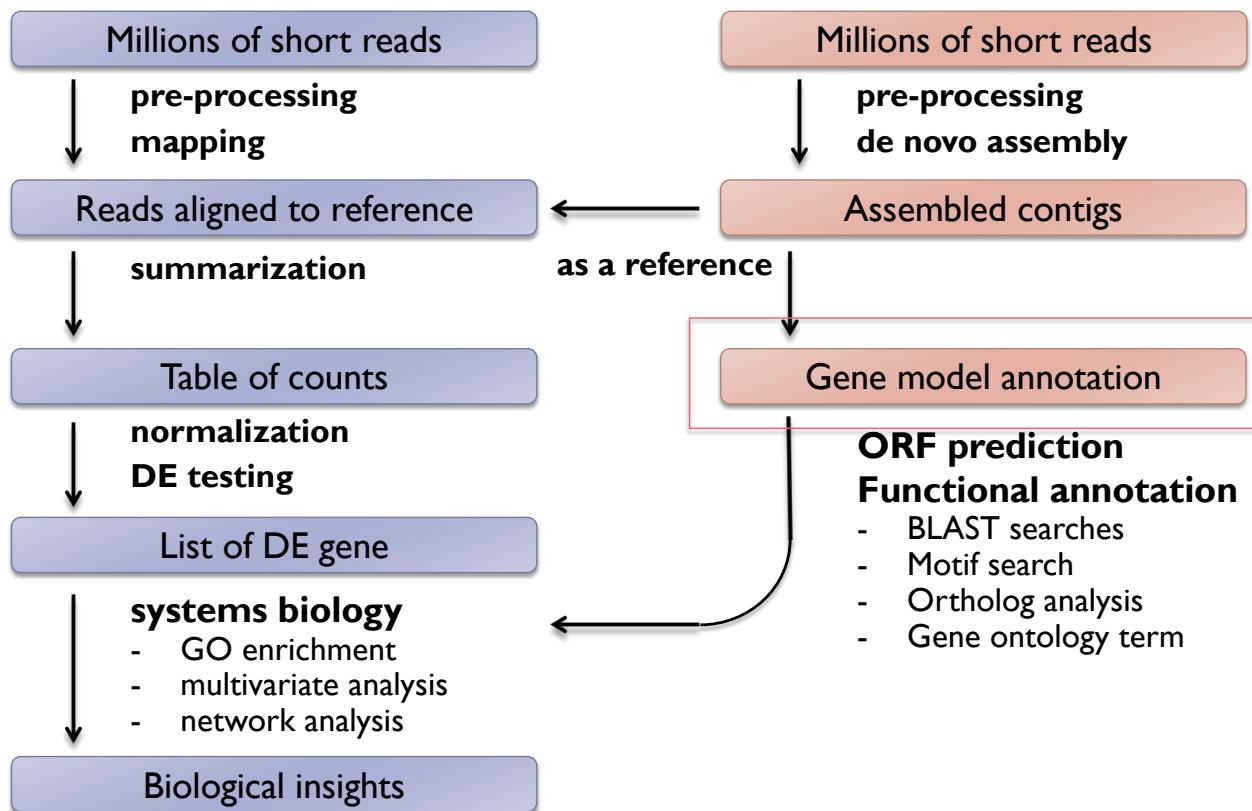
```
# Run Trinity
$ Trinity --seqType fq --left left_all.fq --right right_all.fq \
    --CPU 8 --max_memory 20G
```

(Trinity is supported on only Linux)

## RNA-seq analysis pipeline (*de novo* strategy)



# RNA-seq analysis pipeline (*de novo* strategy)



## ORF prediction

- ▶ Special consideration in ORF prediction after *de novo* RNA-seq assembly
  - ▶ Sometimes partial: Start Met or terminal codon may be missing.
  - ▶ Ideally one ORF is present per contig, but erroneously joined contigs may include multiple ORFs.
  - ▶ Possible frame shifts.
    - ▶ Frame shifts do not occur so often in Illumina, while it happens very frequently in 454 and IonProton.
- ▶ Recommended software: TransDecoder

# Functional Annotation of Predicted ORFs

- ▶ **BLAST**
  - ▶ NCBI NR (or UniProt)
  - ▶ species of interest (model organisms, close relatives etc)
  - ▶ specific DB (SwissProt, rRNA DB, CEGMA etc)
  - ▶ self (assembly v.s. assembly)
- ▶ **Motif search**
  - ▶ Pfam, SignalP etc.
- ▶ **Ortholog analysis**
  - ▶ vs model organism
  - ▶ ortholog database (OrthoDB, eggNOG, OrthoMCL etc)
  - ▶ close relatives
- ▶ **Gene Ontology term assignment**

## Quick annotation by BLASTX

- ▶ **Query: assembled contigs**  
(nucleotide sequences in multi-fasta format)
- ▶ **DB: Protein sequences of a model organism**

### Format DB

```
$ makeblastdb -in protein.fa -dbtype prot
```

### Search

```
$ blastx -query trinity_contigs -db protein.fa \
-num_threads 8 -evalue 1.0e-8 -outfmt 0 > blastxout.txt
```



## 基礎生物学研究所 ゲノムインフォマティクス・トレーニングコース

「BLAST自由自在～配列解析の極意をマスターする」

日時：2016年12月1日（木）10：30～17：30

場所：基礎生物学研究所 （愛知県岡崎市）

受講申込み終了

2018年は秋に開催  
予定

### 講師：

重信 秀治（基礎生物学研究所 生物機能解析センター 特任准教授）

内山 郁夫（基礎生物学研究所 生物機能解析センター 助教）



重信 秀治 特任准教授



内山 郁夫 助教

## Protein motif search using InterProScan

- ▶ Query: Translated ORF sequences
- ▶ Software: InterProScan
  - ▶ <https://github.com/ebi-pf-team/interproscan/wiki>

### Search

```
$ interproscan.sh -I proteins.fasta -f XML,TSV --goterms  
--pathways
```

# Assign Gene Ontology terms

- ▶ Tools
  - ▶ InterProScan
  - ▶ BLAST2GO
  - ▶ Transfer model organisms GO terms based on orthology.

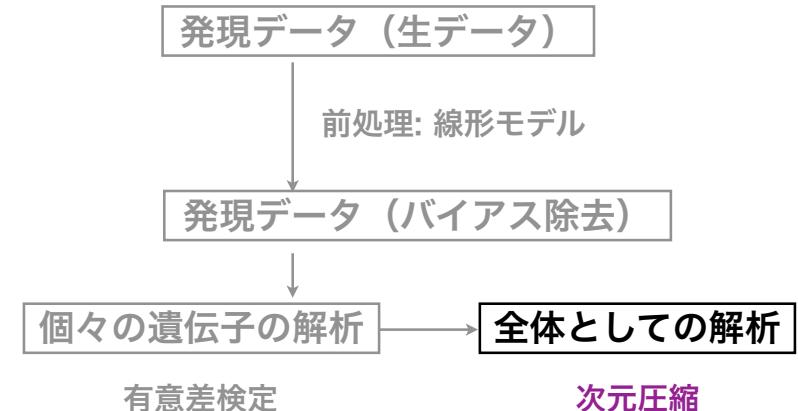
13

# 多変量解析 (特徴空間分割・次元圧縮)

北海道大学 農学研究院  
佐藤昌直

モチベーション:  
多次元（例: 多パラメーター）を  
より少ない指標を使って理解する  
↓  
N個のサンプルをM個 ( $M < N$ ) の  
グループに分類する  
→ 人間が新たな解釈を与える

## 解析の流れ

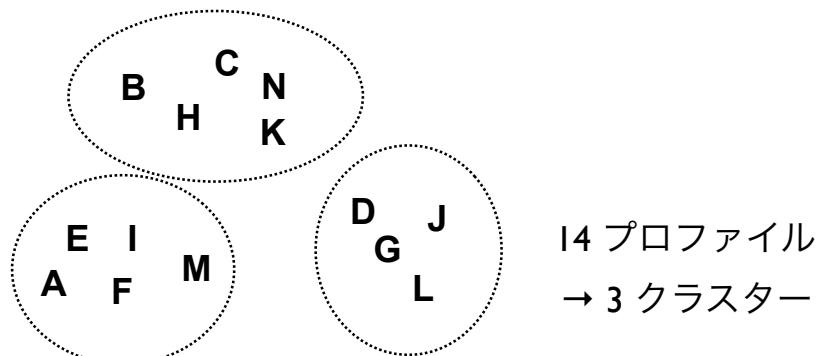


下記のデータセットに含まれる数値を俯瞰してみましょう。データの特徴を読み取れるでしょうか？

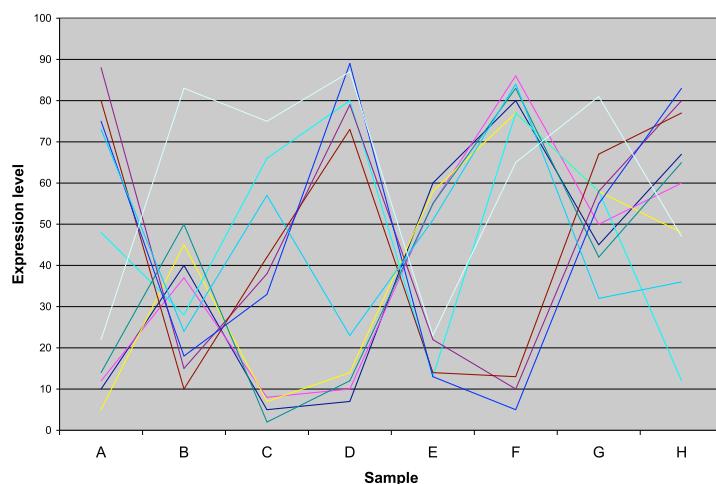
```
inputMatrix<- read.delim("~/data/MS/Sato_A_thaliana-P_syringae_arvRpt2_6h_expRatio_small.txt", header=TRUE, row.names=1)
head(inputMatrix) #読み込みデータの一部を表示
image(t(inputMatrix)) #カラーコードによって可視化
heatmap(as.matrix(inputMatrix)) #階層クラスタリングで解析し、簡易表示
```

高次元（多パラメーター）データの認識における問題をどう扱うか？

クラスタリングによる分類



トランск립トームデータのある一部について可視化してみる

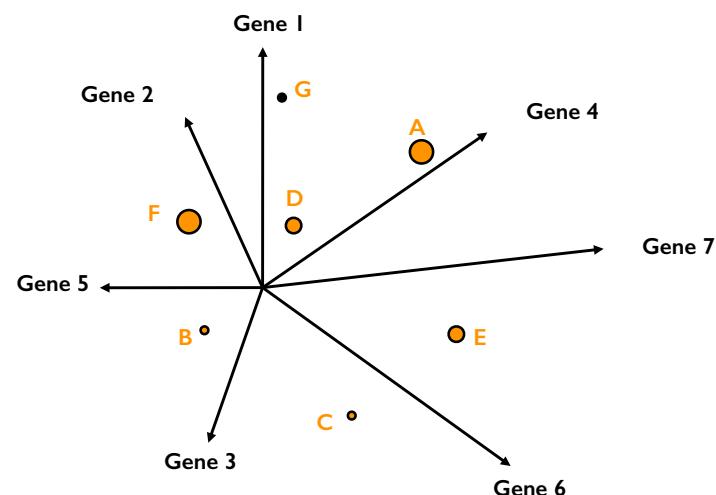


## 多変量解析のポイント

教師有りか無しか  
(supervised or unsupervised)?

どのような距離行列を使うか？

## 7次元の遺伝子発現データセット



コンピューターにどうデータを渡せば  
この問題をどう扱えるか？

人間

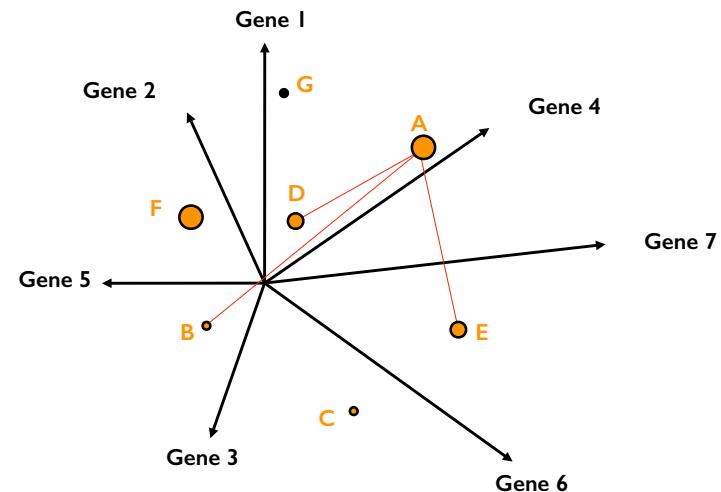
遺伝子発現プロファイル間の  
パターンの比較

問題定義の変換  
(生物学の問題を数学の問題に置き換える)

コンピューター

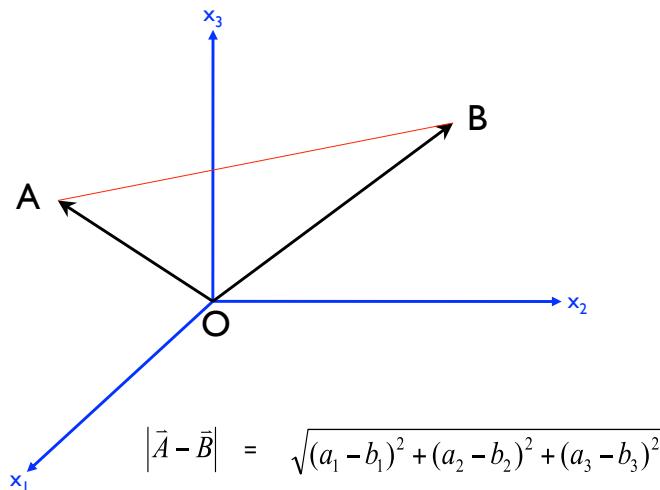
データの大きさに定義される  
次元の空間でのデータポイントの分布の比較

7遺伝子の発現プロファイル間の類似性は  
7次元空間での距離によって決まる

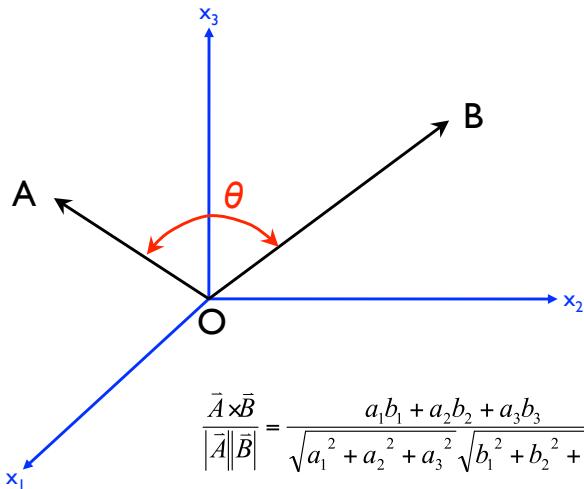


距離の基準を何にするか  
**距離尺度**

ユークリッド距離

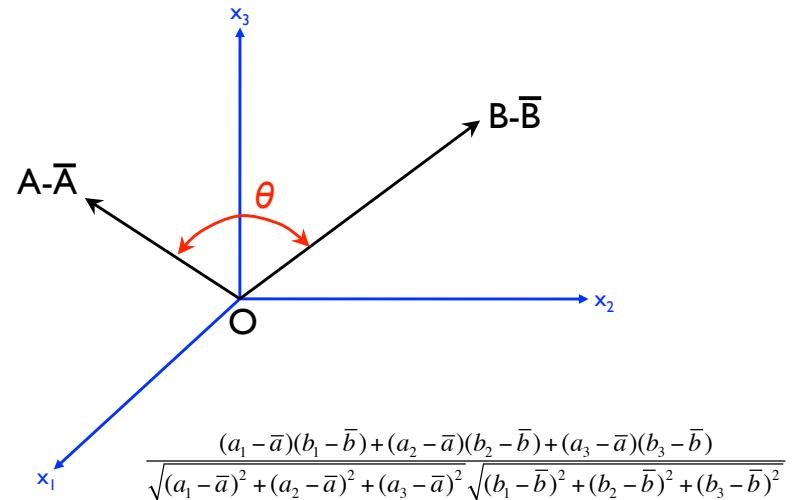


## Uncentered Pearson correlation coefficient = $\cos\theta$



## 相関係数

## Pearson correlation coefficient

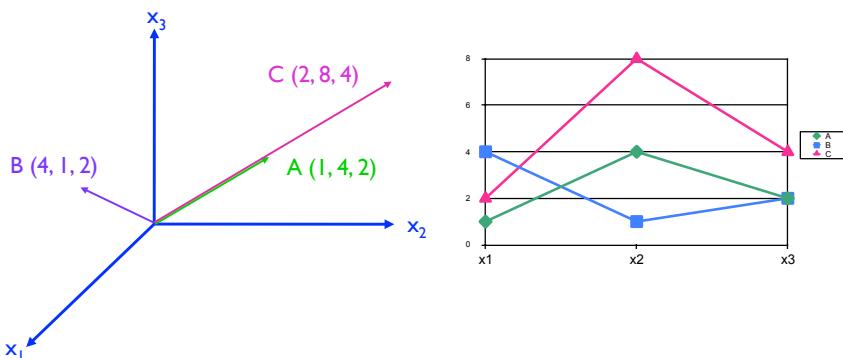


距離尺度の違い→解析対象の違い:

### ポイント

遺伝子発現プロファイルの形と大きさ

- 形: ベクトルの方向
- 大きさ: ベクトルのサイズ



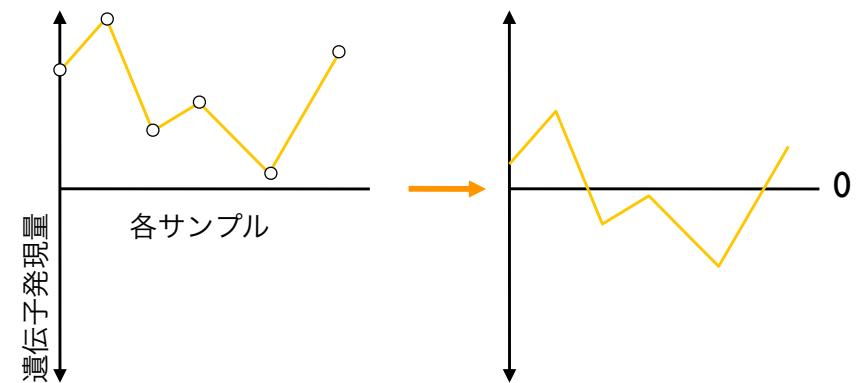
どの距離係数を使うか?

- どんなプロファイルを同じプロファイルと定義するか?
- 距離係数計算の背後にあるものを意識して選択する。

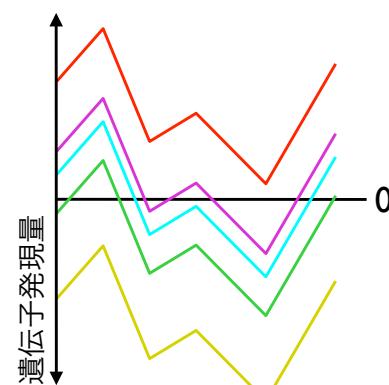
距離係数計算の過程には

- **Centering**: 平均値をゼロにする
- **Scaling**: ベクトルの大きさを1にする

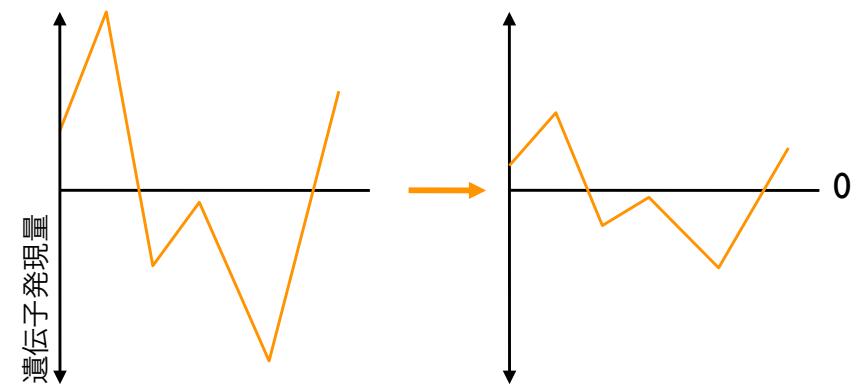
## Centering



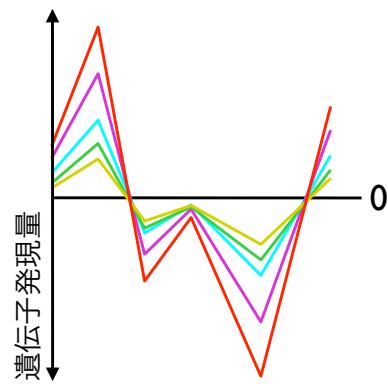
これらはcentering後は  
全く同じプロファイルになる



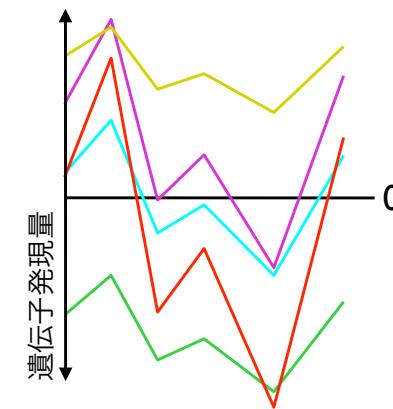
## Scaling



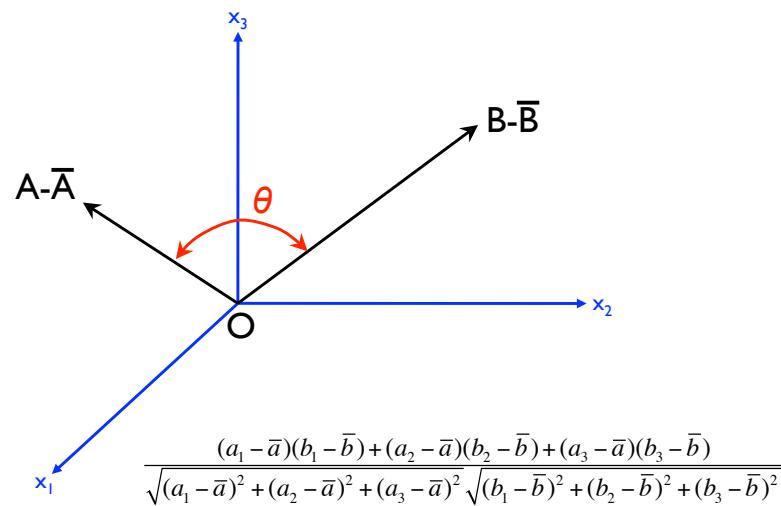
これらはscaling後は  
全く同じプロファイルになる



これらはcentering, scaling後は  
全く同じプロファイルになる



アルゴリズムに注目: 相関係数の場合



ポイント

多変量解析における注意点

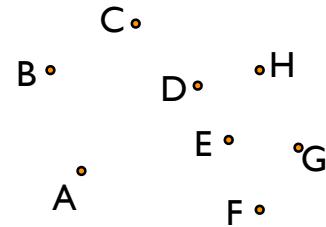
方法依存的に抽出される特徴:

どのような特徴を認識したいのか  
しているのか意識すること

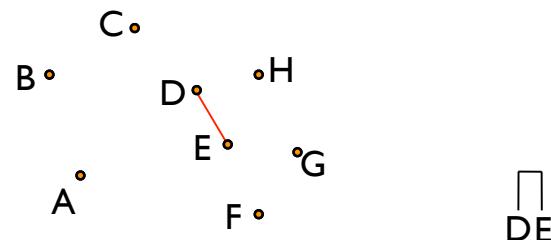
# 多変量解析の実際

階層クラスタリング

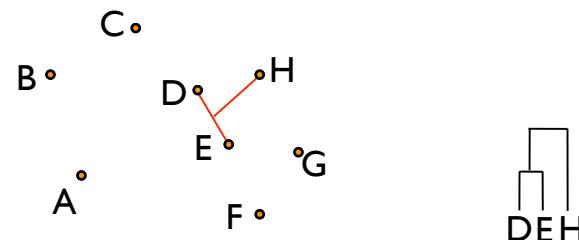
Agglomerative hierarchical clustering



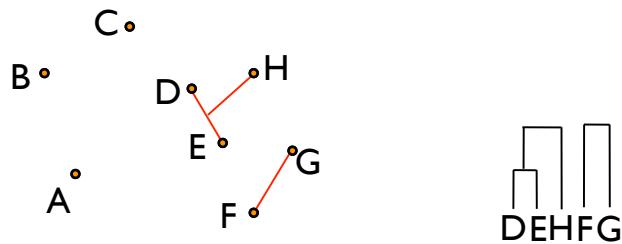
Agglomerative hierarchical clustering



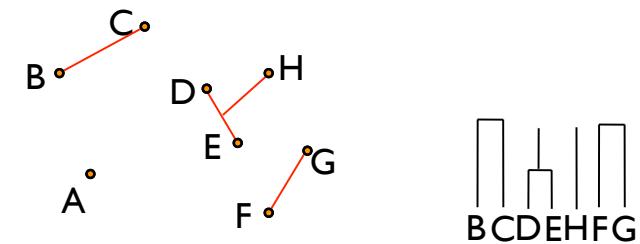
Agglomerative hierarchical clustering



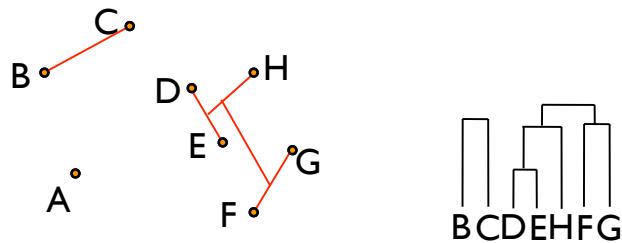
## Agglomerative hierarchical clustering



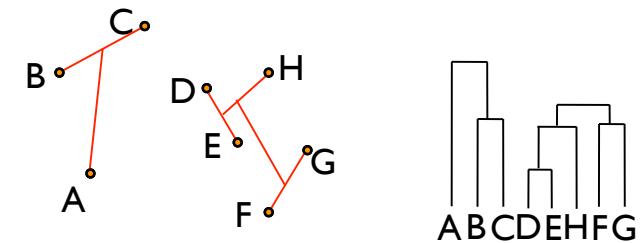
## Agglomerative hierarchical clustering



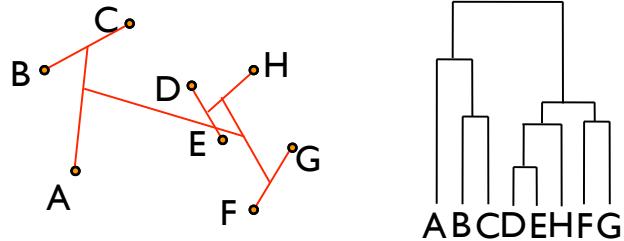
## Agglomerative hierarchical clustering



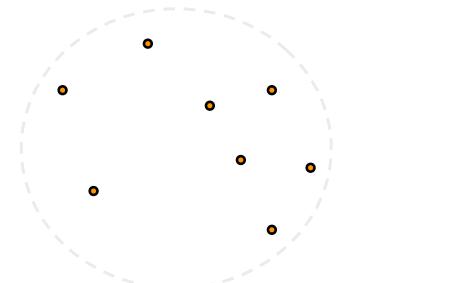
## Agglomerative hierarchical clustering



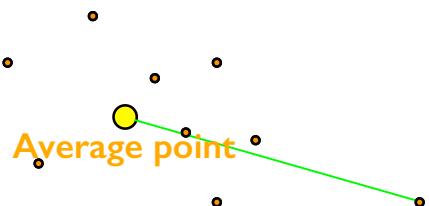
## Agglomerative hierarchical clustering



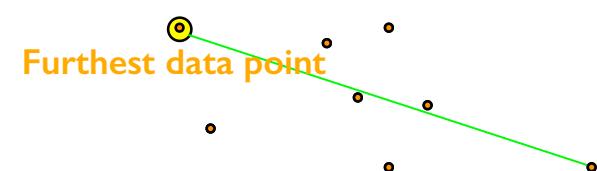
## クラスター定義手法



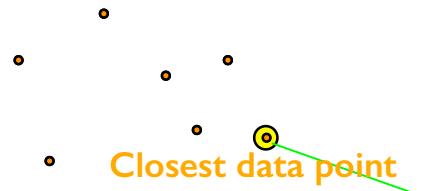
## Average linkage



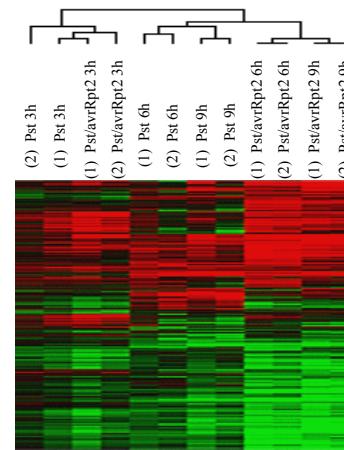
## Complete linkage



# Single linkage



## 階層クラスタリングの利点



- クラスター化してより 少数のカテゴリーを示す
- 人間が認識可能なパターンを示す

## 階層クラスタリングの欠点

- Bottom-up: 非常に「手順」依存性
- 一つの距離のみを指標とした クラスタリング

## 「手順依存的」な方法の欠点を 補うには？

- 偶然、観察されているクラスターを 推定する
  - 同じ手順を繰り返す
  - クロスバリデーション

## クロスバリデーション

- あるクラスターは必然か偶然か？
- leave-one out validation: サンプルを一つ抜いてクラスタリングしてみる
- 少数の特定遺伝子がクラスタリングに影響していないか？
- Bootstrap: 遺伝子サブセットでクラスリングを繰り返してみる

## 主成分分析

## 多変量解析(I)のまとめ

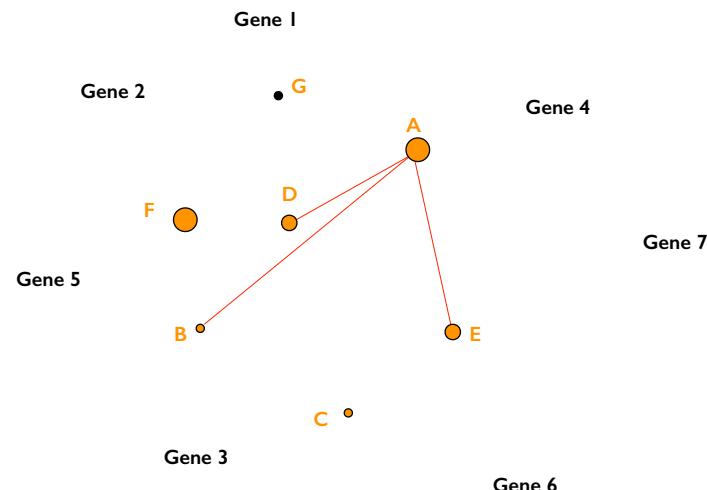
- 教師有りか無しか  
(supervised or unsupervised) ?
- 事前情報、前提はあるか？
  - ある場合はk-means法などの利用を検討
- どのような距離行列を使うか？
- プロファイルの大きさ
  - プロファイルの角度 など

## 主成分分析とは？

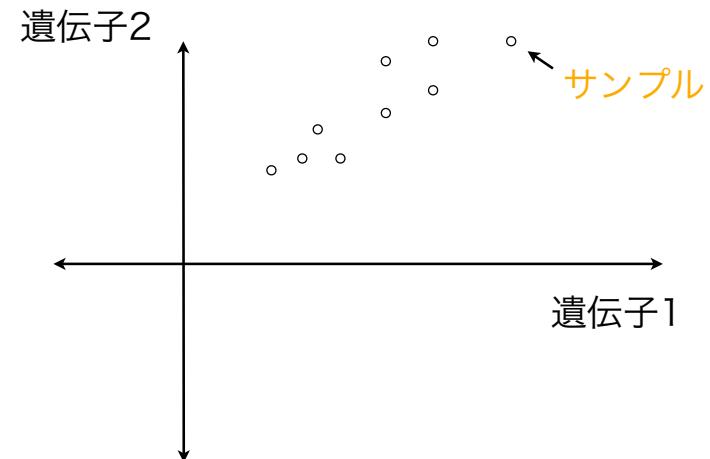
モチベーション:

多数の遺伝子で構成される多次元データ（サンプル）の中で相関のある遺伝子群を使って新たな軸を作り、データを見直す  
→ 人間が新たな解釈を与える

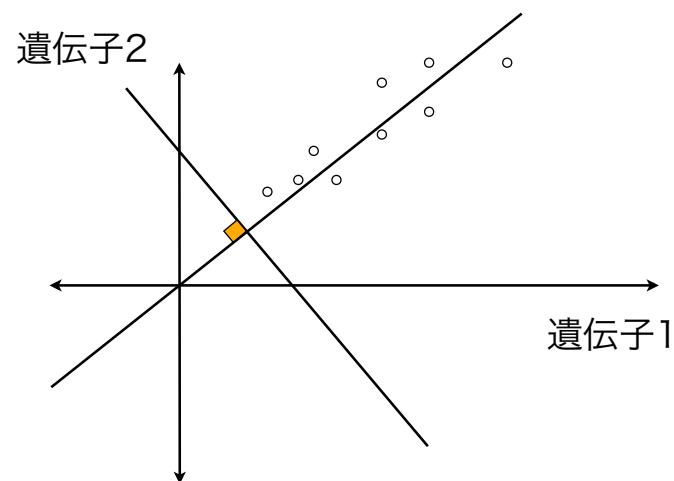
階層クラスタリング、k-means法:  
プロファイル間の類似性は空間での**1つの距離**によって決まる



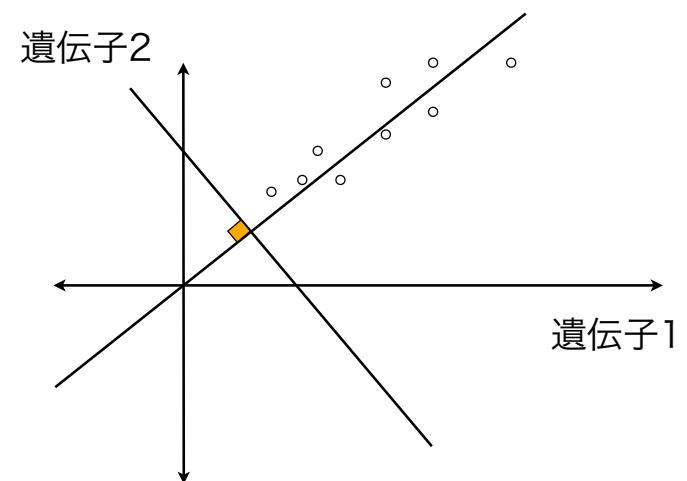
PCAは何をするのか？



PCAは何をするのか？



PCAは何をするのか？



# PCAの概略(2次元)

- 各サンプル ( $1..n$ ) の観察値( $x_n, y_n$ )を

$$u_n = a_1 x_n + b_1 y_n$$
$$v_n = a_2 x_n + b_2 y_n$$

とおく

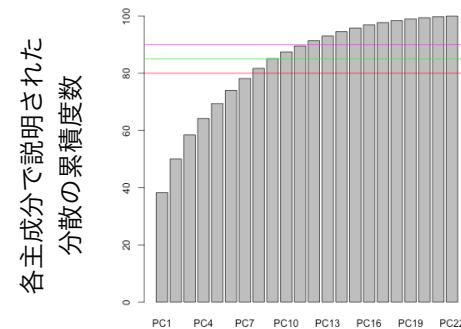
- $a^2 + b^2 = 1$ ,  $u$ と $v$ の相関係数0という制約の下でこれを解いて  $a_n, b_n$  を求める。

## PCAで得られる重要な統計量

- 寄与率
- 因子負荷量
- 主成分得点

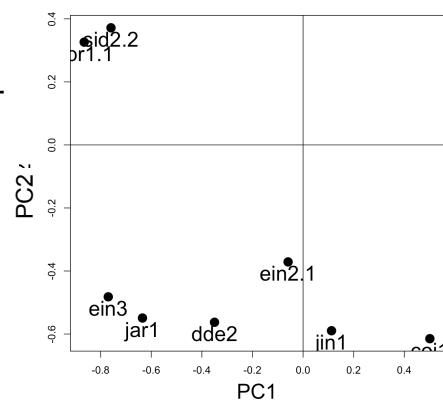
## 寄与率

- 各主成分が説明する分散の割合



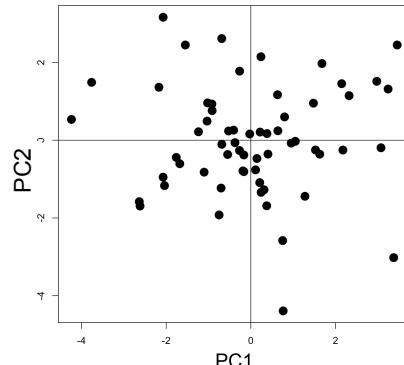
## 負荷量 loadings

- 得られた主成分と元データのパラメーターの相関
- 各パラメーターがもとのデータの情報をどれだけ有するか



## 主成分得点 scores

- 各パラメーターの値を各主成分について標準化したもの



標準化: 平均0, SD=1

## 主成分分析(まとめ)

- 主成分分析はデータの分散を説明する新たな軸を計算する方法
  - 寄与率
  - 因子負荷量
  - 主成分得点

## 注意点

1. デフォルトのprincompでは返り値loadingsは因子負荷量ではない。

2. 相関を使うか、分散共分散行列を使うか

## 多次元尺度構成法

Multi-dimensional scaling(MDS),  
Principle coordinate analysis

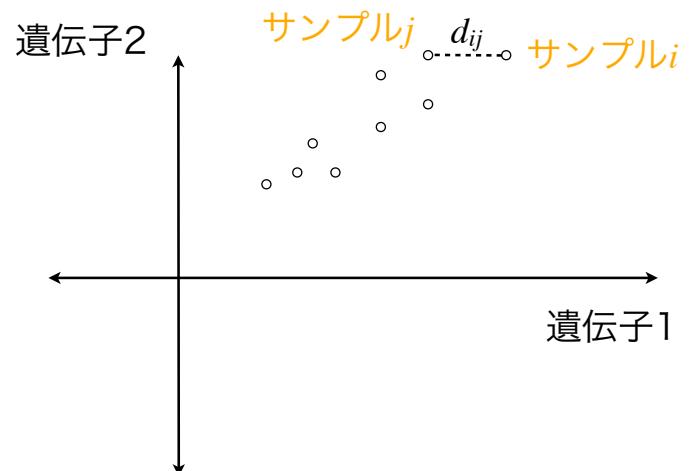
## 多次元尺度構成法とは？

### モチベーション：

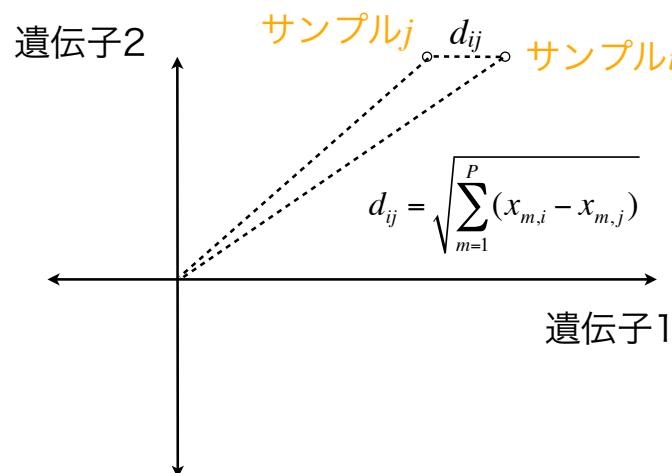
多数の遺伝子で構成される多次元の中で各サンプル間の違いを低次元で表現する

距離係数を元に次元圧縮するため、非線形の関係にも対応 (PCA: 分散を使う [線形]。計算手法によってはPCAと同義になる)

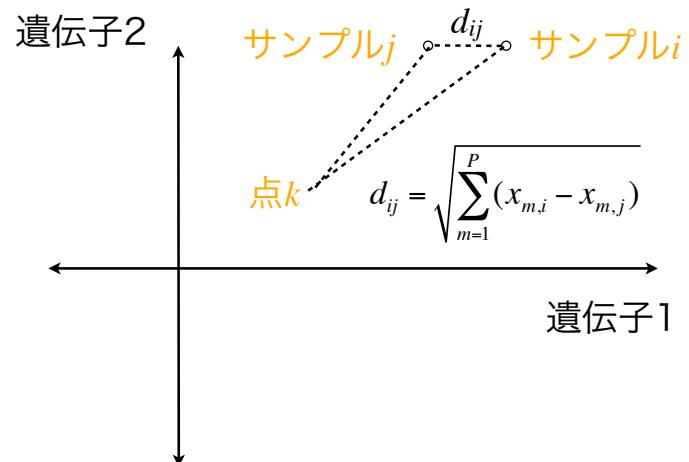
## MDSは何をするのか？



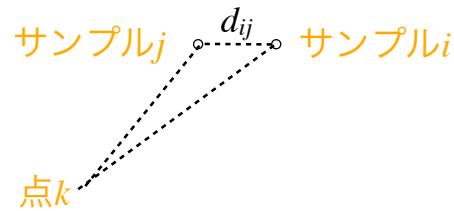
サンプル間の距離をまず計算する



この定理はサンプル*i,j*に対し、どこを原点（点*k*）としても成り立つ



この定理はサンプル*i,j*に対し、どこを原点（点*k*）としても成り立つ

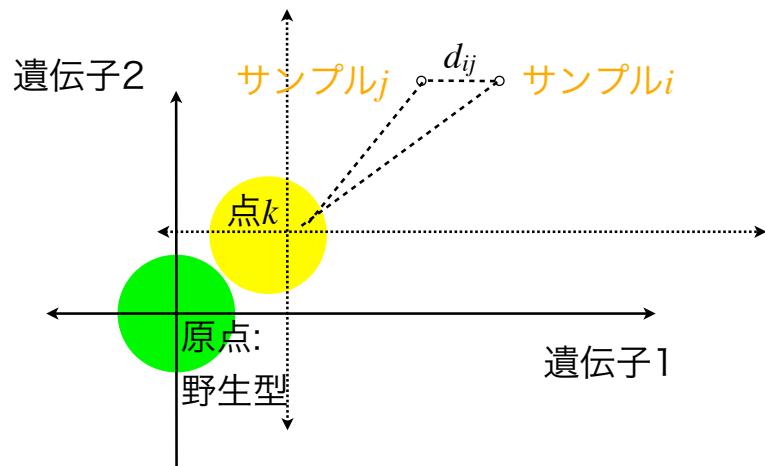


$$d_{ij}^2 = d_{ik}^2 + d_{jk}^2 - 2d_{ik}d_{jk} \cos\theta$$

この定理はサンプル*i,j*に対し、どこを原点（点*k*）としても成り立つ



例:入力データが野生型・変異体プロファイルの比であったら？

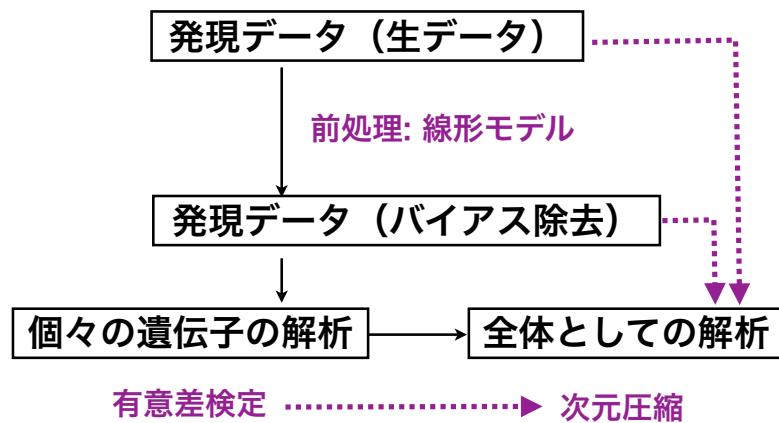


## 多変量解析(2)のまとめ

### PCA/MDS

- データがもつ類似性を低次元で表現し、評価・可視化する
- 重心の置き方に違い: 入力データをどのように前処理するか

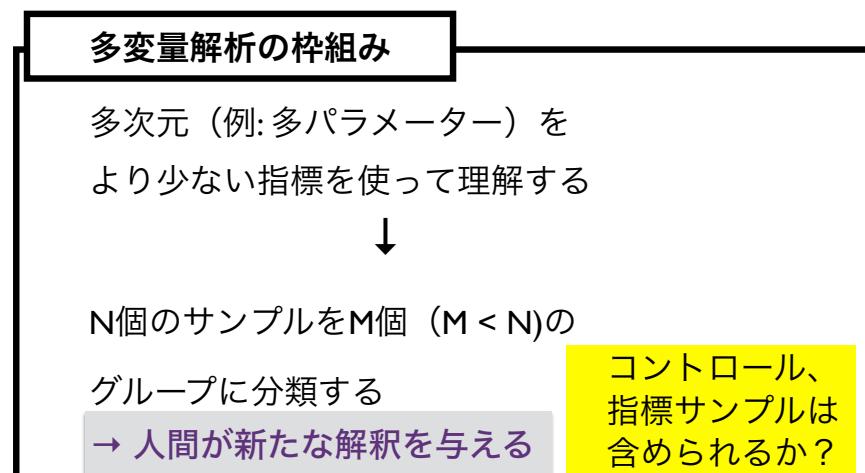
## 多変量解析をもう一歩進めて: 入力データは何を使うか?



## 多変量解析をもう一歩進めて: 研究の目的、実験デザイン、多変量解析

目的	実験デザイン	多変量解析
・何を知りたいか	・線形モデル	・入力データ前処理
明確に	・比較、因子	・距離尺度
・実施の制約	・検出力	・アルゴリズム
・予算		
・時間、労力		

## 多変量解析をもう一歩進めて: 人間の解釈をアシストするデータ取得を心がける



## 今回のトレーニングコースで 扱わなかった重要項目

- 確率分布
- 回帰、相関
- 線形モデルにおける交互作用
- 非線形クラスタリング・次元圧縮
- self-organization mapなど

# Gene Ontology解析

Shuji Shigenobu  
重信 秀治

基礎生物学研究所  
生物機能解析センター



## What is Gene Ontology (GO)?

- ▶ GO project describes gene products from all organisms using a consistent and computable language.
- ▶ GO produces sets of explicitly defined, structured vocabularies in both a computer- and human-readable manner.
- ▶ 3 categories
  - ▶ Biological processes
  - ▶ Molecular functions
  - ▶ Cellular components
- ▶ 2 components
  - ▶ Ontology: term definition terms and the structured relationships between them
  - ▶ Associations between gene products and the GO terms.

# Two components of GO

- ▶ Ontology
- ▶ Gene associations

**Gene Ontology Consortium**

Search GO data

Search for terms and gene products...

**Search**

**Ontology**

Filter classes [Download ontology](#)

Gene Ontology: the framework for the model of biology. The GO defines concepts/classes used to describe gene function, and relationships between these concepts. It classifies functions along three aspects:

**molecular function**  
molecular activities of gene products

**cellular component**  
where gene products are active

**biological process**  
pathways and larger processes made up of the activities of multiple gene products.

[more](#)

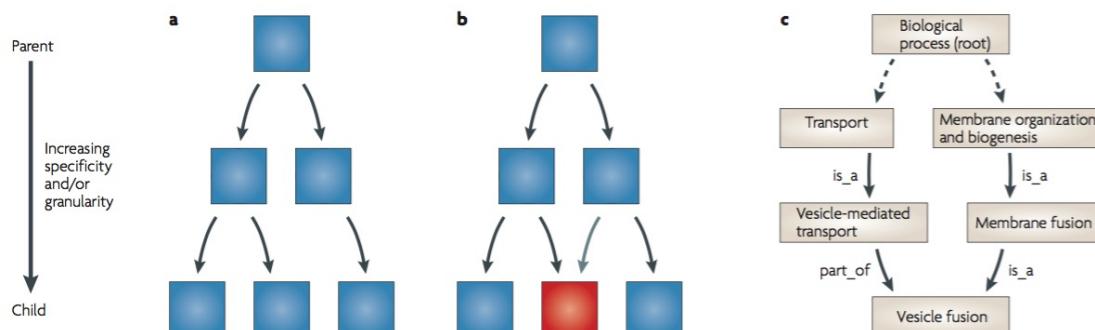
**Annotations**

[Download annotations \(standard files\)](#)  
[Filter and download \(customizable files <100k lines\)](#)

GO annotations: the model of biology. Annotations are statements describing the functions of specific genes, using concepts in the Gene Ontology. The simplest and most common annotation links one gene to one function, e.g. FZD4 + Wnt signaling pathway. Each statement is based on a specified piece of evidence. [more](#)

# Ontology structure

- ▶ Ontologies are represented as a directed acyclic graph (DAG).
- ▶ Parent-child relationship
  - ▶ `is_a`
  - ▶ `part_of`
- ▶ Ontology can be changed / updated



Rhee et al., 2008

# vesicle fusion

## Term Information ?

**Accession** GO:0006906  
**Name** vesicle fusion  
**Ontology** biological\_process  
**Synonyms** None  
**Alternate IDs** None  
**Definition** Fusion of the membrane of a transport vesicle with its target membrane. Source: GOC;jid  
**Comment** None  
**History** See term [history for GO:0006906](#) at QuickGO  
**Subset** None  
**Related**

- [Link](#) to all **genes and gene products** annotated to vesicle fusion.
- [Link](#) to all direct and indirect **annotations** to vesicle fusion.
- [Link](#) to all direct and indirect **annotations download** (limited to first 10,000) for vesicle fusion.

Data health

[Annotations](#)   [Graph Views](#)   [Inferred Tree View](#)   [Neighborhood](#)   [Mappings](#)

P GO:0008150 biological\_process
 

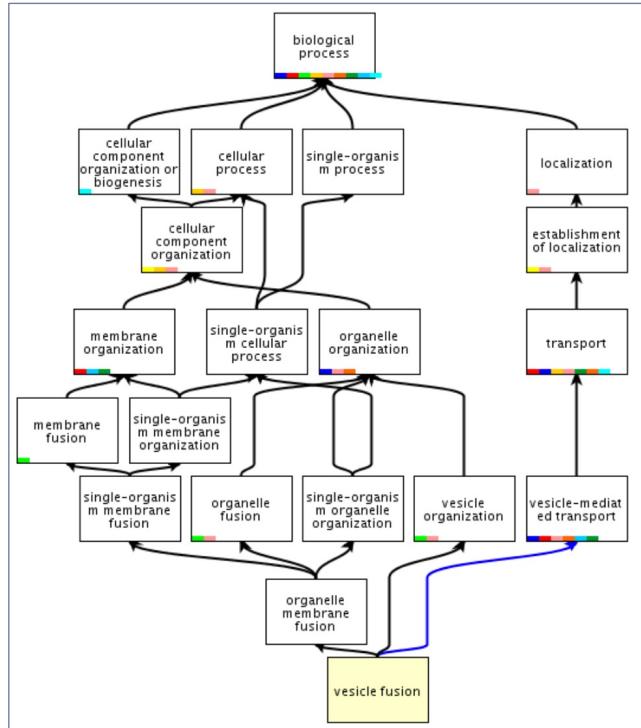
- I GO:0071840 cellular component organization or biogenesis
- I GO:0009987 cellular process
- I GO:0016043 cellular component organization
- I GO:0044699 single-organism process
- P GO:0051179 localization
- I GO:0061024 membrane organization
- I GO:0044763 single-organism cellular process
- P GO:0051234 establishment of localization
- I GO:0061025 membrane fusion
- I GO:0006996 organelle organization
- I GO:0044802 single-organism membrane organization
- I GO:0048284 organelle fusion

<http://amigo.geneontology.org/amigo/term/GO:0006906>

P GO:0008150 biological\_process
 

- I GO:0071840 cellular component organization or biogenesis
- I GO:0009987 cellular process
- I GO:0016043 cellular component organization
- I GO:0044699 single-organism process
- P GO:0051179 localization
- I GO:0061024 membrane organization
- I GO:0044763 single-organism cellular process
- P GO:0051234 establishment of localization
- I GO:0061025 membrane fusion
- I GO:0006996 organelle organization
- I GO:0044802 single-organism membrane organization
- I GO:0048284 organelle fusion
- I GO:0044801 single-organism membrane fusion
- I GO:1902589 single-organism organelle organization
- P GO:0006810 transport
- I GO:0090174 organelle membrane fusion
- I GO:0016050 vesicle organization
- P GO:0016192 vesicle-mediated transport
- ▼ **GO:0006906 vesicle fusion**
  - I GO:0034058 endosomal vesicle fusion
  - I GO:0048210 Golgi vesicle fusion to target membrane
  - P GO:0031339 negative regulation of vesicle fusion
  - I GO:0090385 phagosome-lysosome fusion
  - A GO:0031340 positive regulation of vesicle fusion
  - R GO:0031338 regulation of vesicle fusion
  - [capable\_of part of relation] GO:0031201 SNARE complex
  - P GO:0035493 SNARE complex assembly
  - I GO:0099500 vesicle fusion to plasma membrane
  - I GO:0048279 vesicle fusion with endoplasmic reticulum
  - I GO:1990668 vesicle fusion with endoplasmic reticulum-Golgi interface
  - I GO:0048280 vesicle fusion with Golgi apparatus
  - I GO:1990670 vesicle fusion with Golgi cis cisterna membrane
  - I GO:0007086 vesicle fusion with nuclear membrane involved in
  - I GO:0019817 vesicle fusion with peroxisome
  - I GO:0051469 vesicle fusion with vacuole
  - I GO:0061782 vesicle fusion with vesicle

membrane



# Gene association

- ▶ Gene <=> GO
- ▶ A gene may associate with multiple GO terms.
- ▶ Evidence codes.

Evidence code	Evidence code description	Source of evidence	Manually checked
IDA	Inferred from direct assay	Experimental	Yes
IEP	Inferred from expression pattern	Experimental	Yes
IGI	Inferred from genetic interaction	Experimental	Yes
IMP	Inferred from mutant phenotype	Experimental	Yes
IPI	Inferred from physical interaction	Experimental	Yes
ISS	Inferred from sequence or structural similarity	Computational	Yes
RCA	Inferred from reviewed computational analysis	Computational	Yes
ICC	Inferred from genomic context	Computational	Yes
IEA	Inferred from electronic annotation	Computational	No
IC	Inferred by curator	Indirectly derived from experimental or computational evidence made by a curator	Yes
TAS	Traceable author statement	Indirectly derived from experimental or computational evidence made by the author of the published article	Yes
NAS	Non-traceable author statement	No 'source of evidence' statement given	Yes
ND	No biological data available	No information available	Yes
NR	Not recorded	Unknown	Yes

## nanos

Gene Product Information [?](#)

[http://amigo.geneontology.org/amigo/gene\\_product/  
FB:FBgn0002962](http://amigo.geneontology.org/amigo/gene_product/FB:FBgn0002962)

Symbol	nos	Name(s)	nanos							
Total annotations: 29; showing: 1-10	<a href="#">«First</a>	<a href="#">&lt;Prev</a>	<a href="#">Next&gt;</a>	<a href="#">Last»</a>	<a href="#">Download (up to 100000)</a>					
Results count <input type="text" value="10"/> <a href="#">▼</a>										
Gene/product name	Annotation qualifier	GO class (direct)	Annotation extension	Contributor	Organism	Evidence	Evidence with	PANTHER family	Isoform	Reference
<a href="#">nos</a>	nanos	germ cell migration		FlyBase	Drosophila melanogaster	TAS		nanos protein pthr12887		<a href="#">FB:Brf0107500</a> <a href="#">PMID:9988212</a>
<a href="#">nos</a>	nanos	oogenesis		FlyBase	Drosophila melanogaster	IMP		nanos protein pthr12887		<a href="#">FB:Brf0107609</a> <a href="#">PMID:10101171</a>
<a href="#">nos</a>	nanos	spermatogenesis		FlyBase	Drosophila melanogaster	IMP		nanos protein pthr12887		<a href="#">FB:Brf0107609</a> <a href="#">PMID:10101171</a>
<a href="#">nos</a>	nanos	pole plasm		FlyBase	Drosophila melanogaster	TAS		nanos protein pthr12887		<a href="#">FB:Brf0110978</a> <a href="#">PMID:10449356</a>
<a href="#">nos</a>	nanos	anterior/posterior axis specification, embryo		FlyBase	Drosophila melanogaster	TAS		nanos protein pthr12887		<a href="#">FB:Brf0111327</a> <a href="#">PMID:10494038</a>
<a href="#">nos</a>	nanos	oocyte anterior/posterior axis specification		FlyBase	Drosophila melanogaster	NAS		nanos protein pthr12887		<a href="#">FB:Brf0128774</a> <a href="#">PMID:10878576</a>
<a href="#">nos</a>	nanos	protein binding		FlyBase	Drosophila melanogaster	IPI	<a href="#">FB:FBgn0000392</a>	nanos protein pthr12887		<a href="#">FB:Brf0131417</a> <a href="#">PMID:11060247</a>
<a href="#">nos</a>	nanos	germ-line stem cell division		FlyBase	Drosophila melanogaster	NAS		nanos protein pthr12887		<a href="#">FB:Brf0132358</a> <a href="#">PMID:11131516</a>
<a href="#">nos</a>	nanos	protein binding		UniProt	Drosophila melanogaster	IPI	<a href="#">FB:FBgn0010300</a>	nanos protein pthr12887		<a href="#">FB:Brf0135777</a> <a href="#">PMID:11274060</a>
<a href="#">nos</a>	nanos	female meiosis chromosome segregation		FlyBase	Drosophila melanogaster	IMP		nanos protein pthr12887		<a href="#">FB:Brf0135802</a> <a href="#">PMID:11290718</a>

## How to annotate GO for non-model organisms?

- ▶ Ortholog grouping with a model organism and then transfer the GO terms from the reference organism to your target organism.
- ▶ BLAST2GO
- ▶ InterProScan

## Gene Ontology enrichment analysis

- ▶ What is GO enrichment analysis?
- ▶ Why GO enrichment analysis is required in DEG studies?
- ▶ Type of GO enrichment analysis.
  - ▶ gene set
  - ▶ gene score
- ▶ Software
  - ▶ gene set type: DAVID (web), metascape (web), goseq (R), GOstat (R)
  - ▶ gene score: GSEA, roast, camera
  - ▶ both: ErmineJ

## Basic over-representation test: 2 x 2 table and Fisher's exact test

- ▶ Suppose we perform a test of DE and find a list of 200 significant genes out of 10,000
- ▶ Consider a specific GO term, apoptosis. Among the 200 DE genes, 20 genes are annotated as apoptosis related, while 300 / 10,000 are associated with apoptosis in the whole gene set.
- ▶ Question: Is the gene set “apoptosis” over-represented among “significant” genes?

	apoptosis	non-apoptosis	total
DE	20		200
non-DE			
total	300		10,000

	apoptosis	non-apoptosis	total
DE	20	180	200
non-DE	280	9,520	9,800
total	300	9,700	10,000

```
> mat <- matrix(c(20,200-20,300-20, 10000-300-(200-20)), nrow=2, byrow=T)
> fisher.test(mat, alternative="greater")

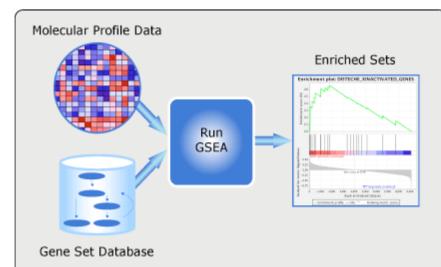
Fisher's Exact Test for Count Data

data: mat
p-value = 2.269e-06
alternative hypothesis: true odds ratio is greater than 1
95 percent confidence interval:
2.418508      Inf
sample estimates:
odds ratio
3.777069
```

## Gene score type enrichment analysis

- ▶ Drawback of basic 2x2 table method
  - ▶ Threshold value is arbitral
  - ▶ Magnitude of significance is ignored

- ▶ GSEA
  - ▶ <http://software.broadinstitute.org/gsea/index.jsp>
- ▶ ROAST, CAMERA
  - ▶ implemented within edgeR



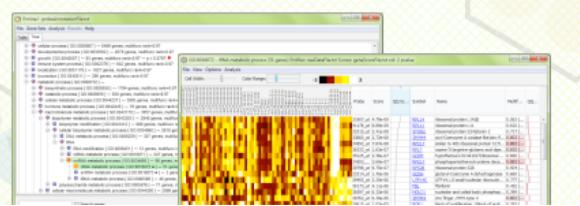
# Tutorial: ErmineJ

- ▶ <http://erminej.chibi.ubc.ca/>



**ermineJ**

Gene set analysis software



- ▶ Easy to use Java software with both GUI and CUI
- ▶ Three enrich methods supported
  - ▶ ORA: overrepresentation analysis
  - ▶ GSR: gene score resampling
  - ▶ ROC: rank-based gene score in receiver-operator curves

## ex11

Shuji Shigenobu edited this page 4 hours ago · 3 revisions

## ex11: ErmineJ tutorial

ErmineJを使って、ex5のシロイヌナズナの細菌感染実験のRNA-seq解析によって得られた、感染によって発現が有意に上昇する遺伝子群の、GO enrichment解析を行う。

### Files

(ファイルは ~/data/SS/ErmineJ 以下にある)

- arab2.deg.txt : edgeRのdeg解析の結果 (ex5)
- arab2.normalized.cpm.txt : edgeRでTMM法でノーマライズしたカウントデータ (count per million) (ex5)
- go\_daily-termdb.rdf-xml.gz : Gene Ontology の定義ファイル。GeneOntologyコンソーシアムのホームページからダウンロード。
- gene\_association.tair.ej.txt : シロイヌナズナ全遺伝子のGOアノテーションファイル。GeneOntologyコンソーシアムで提供されている、gene\_association.tair.gz をもとに、今回のトレーニングコース用にErmineJ専用のフォーマットに整形した。

```
[arab2.deg.txt]
id          logFC           logCPM        PValue
FDR
AT2G19190    4.54846288855724    7.3891028061101  1.62416309808535e-21
4.25871805948959e-17
AT4G12500    4.39110651359378    10.4369475637278
1.99859791508663e-20   2.45199569369715e-16
AT2G44370    5.43556520055014    5.2123645146869  3.21677527683902e-20
2.45199569369715e-16
AT3G46280    4.92503598519659    8.1242805484535  4.40769774863598e-20
2.45199569369715e-16
...
...
```

```
[gene_association.tair.ej.txt]
AT1G01110    IQD18    AT1G01110|IQD18|IQ-domain 18|T25K16.10|T25K16_10
G0:0003674
AT1G01420    UGT72B3 AT1G01420|UGT72B3|UDP-glucosyl transferase
72B3|F6F3_22|F6F3_22 G0:0008152|G0:0008194|G0:0016757|G0:0080043|G0:0080044
...
...
```

▼ Pages 17

Find a Page...

[Home](#)

[case1](#)

[case2](#)

[ex1](#)

[ex10](#)

[ex11](#)

[ex12](#)

[ex2](#)

[ex3](#)

[ex4](#)

[ex5](#)

[ex6](#)

[ex8](#)

[ex9](#)

[exercises](#)

[Show 2 more pages...](#)

[Clone this wiki locally](#)

<https://github.com/nibb-> 

[!\[\]\(f4520d842dd1517cf327cf7e4faf3240\_img.jpg\) Clone in Desktop](#)

### ErmineJの起動

ErmineJのHP から、WebStart でErmineJを起動する。(Run ermineJ now with WebstartセクションのClick "here" をクリック; セキュリティ等の理由で自動で起動しない場合は、Controlキーを押しながら起動。)

### GeneOntology 設定ファイルのロード

ErmineJを起動すると、GeneOntologyの定義ファイルと、遺伝子アノテーションファイルを指定するよう促される。

- Gene Ontology XML file => go\_daily-termdb.rdf-xml.gz を指定する。
- Gene annotation file => gene\_association.tair.ej.txt を指定する。フォーマットはErmineJ。

ロードが正常に完了すると以下ののようなテーブルのウィンドウが表示される。

Name	Description	Size	MultiFunc
GO:0000002	mitochondrial genome maintenance	6	0.32
GO:0000003	reproduction	1607 [ 1658 ]	0.95
GO:0000009 •	single-stranded DNA binding	2	0.06
GO:0000012	single-strand break repair	2	0.03
GO:0000014	single-stranded DNA endonuclease/nuclelease activity	8	0.43
GO:0000015 •	single-stranded DNA nucleic acid complex	3	0.12
GO:0000018 •	regulation of DNA recombination	12	0.63
GO:0000023	maltoose metabolic process	4	0.27
GO:0000024 •	maltoose metabolic process	2	0.18
GO:0000027	ribosomal large subunit assembly	23 [ 24 ]	0.46
GO:0000028	ribosomal small subunit assembly	10 [ 12 ]	0.49
GO:0000029	ribosomal subunit assembly	19	0.55
GO:0000032 •	cell wall mannoprotein biosynthetic process	2	0.35
GO:0000035	acyl binding	9	0.51
GO:0000036 •	acyl-coenzyme A acyltransferase attachment site binding involved in fatty acid biosynthesis	31 [ 33 ]	0.61
GO:0000038	very long-chain fatty acid metabolic process	79 [ 85 ]	0.66
GO:0000041	transition metal ion transport	9 [ 10 ]	0.83
GO:0000042 •	zinc ion transmembrane assembly	17	0.57
GO:0000049	tRNA binding	2	0.23
GO:0000050 •	urea cycle	13	0.64
GO:0000053	ribosomal subunit export from nucleus	4	0.20
GO:0000061	protein import into nucleus, translocation	8 [ 9 ]	0.54
GO:0000062	translocation across membrane	6	0.19
GO:0000064	L-ornithine transmembrane transporter activity	3	0.36
GO:0000066 •	mitochondrial ornithine transporter	2	0.16
GO:0000070	mitochondrial protein segregation	22	0.84
GO:0000073	cell cycle checkpoint	17	0.79
GO:0000075	DNA replication	3	0.25
GO:0000077	DNA synthesis	4	0.57
GO:0000079 •	regulation of cyclin-dependent protein serine/threonine kinase activity	26	0.78
GO:0000082	G1/S transition of mitotic cell cycle	10	0.65
GO:0000084 •	G1/S transition of cell cycle	3	0.14
GO:0000093	S-adenosyl-L-methionine transmembrane transporter activity	3	0.12
GO:0000096	sulfur amino acid metabolic process	59 [ 60 ]	0.87
GO:XXXXXX9		26 [ 26 ]	0.46

Search genes    Ready.

## Gene Ontology enrichment analysis (ORA)

ウィンドウ上部のAnalysisメニューから、Run Analysis を選択。"ORA" を選択。

Gene score file => arab2.deg.txt を指定。このファイルの5列目のFDRを今回の解析の「スコア」に使うので、Column = 5 を指定する。

Data profiles file => arab2.normalized.cpm.txtを指定。First data column = 2。

Select the aspects to include in the analysis の選択画面では、ここでは、3つのカテゴリーすべてをチェック。

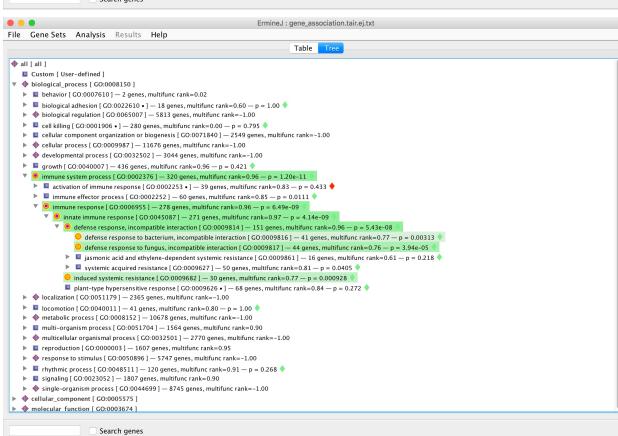
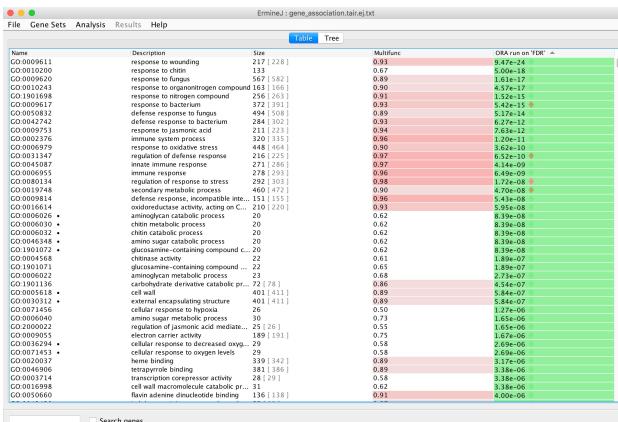
Maximum gene set size = 500, Minimum gene set size = 5, Gene replicate treatment はどちらでもよい（今回のデータセットでは意味がない）。

Adjust settings specific for your analysis method の選択画面では、

- Take the negative log of the gene scores をチェック。
- Large scores in your gene score file are better はチェックしない。
- Gene score threshold = 0.05 (thresholdを入力すると下部ウィンドウに何個の遺伝子が含まれるかが表示されるので便利)

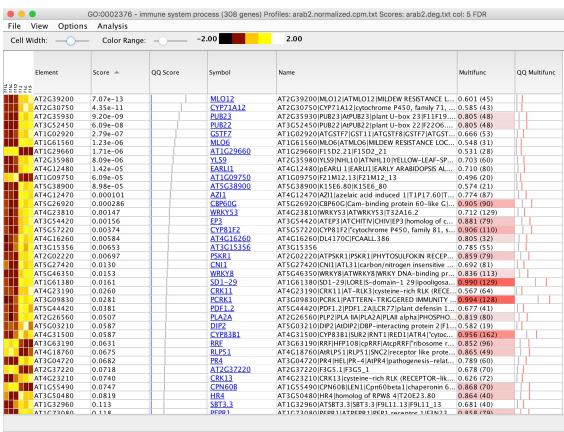
Finishボタンを押すと計算が始まる。

(結果例)



テーブルの興味のあるGOを右クリックし、Find this set in the tree panel を選択すると、Tree表示にかわる。

テーブルの興味のあるGOをダブルクリックすると、該当カテゴリーに含まれる遺伝子の発現プロファイルが表示される。



計算結果をテキストファイルで保存する。

- メニュー : Analysis > Save Analysis
- 保存したい解析を選択し、"Include all genes in output" をチェックして "OK"ボタン。

## Gene Ontology enrichment analysis (GSR)

ORAとほぼ同じ手順。Create New AnalysisのStep 1 of 5 でGSRを選択。

ORAの結果と比べてみよう。

### 発展

- 上の実習では、edgeRで得られたFDRをスコアとして使い、閾値を0.05に設定した。この場合、細菌感染によってup-regulated, down-regulated 両方の遺伝子を含むことになる。up-regulatedなものだけを使いたい場合はどうすればよいだろうか？また、fold-changeの値も考慮して、「2倍以上up-regulated」なDEGだけを調べたいケースも多い。このような場合はどのようにすればよいだろうか？