Introduction to Bioinformatics

Week 11. RNA-seq Tutorial

11/10 (월) 3-4교시

Dayeon Kim
dayeonkim@hufs.ac.kr
Hankuk University of Foreign Studies
Division of Biomedical Engineering

Systems Biology Lab

실습 준비

- R 4.0 이하버전에서는 CummeRbund 지원하지 않음

➔ R 4 버전을 로컬에 설치하여 사용가능!
참고자료(https://blog.naver.com/songsite123/223334808151)

https://github.com/dayeon24/RNAseq_anlaysis

➔ **diff_out.zip** (cuffdiff 최종 결과 파일)

➔ 실습 ppt 다운로드

Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks

Cole Trapnell[1,2], Adam Roberts[3], Loyal Goff[1,2,4], Geo Pertea[5,6], Daehwan Kim[5,7], David R Kelley[1,2], Harold Pimentel[3], Steven L Salzberg[5,6], John L Rinn[1,2] & Lior Pachter[3,8,9]

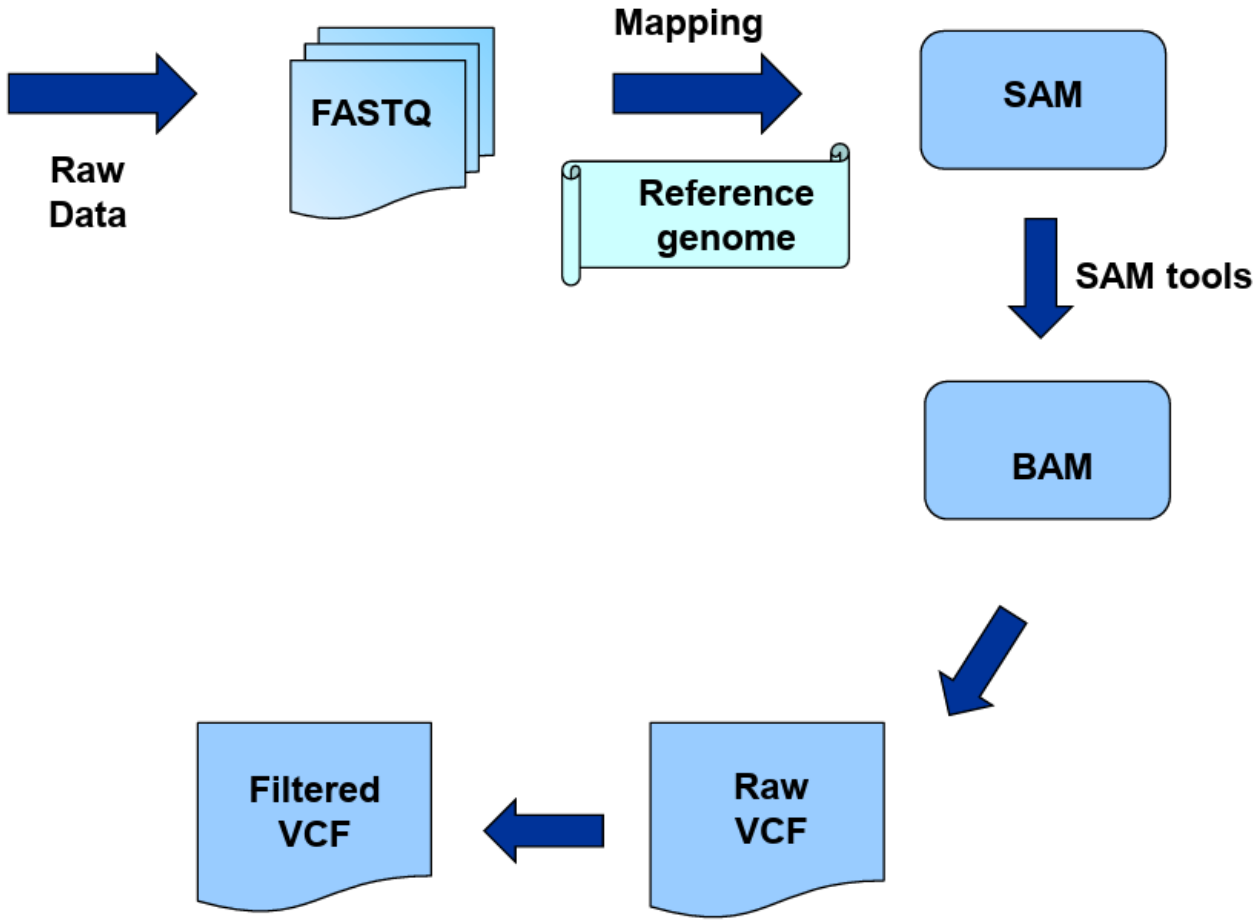
[1]Broad Institute of MIT and Harvard, Cambridge, Massachusetts, USA. [2]Department of Stem Cell and Regenerative Biology, Harvard University, Cambridge, Massachusetts, USA. [3]Department of Computer Science, University of California, Berkeley, California, USA. [4]Computer Science and Artificial Intelligence Lab, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA. [5]Department of Medicine, McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University School of Medicine, Baltimore, Maryland, USA. [6]Department of Biostatistics, Johns Hopkins University, Baltimore, Maryland, USA. [7]Center for Bioinformatics and Computational Biology, University of Maryland, College Park, Maryland, USA. [8]Department of Mathematics, University of California, Berkeley, California, USA. [9]Department of Molecular and Cell Biology, University of California, Berkeley, California, USA. Correspondence should be addressed to C.T. (cole@broadinstitute.org).

Published online 1 March 2012; corrected after print 7 August 2014; doi:10.1038/nprot.2012.016

Recent advances in high-throughput cDNA sequencing (RNA-seq) can reveal new genes and splice variants and quantify expression genome-wide in a single assay. The volume and complexity of data from RNA-seq experiments necessitate scalable, fast and mathematically principled analysis software. TopHat and Cufflinks are free, open-source software tools for gene discovery and comprehensive expression analysis of high-throughput mRNA sequencing (RNA-seq) data. Together, they allow biologists to identify new genes and new splice variants of known ones, as well as compare gene and transcript expression under two or more conditions. This protocol describes in detail how to use TopHat and Cufflinks to perform such analyses. It also covers several accessory tools and utilities that aid in managing data, including CummeRbund, a tool for visualizing RNA-seq analysis results. Although the procedure assumes basic informatics skills, these tools assume little to no background with RNA-seq analysis and are meant for novices and experts alike. The protocol begins with raw sequencing reads and produces a transcriptome assembly, lists of differentially expressed and regulated genes and transcripts, and publication-quality visualizations of analysis results. The protocol's execution time depends on the volume of transcriptome sequencing data and available computing resources but takes less than 1 d of computer time for typical experiments and ~1 h of hands-on time.

2012, Nature protocols, 13733 citation

RNA-seq analysis



Bowtie
Extremely fast, general purpose short read aligner

TopHat
Aligns RNA-Seq reads to the genome using Bowtie
Discovers splice sites

Cufflinks package

Cufflinks
Assembles transcripts

Cuffcompare
Compares transcript assemblies to annotation

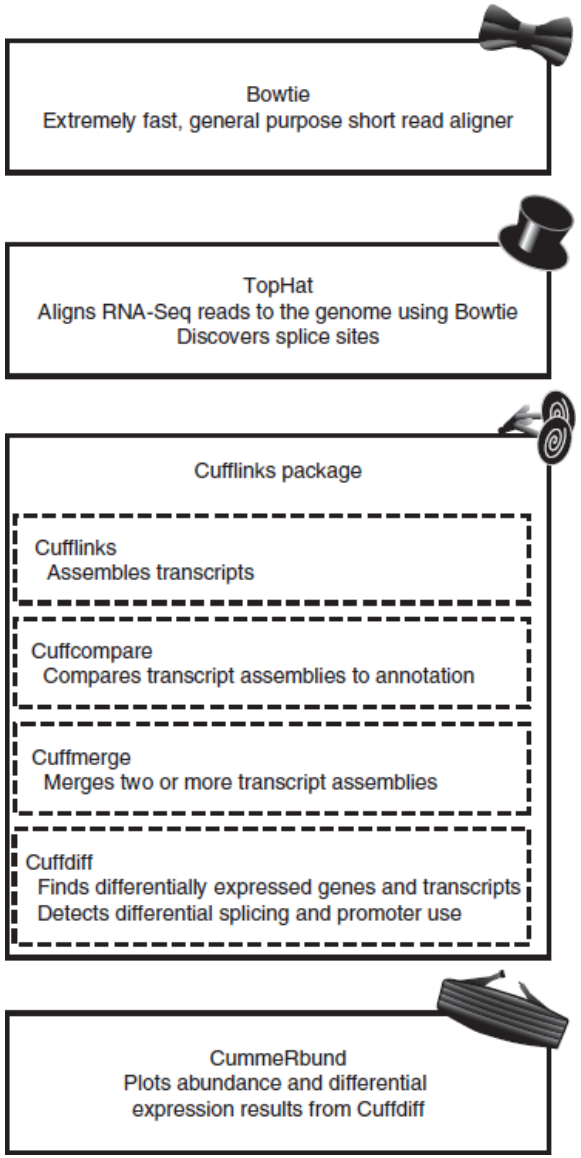
Cuffmerge
Merges two or more transcript assemblies

Cuffdiff
Finds differentially expressed genes and transcripts
Detects differential splicing and promoter use

CummeRbund
Plots abundance and differential expression results from Cuffdiff

RNA-seq analysis

1. Align sequencing reads to the reference genome
2. Assemble aligned reads into full-length transcripts
3. Quantify the expression levels of each transcript
4. Compare expression levels across experimental conditions



실습 목표

초파리 시뮬레이션 데이터

	Control case (C1)	Perturbed case (C2)
R1 (초파리 A)	C1_R1_1.fq, C1_R1_2.fq	C2_R1_1.fq, C2_R1_2.fq
R2 (초파리 B)	C1_R2_1.fq, C1_R2_2.fq	C2_R2_1.fq, C2_R2_2.fq
R3 (초파리 C)	C1_R3_1.fq, C1_R3_2.fq	C2_R3_1.fq, C2_R3_2.fq

_1 paired-end의 forward read

_2 paired-end의 reverse read

“Control(C1)과 비교했을 때,
Perturbed(C2)에서 발현량이 달라진 유전자 찾기”

실습 환경

- 터미널 혹은 putty에서 접속 **ssh -p 35790 bioinfo25@sysbio.hufs.ac.kr**
- 비번 **bioinfo1110**

데이터 위치 /home/bioinfo25/DATA
Software 설치 방법 supplementary 페이지 참고

1. 개인 디렉토리 생성 후, 해당 디렉토리에서 이용할 것:
   mkdir dykim; cd dykim
2. 데이터 다운로드: WinSCP 소프트웨어를 이용하거나 scp 명령어 이용
   참고자료) https://eehoeskrap.tistory.com/543
3. 본인 데이터 외 **데이터 삭제하지 않도록 주의 !**

실습 데이터

- .fq : fastaq 파일

```
@1
GCTTGAATATTGACTCTTCGCGTAGATATTTCTCTAGAGCGTCCAGCACTGGTCGGTTTGGCTTTCCCTTTGAGT
+
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
@2
AATCCTTGCACCGTCGTTTTGCCAGCTGCGTCAACTTAAATGATGACTTCGCCGAGCAATTTAAAGCAAGAGCGG
+
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
```

read ID
시퀀스
+
품질 점수

- .gz 압축된 파일, 그냥 읽을 수 없음
  - gunzip 으로 압축을 풀거나 gcat 으로 파일을 읽어야 함.

문자 `"I"`의 ASCII 코드는 **73**

→ Phred+33 체계에서는

$$Q = 73 - 33 = 40$$

$$P = 10^{(-Q/10)} = 10^{-4} = 0.0001$$

Q	P_error	ASCII
33	0.00050	66 B
34	0.00040	67 C
35	0.00032	68 D
36	0.00025	69 E
37	0.00020	70 F
38	0.00016	71 G
39	0.00013	72 H
40	0.00010	73 I
41	0.00008	74 J
42	0.00006	75 K

1. Align sequencing reads to the reference genome

Dayeon Kim(dayeonkim@hufs.ac.kr)

Align with TopHat

- RNA-seq 실험에서 얻은 short read(FASTQ)를 **reference genome**에 정렬

```
$ tophat -p 8 -G genes.gtf -o C1_R1_thout genome C1_R1_1.fq C1_R1_2.fq
$ tophat -p 8 -G genes.gtf -o C1_R2_thout genome C1_R2_1.fq C1_R2_2.fq
```

옵션	의미	목적
-p 8	병렬로 8개의 CPU 코어 사용	빠른 실행 (병렬처리)
-G genes.gtf	주어진 유전자 annotation 파일 사용	GTF 파일로 알려진 exon/splice 정보를 참조하여 보다 정확한 정렬
-o C1_R1_thout	결과(output) 저장 폴더 이름	sample별로 폴더를 다르게 지정
genome	✅참조 유전체 인덱스(prefix)	Bowtie로 미리 만든 인덱스 파일 세트
C1_R1_1.fq C1_R1_2.fq	✅입력 FASTQ 파일 (paired-end)	_1은 forward, _2는 reverse read

Align with TopHat - genome

- genome은 Bowtie2 인덱스 파일(binary)의 prefix 이름
- TopHat은 내부적으로 Bowtie를 호출하여 read를 인덱스에 매핑
  - 매우 긴 문자열로 이뤄진 fa 파일
  - read를 그대로 하나씩 비교하면 시간이 너무 오래 걸리기 때문에, 색인(index) 페이지를 이용해 필요한 위치를 바로 찾음.

genome.fa

```
>2L
CGACAATGCACGACAGAGGAAGCAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCC
ATATTATAGGGAGAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTTT
GATTTTTTGGCAACCCAAAATGGTGGCGGATGAACGAGATGATAATATATTCAAGTTGCC
GCTAATCAGAAATAAATTCATTGCAACGTTAAATACAGCACAATATATGATCGCGTATGC
GAGAGTAGTGCCAACATATTGTGCTAATGAGTGCCTCTCGTTCTCTGTCTTATATTACCG
CAAACCCAAAAAGACAATACACGACAGAGAGAGAGAGCAGCGGAGATATTTAGATTGCCT
ATTAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTCTATATAATGAC
TGCCTCTCATTCTGTCTTATTTTACCGCAAACCCAAATCGACAATGCACGACAGAGGAAG
CAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCCATATTATAGGGAGAAATATGAT
```

초파리 reference genome

Bowtie2

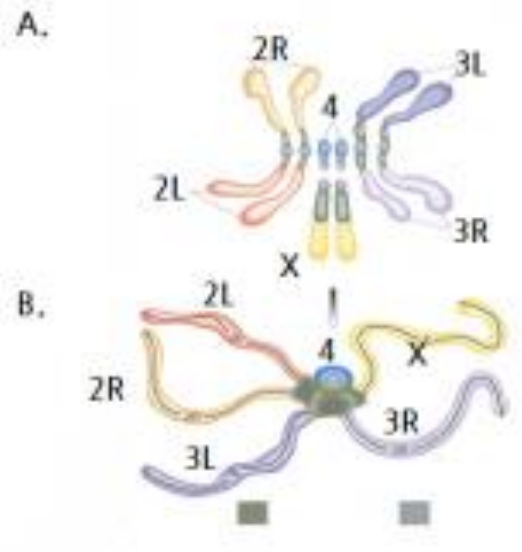
```
genome.1.bt2
genome.2.bt2
genome.3.bt2
genome.4.bt2
genome.rev.1.bt2
genome.rev.2.bt2
```

Align with TopHat – general feature format (gtf)

- .gtf : 유전자 주석 정보 파일(탭으로 구분됨)

chrom, source, feature, start, end, score, strand, frame, attribute

```
2L      protein_coding  exon    7529    8116    .       +       .       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG11023";
2L      protein_coding  exon    7529    8116    .       +       .       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG11023";
2L      protein_coding  CDS     7680    8116    .       +       0       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG11023";
2L      protein_coding  CDS     7680    8116    .       +       0       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG11023";
2L      protein_coding  start_codon     7680    7682    .       +       0       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG
2L      protein_coding  start_codon     7680    7682    .       +       0       exon_number "1"; gene_id "FBgn0031208"; gene_name "CG
```



Exon (7529–8116)
└——— Coding region (CDS: 7680–8116)
└——— Start codon (7680–7682)

Align with TopHat – 실습하기1

- 실습 준비

```
bioinfo25@sysbio:~$ mkdir dykim
bioinfo25@sysbio:~$ ls
anaconda3_activate.sh  DATA  dykim  OUT  TOOL
bioinfo25@sysbio:~$ cd dykim/
bioinfo25@sysbio:~/dykim$ ls
bioinfo25@sysbio:~/dykim$ tophat tophat -p 8 -G genes.gtf -o C1_R3_thout genome C1_R1_1.fq C1_R1_2.fq^C
bioinfo25@sysbio:~/dykim$ tophat -p 8 -G genes.gtf -o C1_R3_thout genome C1_R1_1.fq C1_R1_2.fq

[2025-11-09 13:30:53] Beginning TopHat run (v2.1.1)
-----------------------------------------
[2025-11-09 13:30:53] Checking for Bowtie
                  Bowtie version:        2.3.4.2
Error: cannot find transcript file genes.gtf
bioinfo25@sysbio:~/dykim$ ln -s ../DATA/* .
bioinfo25@sysbio:~/dykim$ ls
C1_R1_1.fq.gz  C1_R2_2.fq.gz  C1_R3_thout    C2_R2_1.fq.gz  C2_R3_2.fq.gz  genome.2.bt2  genome.fa
C1_R1_2.fq.gz  C1_R3_1.fq.gz  C2_R1_1.fq.gz  C2_R2_2.fq.gz  genes.gtf      genome.3.bt2  genome.rev.1.bt2
C1_R2_1.fq.gz  C1_R3_2.fq.gz  C2_R1_2.fq.gz  C2_R3_1.fq.gz  genome.1.bt2   genome.4.bt2  genome.rev.2.bt2
```

Align with TopHat – 실습하기2

- 코드 실행

```
$ tophat -p 8 -G genes.gtf -o C1_R1_thout genome C1_R1_1.fq.gz C1_R1_2.fq.gz
$ tophat -p 8 -G genes.gtf -o C1_R2_thout genome C1_R2_1.fq.gz C1_R2_2.fq.gz
```

```
IOError: [Errno 2] No such file or directory: 'C1_R1_1.fq'
bioinfo25@sysbio:~/dykim$ tophat -p 8 -G genes.gtf -o C1_R1_thout genome C1_R1_1.fq.gz C1_R1_2.fq.gz

[2025-11-09 13:32:27] Beginning TopHat run (v2.1.1)
-----------------------------------------------
[2025-11-09 13:32:27] Checking for Bowtie
                  Bowtie version:        2.3.4.2
[2025-11-09 13:32:27] Checking for Bowtie index files (genome)..
[2025-11-09 13:32:27] Checking for reference FASTA file
[2025-11-09 13:32:27] Generating SAM header for genome
```

Align with TopHat – 실습하기3

- 실행 결과

```
bioinfo25@sysbio:~$ ls
anaconda3_activate.sh  DATA  dykim  OUT  TOOL
bioinfo25@sysbio:~$ cd OUT/
bioinfo25@sysbio:~/OUT$ ls
C1_R1_clout  C1_R2_clout  C1_R3_clout  C2_R1_clout  C2_R2_clout  C2_R3_clout  tophat_example.log
C1_R1_thout  C1_R2_thout  C1_R3_thout  C2_R1_thout  C2_R2_thout  C2_R3_thout
bioinfo25@sysbio:~/OUT$ cd C1_R1_thout/
bioinfo25@sysbio:~/OUT/C1_R1_thout$ ls
accepted_hits.bam  deletions.bed   junctions.bed  prep_reads.info
align_summary.txt  insertions.bed  logs           unmapped.bam
```

Align with TopHat - output file

파일 이름	형식	주요 내용	다음 단계에서의 사용
accepted_hits.bam	BAM	성공적으로 매핑된 read들의 정렬 결과. 각 read의 염색체 위치, 방향, mapping quality 등	✅Cufflinks 입력으로 사용됨
align_summary.txt	텍스트	매핑 통계 요약	QC 및 보고용

```
Left reads:  < forward
          Input     :  11607353  < total reads
           Mapped   :  11607353 (100.0% of input)   < 매핑에 성공한 read 수(비율)
            of these:     60103 ( 0.5%) have multiple alignments (82 have >20)
Right reads:                                          ㄴ 여러 위치에 중복 매핑된 read 수 (비율)
          Input     :  11607353
           Mapped   :  11607352 (100.0% of input)
            of these:     60103 ( 0.5%) have multiple alignments (82 have >20)
100.0% overall read mapping rate.  < 전체적인 매핑 성공률

Aligned pairs:  11607352
     of these:     60103 ( 0.5%) have multiple alignments
                       8 ( 0.0%) are discordant alignments  < 예상되는 방향과 거리를 벗어나게 정렬된 read 수
100.0% concordant pair alignment rate.
```

2. Assembly of the alignments into full-length transcripts

Assemble with Cufflinks

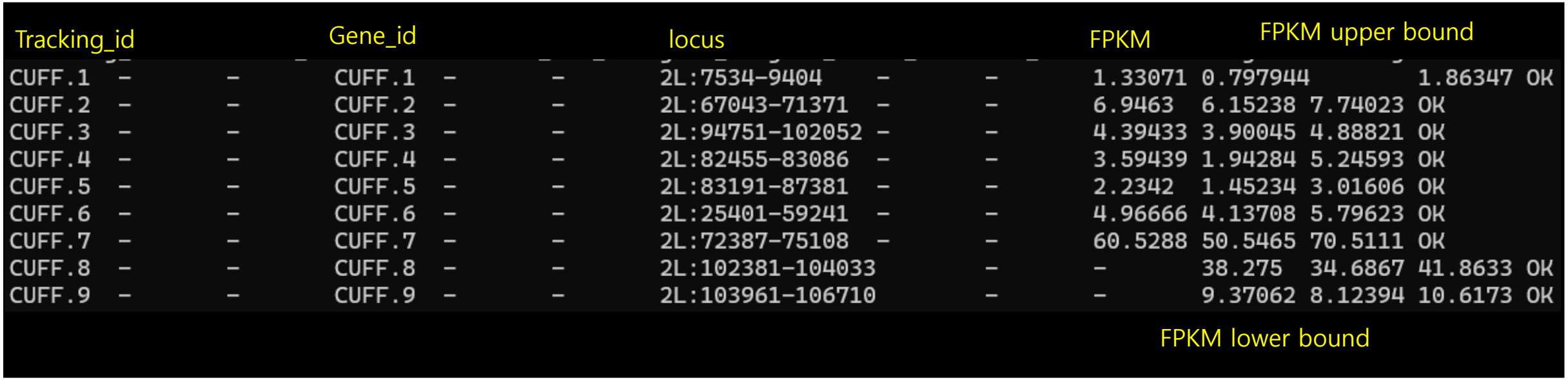
- 전사체(Transcript) 조립(assemble)과 발현량 계산을 수행하는 단계

```
$ cufflinks -p 8 -o C1_R1_clout C1_R1_thout/accepted_hits.bam
$ cufflinks -p 8 -o C1_R2_clout C1_R2_thout/accepted_hits.bam
```

옵션	의미	설명
-p 8	threads = 8	CPU 8개를 병렬로 사용 (속도 향상)
-o C1_R1_clout	output directory	결과를 저장할 폴더 이름 (sample별로 구분)
C1_R1_thout/accepted_hits.bam	✅Input BAM	TopHat이 만든 read 정렬 결과

Assemble with Cufflinks - output file

파일 이름	파일 형식	설명
transcripts.gtf	GTF	✅ Cufflinks가 조립한 **전사체(annotation)** 결과 각 유전자의 exon, transcript 구조를 예측한 파일.
genes.fpkm_tracking	텍스트 (TSV)	**유전자(gene)** 수준의 FPKM 발현량 테이블 파일. 각 gene의 평균, 표준편차, replicate별 값 포함.
isoforms.fpkm_tracking	텍스트 (TSV)	**전사체(isoform)**수준의 FPKM 발현량 테이블 파일. gene 내부의 여러 transcript들의 상대적 발현량 확인 가능.



Tracking_id			Gene_id			locus			FPKM			
CUFF.1	-	-	CUFF.1	-	-	2L:7534-9404	-	-	1.33071	0.797944	1.86347	OK
CUFF.2	-	-	CUFF.2	-	-	2L:67043-71371	-	-	6.9463	6.15238	7.74023	OK
CUFF.3	-	-	CUFF.3	-	-	2L:94751-102052	-	-	4.39433	3.90045	4.88821	OK
CUFF.4	-	-	CUFF.4	-	-	2L:82455-83086	-	-	3.59439	1.94284	5.24593	OK
CUFF.5	-	-	CUFF.5	-	-	2L:83191-87381	-	-	2.2342	1.45234	3.01606	OK
CUFF.6	-	-	CUFF.6	-	-	2L:25401-59241	-	-	4.96666	4.13708	5.79623	OK
CUFF.7	-	-	CUFF.7	-	-	2L:72387-75108	-	-	60.5288	50.5465	70.5111	OK
CUFF.8	-	-	CUFF.8	-	-	2L:102381-104033		-	-	38.275	34.6867	41.8633 OK
CUFF.9	-	-	CUFF.9	-	-	2L:103961-106710		-	-	9.37062	8.12394	10.6173 OK

FPKM

- Fragment Per Kilobase of transcript per Million mapped reads

$$\mathbf{FPKM} = \frac{C}{\left(\frac{L}{1000}\right)\left(\frac{N}{10^6}\right)}$$

- C: 그 feature(유전자/전사체)에 **할당된 fragment 수**
- L: feature의 **exon 길이 합**(bp) → kb로 환산
- N: 그 샘플의 **총 매핑된 fragment 수** → million으로 환산

3. Quantify the expression levels of each transcript

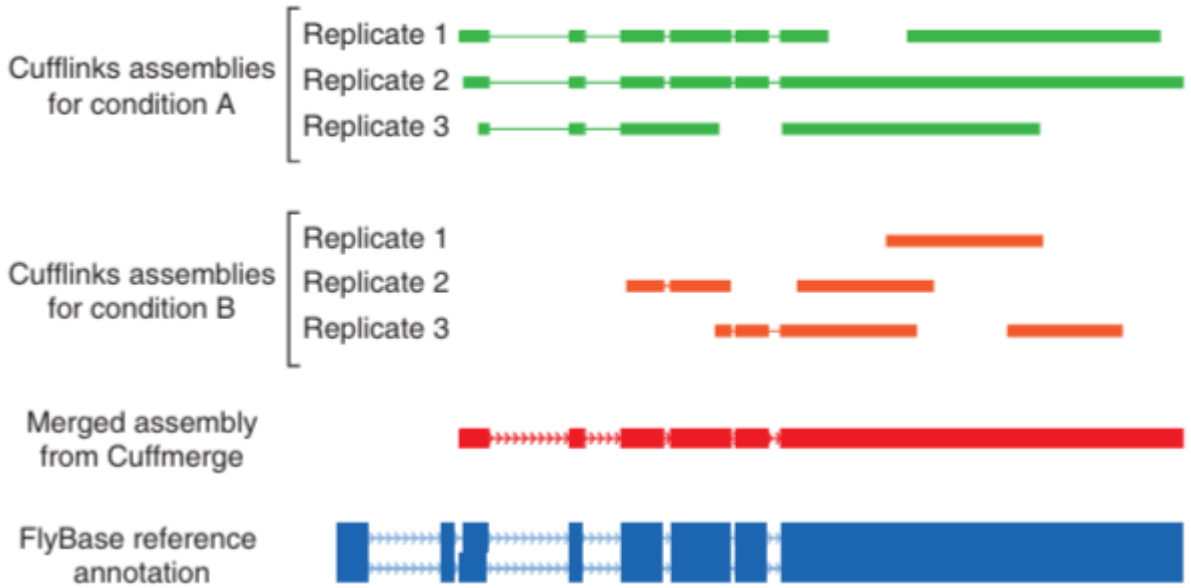
Dayeon Kim(dayeonkim@hufs.ac.kr)

Meta-Assembler Cuffmerge

- 각 샘플 transcripts.gtf 나열한 텍스트 파일 생성
- 하나의 gtf 파일로 통합(cuffmerge)

```
$ vi assemblies.txt
$ cuffmerge -g genes.gtf -s genome.fa -p 8 assemblies.txt
```

```
./C1_R1_clout/transcripts.gtf
./C2_R2_clout/transcripts.gtf
./C1_R2_clout/transcripts.gtf
./C2_R1_clout/transcripts.gtf
./C1_R3_clout/transcripts.gtf
./C2_R3_clout/transcripts.gtf
~
~
~
```



Meta-Assembler Cuffmerge - output file

파일 이름	파일 형식	설명
merged.gtf	GTF	여러 샘플의 transcript 조립 결과를 통합한 전사체(annotation). 기존 reference(genes.gtf)와 신규 조립 결과를 병합하여 “known + novel isoform”을 모두 포함.

chrom, source, feature, start, end, score, strand, frame, attribute

```
2L      Cufflinks       exon    7529    8116    .       +       .       gene_id "XLOC_000001"; transcript_id "TCONS_00000002"; exon_number "1"; gene_name "CG11023"; oId "FBtr030
0690"; nearest_ref "FBtr0300690"; class_code "="; tss_id "TSS1"; p_id "P1";
2L      Cufflinks       exon    8193    8589    .       +       .       gene_id "XLOC_000001"; transcript_id "TCONS_00000002"; exon_number "2"; gene_name "CG11023"; oId "FBtr030
0690"; nearest_ref "FBtr0300690"; class_code "="; tss_id "TSS1"; p_id "P1";
2L      Cufflinks       exon    8668    9484    .       +       .       gene_id "XLOC_000001"; transcript_id "TCONS_00000002"; exon_number "3"; gene_name "CG11023"; oId "FBtr030
0690"; nearest_ref "FBtr0300690"; class_code "="; tss_id "TSS1"; p_id "P1";
2L      Cufflinks       exon    7529    8116    .       +       .       gene_id "XLOC_000001"; transcript_id "TCONS_00000001"; exon_number "1"; gene_name "CG11023"; oId "FBtr030
0689"; nearest_ref "FBtr0300689"; class_code "="; tss_id "TSS1"; p_id "P2";
```

4. Compare expression levels across experimental conditions

Dayeon Kim(dayeonkim@hufs.ac.kr)

Identify DEG

- 두 조건(C1 vs C2) 사이에서 유전자와 전사체의 발현량 차이를 통계적으로 검정

```
$ cuffdiff -o diff_out -b genome.fa -p 8 –L C1,C2 -u merged_asm/merged.gtf \
./C1_R1_thout/accepted_hits.bam,./C1_R2_thout/accepted_hits.bam,./C1_R3_thout/  accepted_hits.bam \
./C2_R1_thout/accepted_hits.bam,./C2_R3_thout/accepted_hits.bam,./C2_R2_thout/accepted_hits.bam
```

옵션	의미	설명
-o diff_out	Output directory	결과 저장 폴더
-b genome.fa	Reference genome sequence	GC bias, fragment length 보정 등에 사용
-p 8	CPU 스레드 수	병렬 계산 (속도 향상)
-L C1,C2	Labels for conditions	두 조건 이름 (Control vs Treatment)
-u	Use reference annotation	병합된 annotation(merged.gtf)을 사용하여 known+novel isoform 기반 분석 수행
merged_asm/merged.gtf	✅Unified annotation	Cuffmerge에서 만든 통합 전사체 모델
각 샘플의 BAM 리스트	✅Input data	

Identify DEG – output file

파일	내용
gene_exp.diff	각 gene의 발현 차이(FPKM, log2 fold change, p value 등)
isoform_exp.diff	isoform(전사체) 단위의 차등 발현
tss_group_exp.diff	전사 시작점(TSS) 기반 차등 발현
cds_exp.diff	번역 가능 영역(CDS) 단위 차등 발현
promoters.diff, splicing.diff, cds.diff	스플라이싱·프로모터 사용 등 차등 조절 분석
tracking files (.fpkm_tracking)	각 샘플의 발현량 및 ID 대응 정보

test_id	gene_id	gene	locus	sample_1	sample_2	status	value_1	value_2	log2(fold_change)	test_stat	p_value	q_value	significant
XLOC_000001	XLOC_000001	CG11023	2L:7528-9484	C1	C2	OK	1.3722	1.16336	-0.238196	0.586034	0.557853	0.959101	no
XLOC_000002	XLOC_000002	dbr	2L:67043-71390	C1	C2	OK	6.77719	6.53965	-0.0514726	0.284668	0.775898	0.981194	no
XLOC_000003	XLOC_000003	galectin	2L:72387-76211	C1	C2	OK	48.748	78.1106	0.680177	-11.6091	0	0	yes
XLOC_000004	XLOC_000004	CG11374	2L:76445-77639	C1	C2	NOTEST	0.433448	0.270495	-0.680258	0.672943	0.500983	1	no
XLOC_000005	XLOC_000005	CG11376	2L:94751-102086	C1	C2	OK	4.12295	3.66655	-0.169252	0.918392	0.358414	0.900099	no

Identify DEG – output file

컬럼명	의미	설명
test_id	Cufflinks가 부여한 transcript(전사체) ID	각 transcript(또는 gene cluster)에 대한 내부 식별자
gene_id	Cufflinks가 부여한 gene-level ID	transcript들이 속한 유전자 그룹 ID (gene-level expression 시 동일)
gene	실제 annotation 상의 gene symbol 또는 이름	Ensembl, FlyBase, NCBI 등의 주석 기반 gene 이름
locus	유전자의 게놈 위치	염색체 번호와 시작–끝 위치 (예: chr2L 7528~9484)
sample_1	첫 번째 비교 그룹 이름	-L 옵션에서 지정한 첫 그룹 (예: Control)
sample_2	두 번째 비교 그룹 이름	-L 옵션에서 지정한 두 번째 그룹 (예: Treatment)
status	테스트 상태	OK는 정상적으로 테스트 수행, NOTEST는 데이터 부족 등으로 통계 테스트 불가
value_1	그룹 1의 발현량 (FPKM 또는 TPM)	C1의 평균 발현량 (Fragments per Kilobase of exon per Million mapped reads)
value_2	그룹 2의 발현량 (FPKM 또는 TPM)	C2의 평균 발현량
log2(fold_change)	로그2 스케일의 발현 차이	log2(value_2 / value_1) 값. 음수면 C2가 낮음, 양수면 높음
test_stat	통계 검정값 (t-statistic 등)	차등 발현 유무를 판단하기 위한 통계량
p_value	유의확률	두 그룹 간 차이가 우연일 확률 (낮을수록 의미 있음)
q_value	다중검정 보정된 p값 (FDR)	False Discovery Rate 보정 후의 p-value
significant	유의성 여부	q_value < 0.05이면 일반적으로 yes로 표시됨 (유의한 차등발현 유전자)

5. Explore differential analysis results

Dayeon Kim(dayeonkim@hufs.ac.kr)

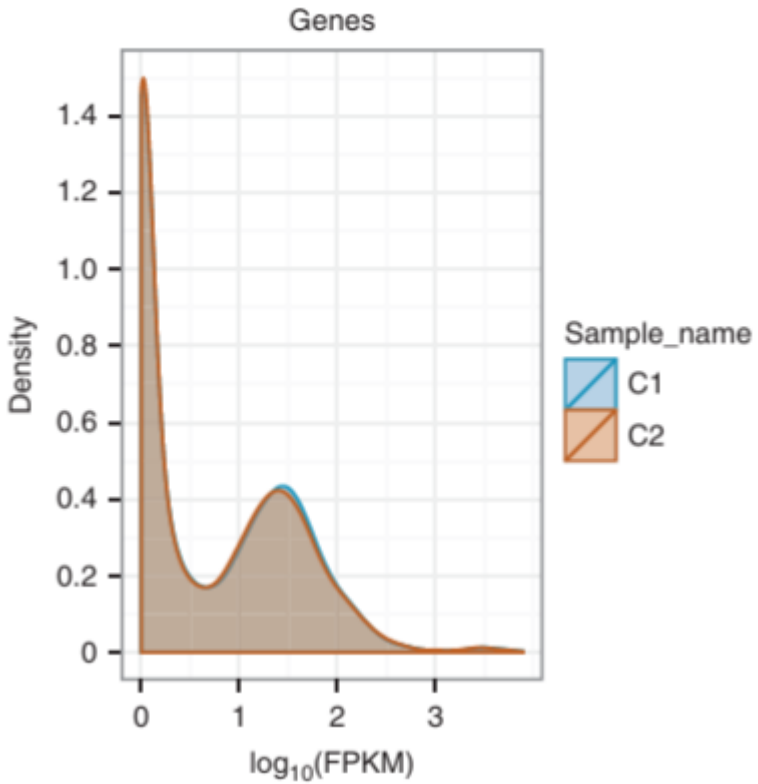
Downstream analysis with CummeRbund

```
#Load the CummeRbund package into R environment:
>if (!requireNamespace("BiocManager", quietly = TRUE))
+    install.packages("BiocManager")

>BiocManager::install("cummeRbund") library(cummeRbund)

#Create a CummeRbund database from the cuffdiff ouput:
>cuff_data<-readCufflinks('diff_out')
```

Downstream analysis with CummeRbund

- Plot the distribution of expression levels for each sample (Fig. 6)

```
$ csDensity(genes(cuff_data))
```

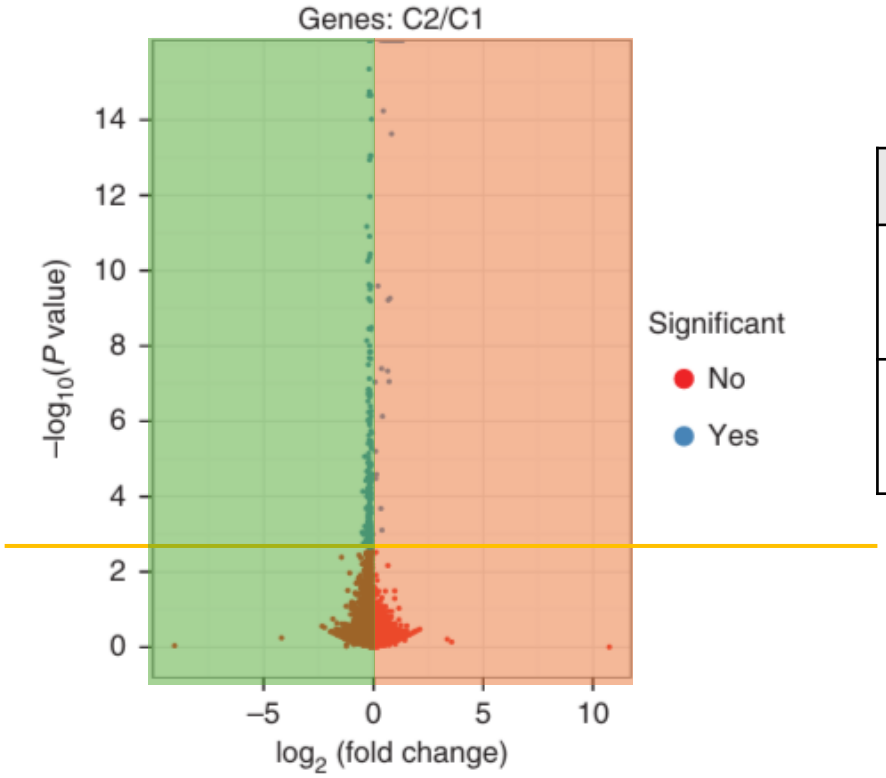


- 정규화 확인: 조건별 샘플의 발현 값 분포가 비슷해야 됨.
- 발현 변화 패턴이 국소적인지 전반적인지 확인 가능

Downstream analysis

- Create a **volcano plot** to inspect differentially expressed genes (Fig. 8)

```
$ csVolcano(genes(cuff_data), 'C1', 'C2')
```



축	의미
X축 ($\log_2$ FC)	C2가 C1 대비 얼마나 발현이 **증가(+)** 또는 **감소(–)**했는지의 로그비율
Y축 ($-\log_{10}$ p-value)	차등발현의 통계적 유의성; **값이 클수록 p-value 작음**(즉 더 유의)

Fold Change (Example)

(1) FPKM table

	Control case (C1)	Perturbed case (C2)
R1 (초파리 A)	21.54	65.22
R2 (초파리 B)	22.81	60.34
R3 (초파리 C)	20.34	68.38

(2) 조건별 요약

$$\text{C1 mean} \approx \frac{21.54 + 22.81 + 20.34}{3} = 21.56$$

$$\text{C2 mean} \approx \frac{65.22 + 60.34 + 68.38}{3} = 64.65$$

(3) log 2 fold change (C1 vs C2)

$$\log_2\left(\frac{64.65}{21.56}\right) \approx \boxed{1.58} \quad (\text{약 } 3.0\text{배 증가})$$

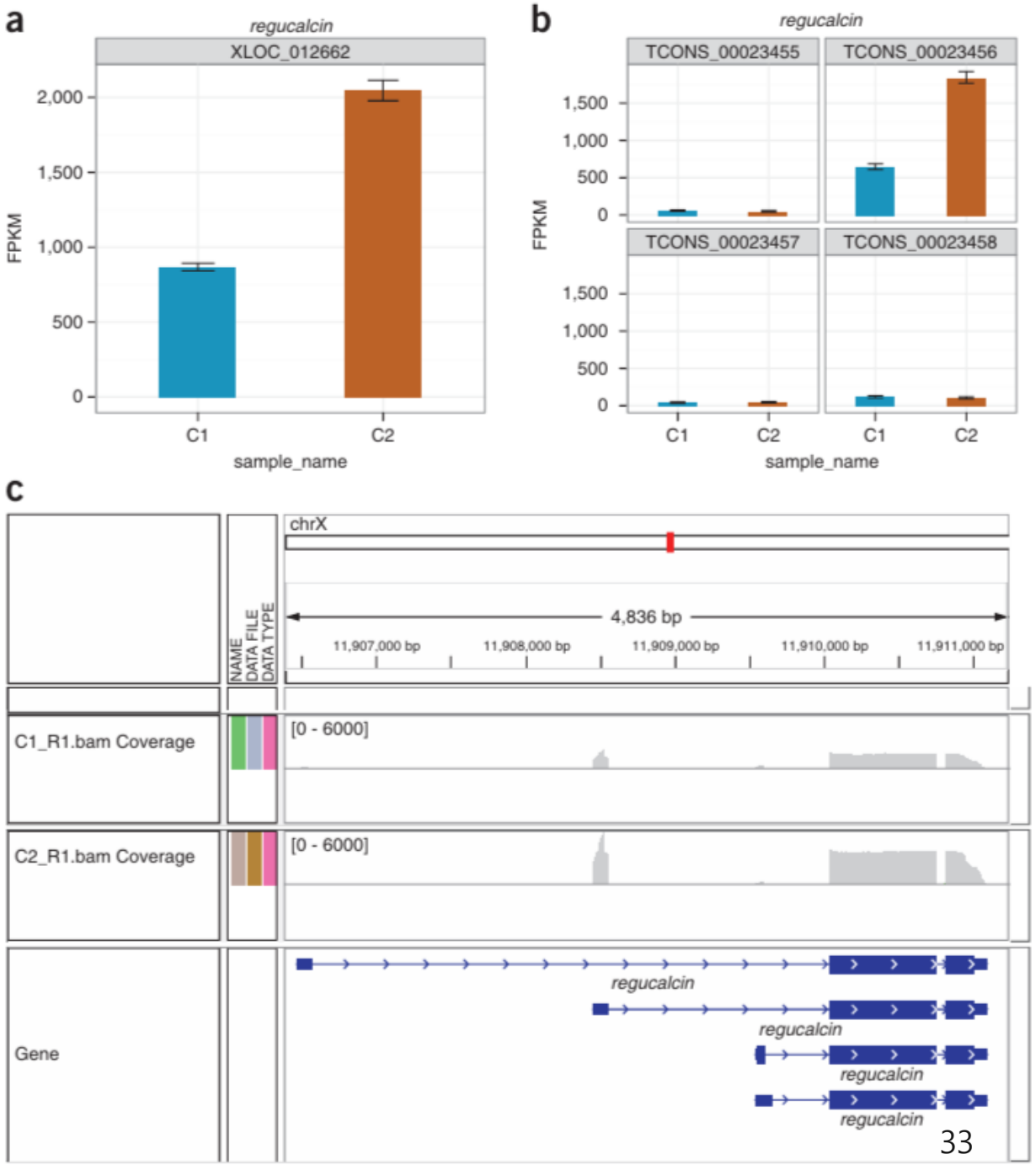
Downstream analysis

- Plot expression levels for genes of interest with bar plots (Fig. 9a)

```
$ csVolcano(genes(cuff_data), 'C1', 'C2')
```

- Plot individual isoform expression levels of selected genes of interest with bar plots (Fig. 9b):

```
$ expressionBarplot(isoforms(mygene))
```



Downstream analysis with CummeRbund

- Create a volcano plot to inspect differentially expressed genes (Fig. 8)

TABLE 5 | Differentially expressed and regulated gene calls made for the example data set.

Differentially expressed genes	312
Differentially expressed transcripts	209
Differentially expressed TSS groups	228
Differentially expressed coding sequences	115
Differentially spliced TSS groups	75
Genes with differential promoter use	204
Genes with differential CDS output	35

Downstream analysis with CummeRbund

TABLE 2 | Troubleshooting table.

Step	Problem	Possible reason	Solution
1	TopHat cannot find Bowtie or the SAM tools	Bowtie and/or SAM tools binary executables are not in a directory listed in the PATH shell environment variable	Add the directories containing these executables to the PATH environment variable. See the man page of your UNIX shell for more details
2	Cufflinks crashes with a 'bad_alloc' error Cufflinks takes excessively long to finish	Machine is running out of memory trying to assemble highly expressed genes	Pass the –max-bundle-frags option to Cufflinks with a value of <1,000,000 (the default). Try 500,000 at first, and lower values if the error is still thrown
5	Cuffdiff crashes with a 'bad_ alloc' error Cuffdiff takes excessively long to finish	Machine is running out of memory trying to quantify highly expressed genes	Pass the –max-bundle-frags option to Cuffdiff with a value of <1,000,000 (the default). Try 500,000 at first, and lower values if the error is still thrown
	Cuffdiff reports FPKM = 0 for all genes and transcripts	Chromosome names in GTF file do not match the names in the BAM alignment files	Use a GTF file and alignments that has matching chromosome names (e.g., the GTF included with an iGenome index)

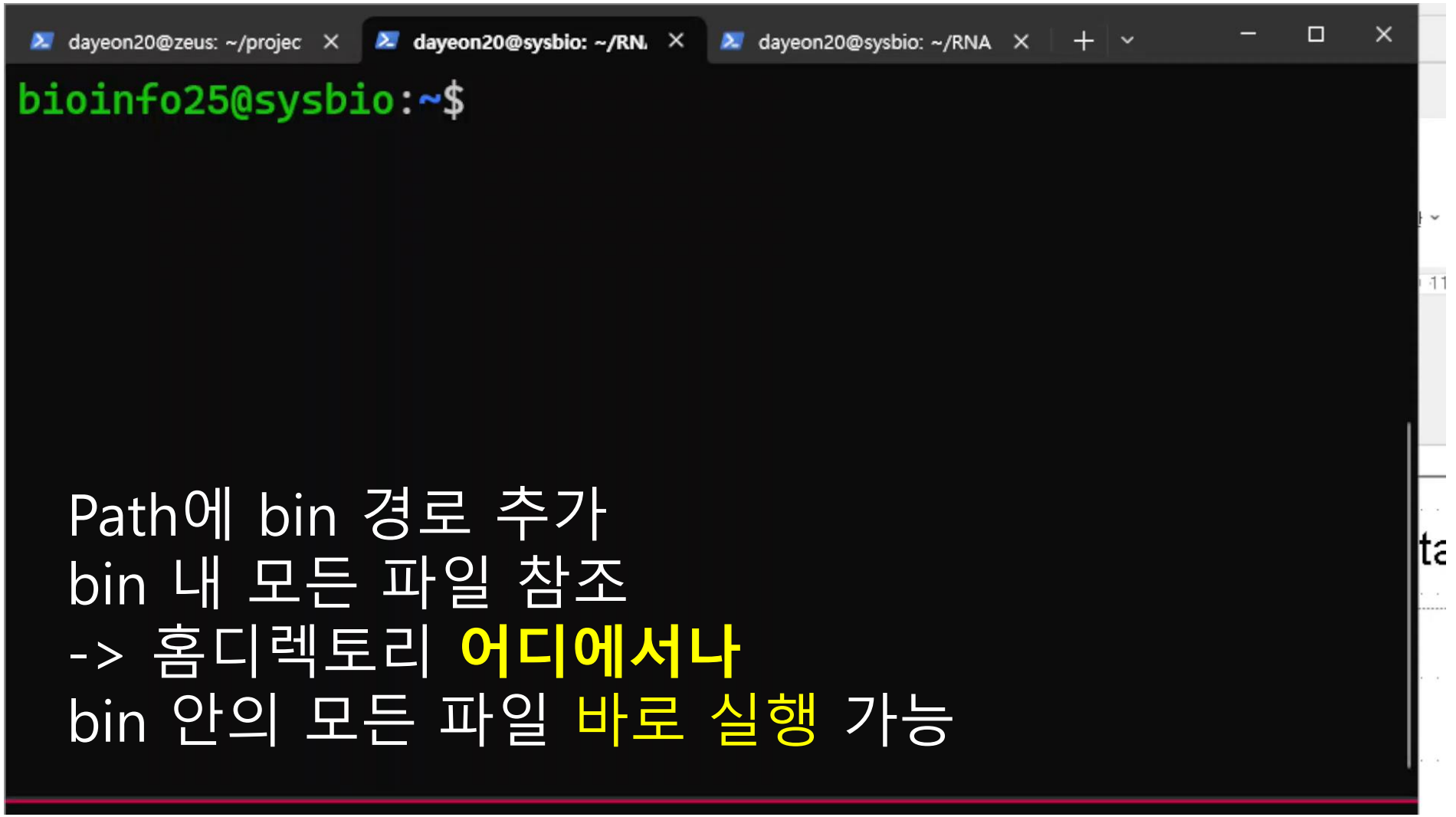
Thank you

Dayeon Kim
dayeonkim@hufs.ac.kr
Hankuk University of Foreign Studies
Division of Biomedical Engineering

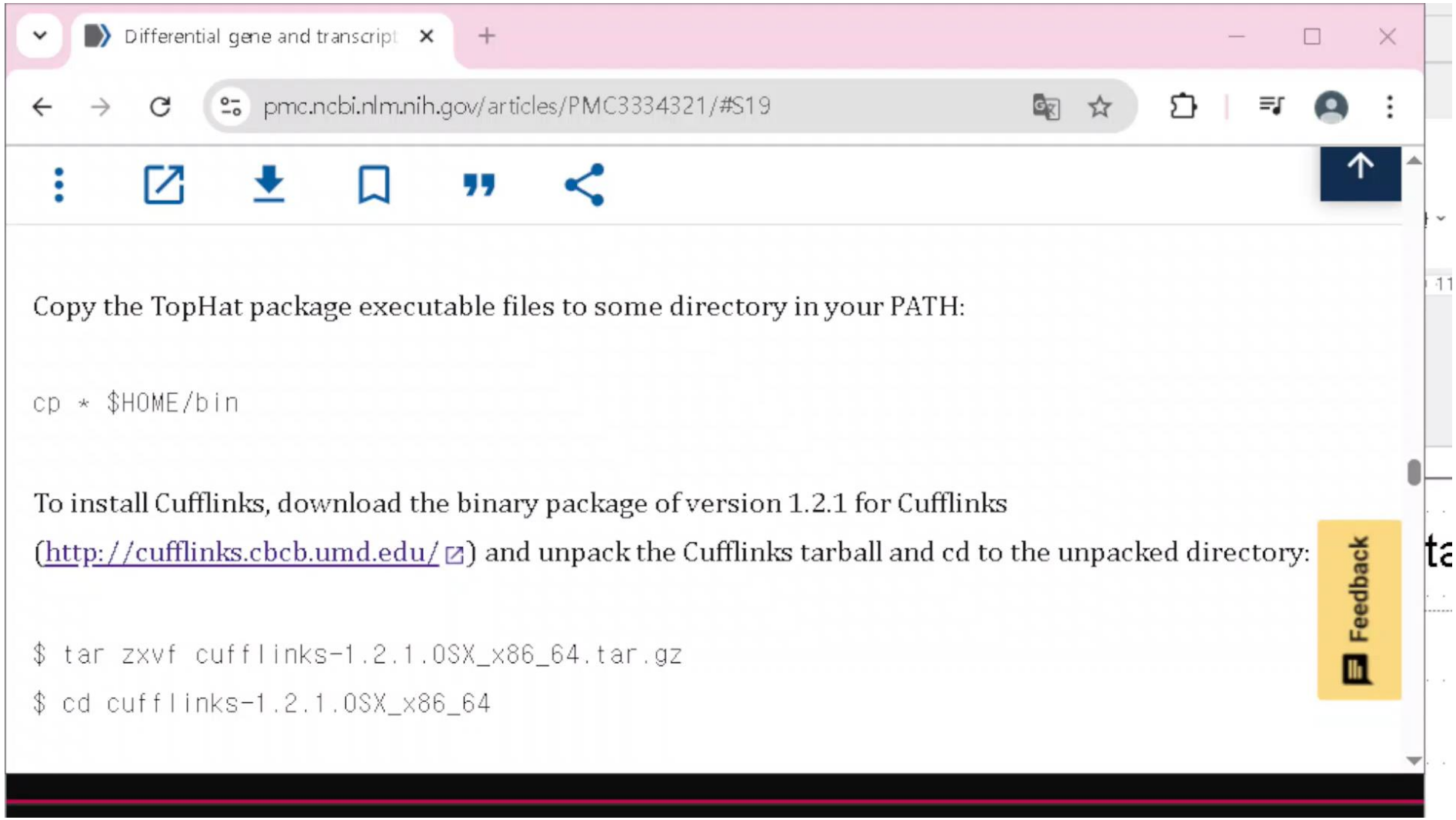
Systems Biology Lab

Supplementary

Downloading and installing software



Downloading and installing software



Dayeon Kim(dayeonkim@hufs.ac.kr)

R execute in Linux

R 4.0 이하버전에서는 CummeRbund 지원하지 않음

➔ 개인 컴퓨터에서 진행, R 4 버전 설치하여 사용가능!

참고자료(https://blog.naver.com/songsite123/223334808151)

Error 1 - TopHat

Not found genes.gtf file

- 실행 위치에 genes.gtf 파일이 있어야 됨
- ln –s /home/bioinf25/DATA . 를 실행해줘야 됨

Error 2 - Cuffdiff

Error: number of labels must match number of conditions

- -L C1,C2 (label 2개)인데,

  실제로는Cuffdiff이 인식한 그룹이 2개가 아니라 더 많거나 적게 인식된 것.

- 이는 주로 **줄바꿈 또는 쉼표·공백 위치** 때문에 발생.

- 동일한 그룹내 파일들은 이름 사이 쉼표(,)랑 온점(.) 사이 공백 없도록.