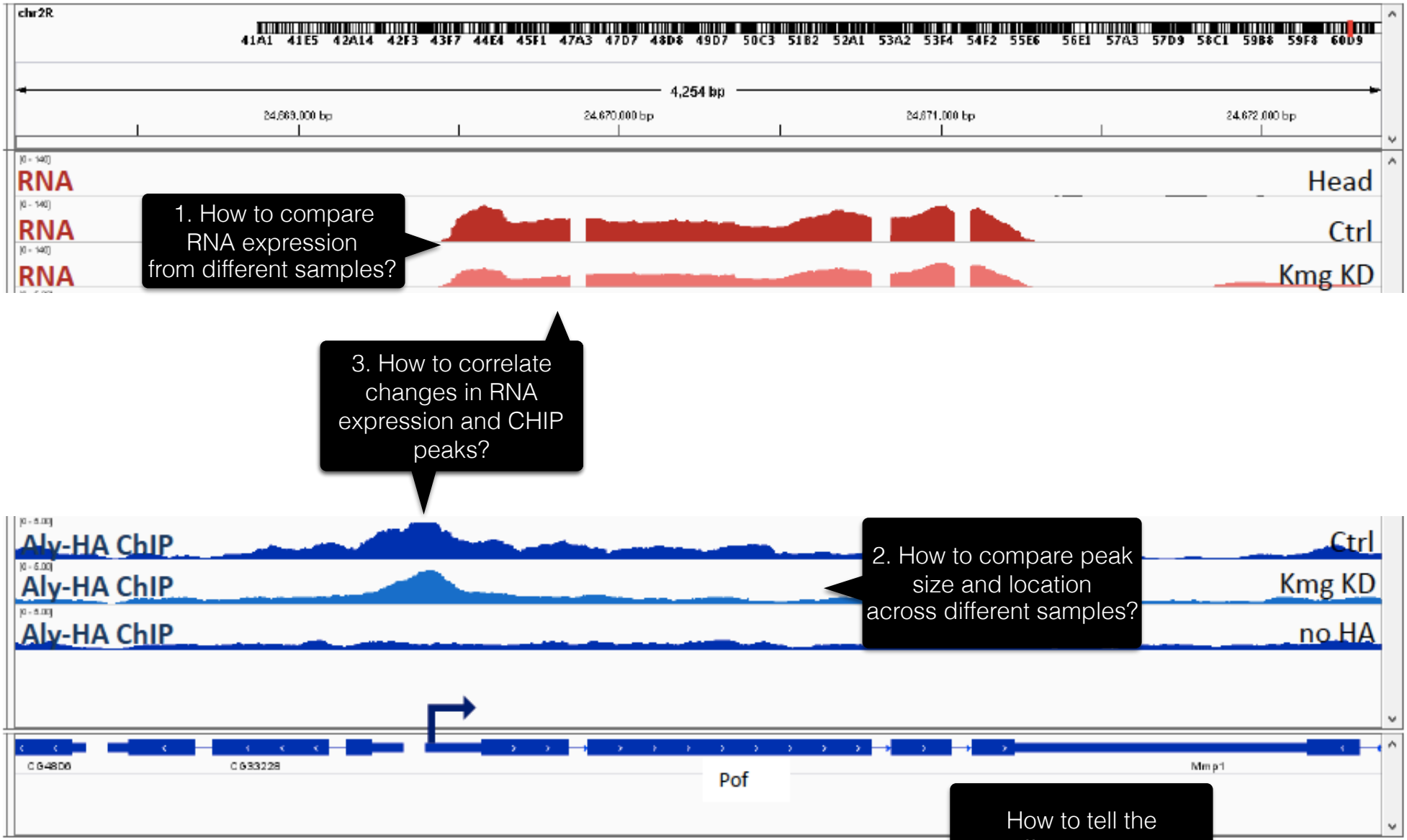
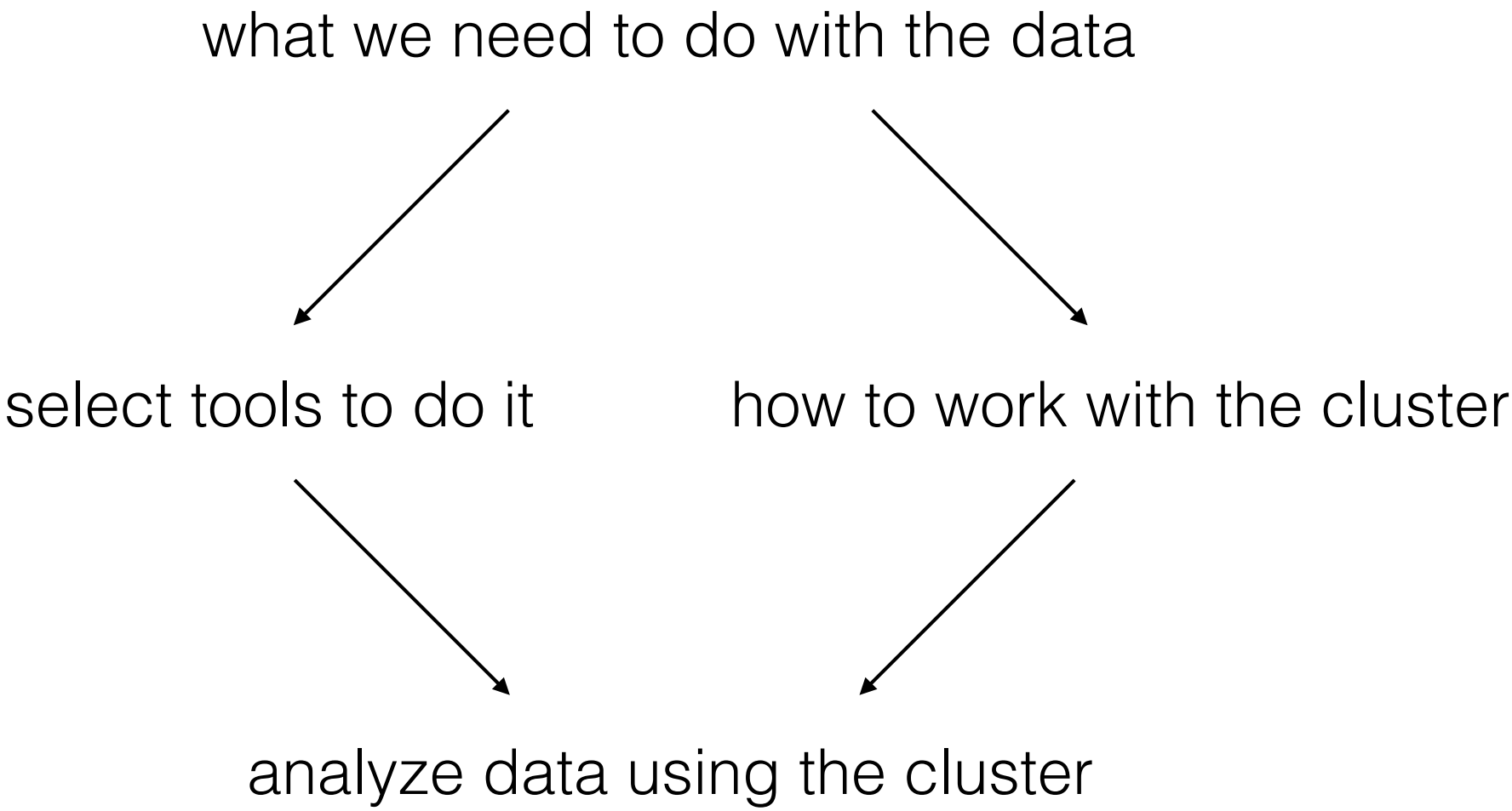
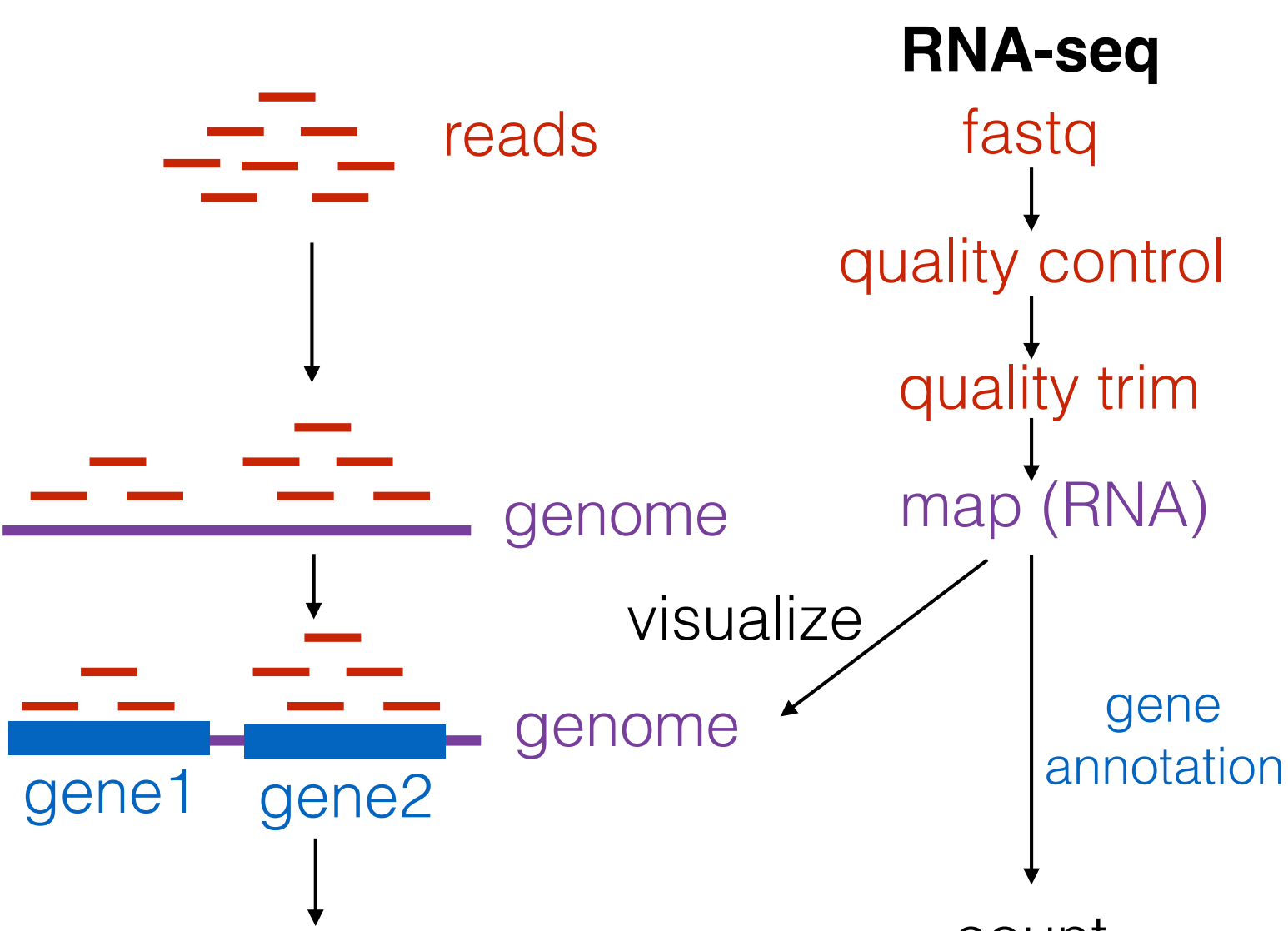
Sequencing data analysis

The Pof Workshop
7/18/16

What do we want to know?

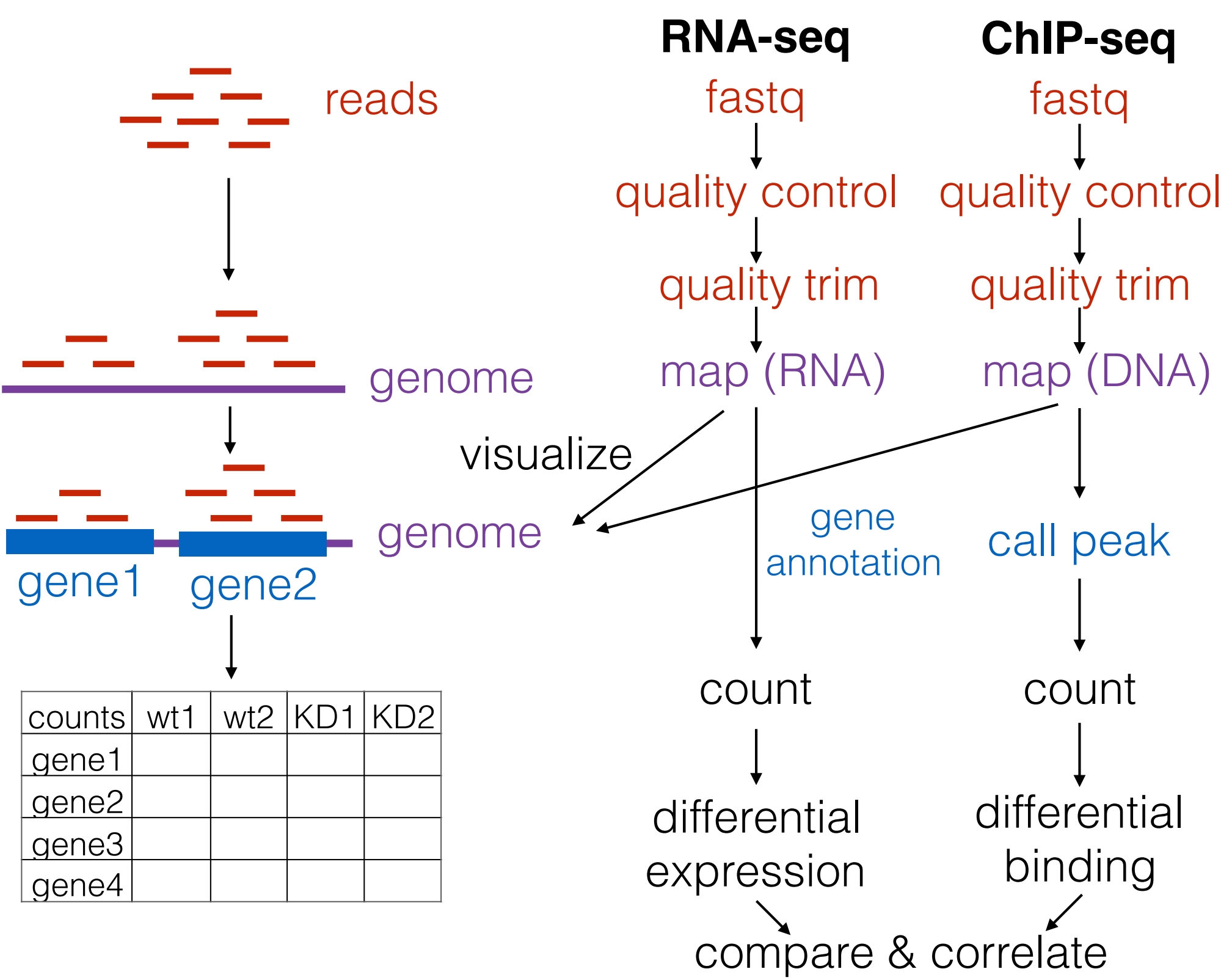



what we need to do with the data
select tools to do it
how to work with the cluster
analyze data using the cluster

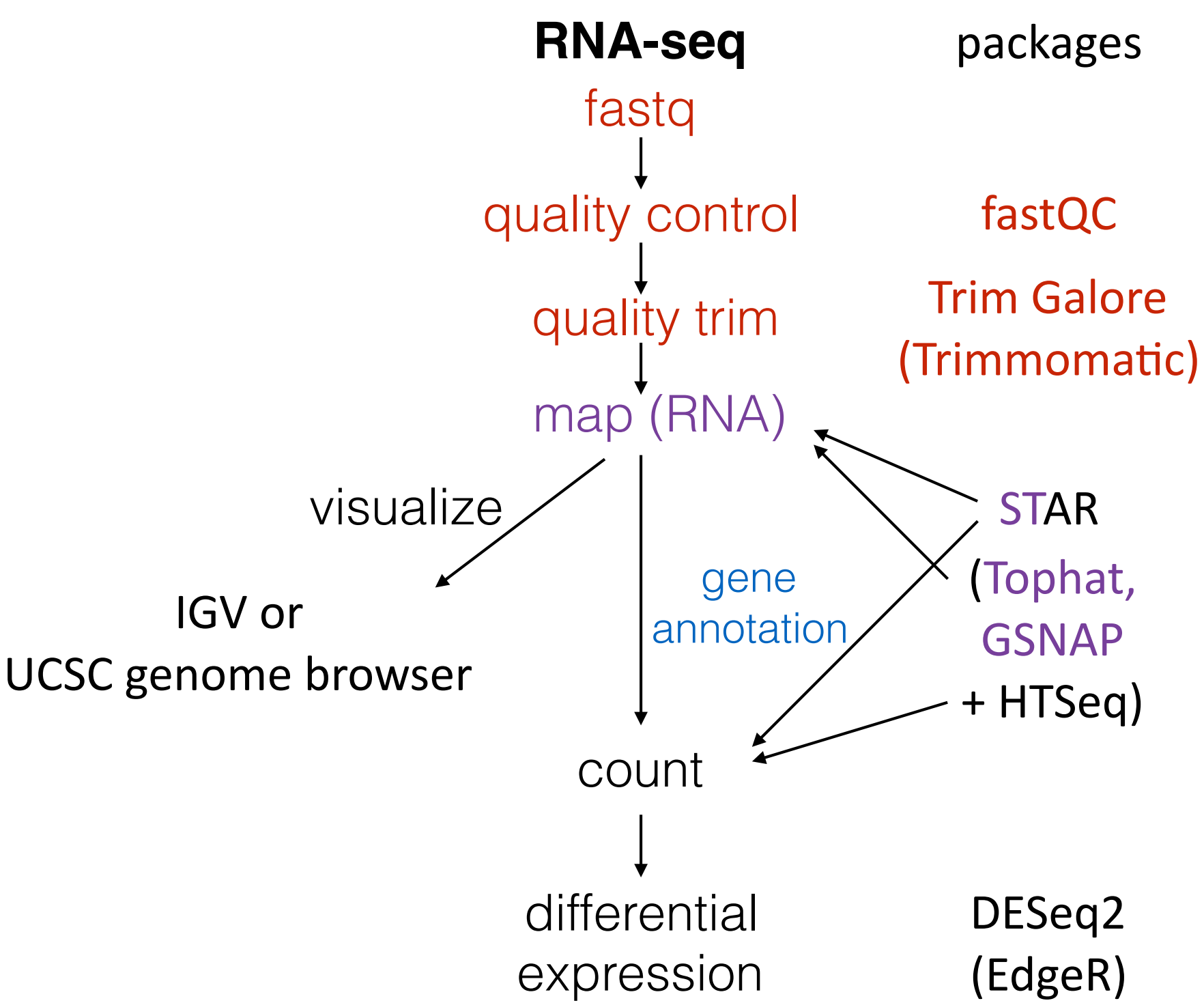


counts	wt1	wt2	KD1	KD2
gene1				
gene2				
gene3				
gene4				

why Page lab may not see pre-pi-RNA?



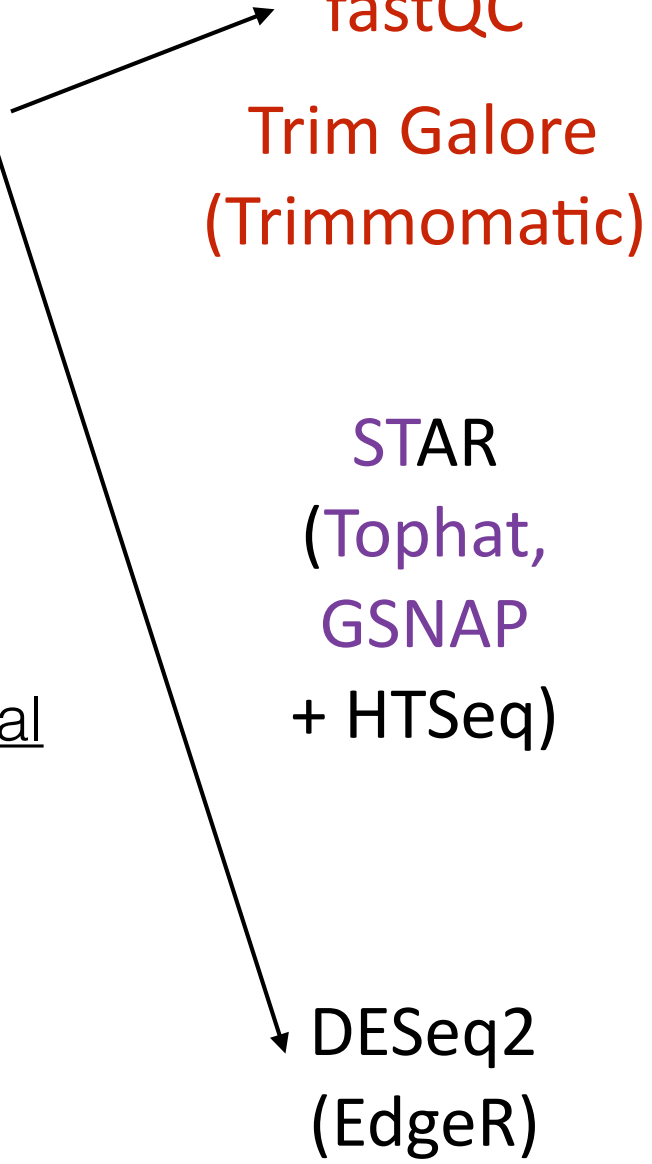
counts	wt1	wt2	KD1	KD2
gene1				
gene2				
gene3				
gene4				


RNA-seq
packages
fastq
quality control
fastQC
quality trim
Trim Galore
(Trimmomatic)
map (RNA)
visualize
STAR
(Tophat,
GSNAP
+ HTSeq)
gene
annotation
IGV or
UCSC genome browser
count
differential
expression
DESeq2
(EdgeR)

packages

How to choose packages to use:

- Most recommended or popular (may not be the best choice but less likely to be questioned)
  - http://homer.salk.edu/homer/basicTutorial/index.html
  - https://www.broadinstitute.org/gatk/guide/best-practices.php
  - https://github.com/griffithlab/rnaseq_tutorial
  - https://www.biostars.org
  - http://seqanswers.com



packages

How to choose packages to use:

- It does what you need

fastQC

Trim Galore
(Trimmomatic)

STAR
(Tophat,
GSNAP
+ HTSeq)

DESeq2
(EdgeR)

packages

How to choose packages to use:

fastQC

Trim Galore
(Trimmomatic)

- Well-maintained and reliable, less frequent to give errors
- Perform well and fast (most important)

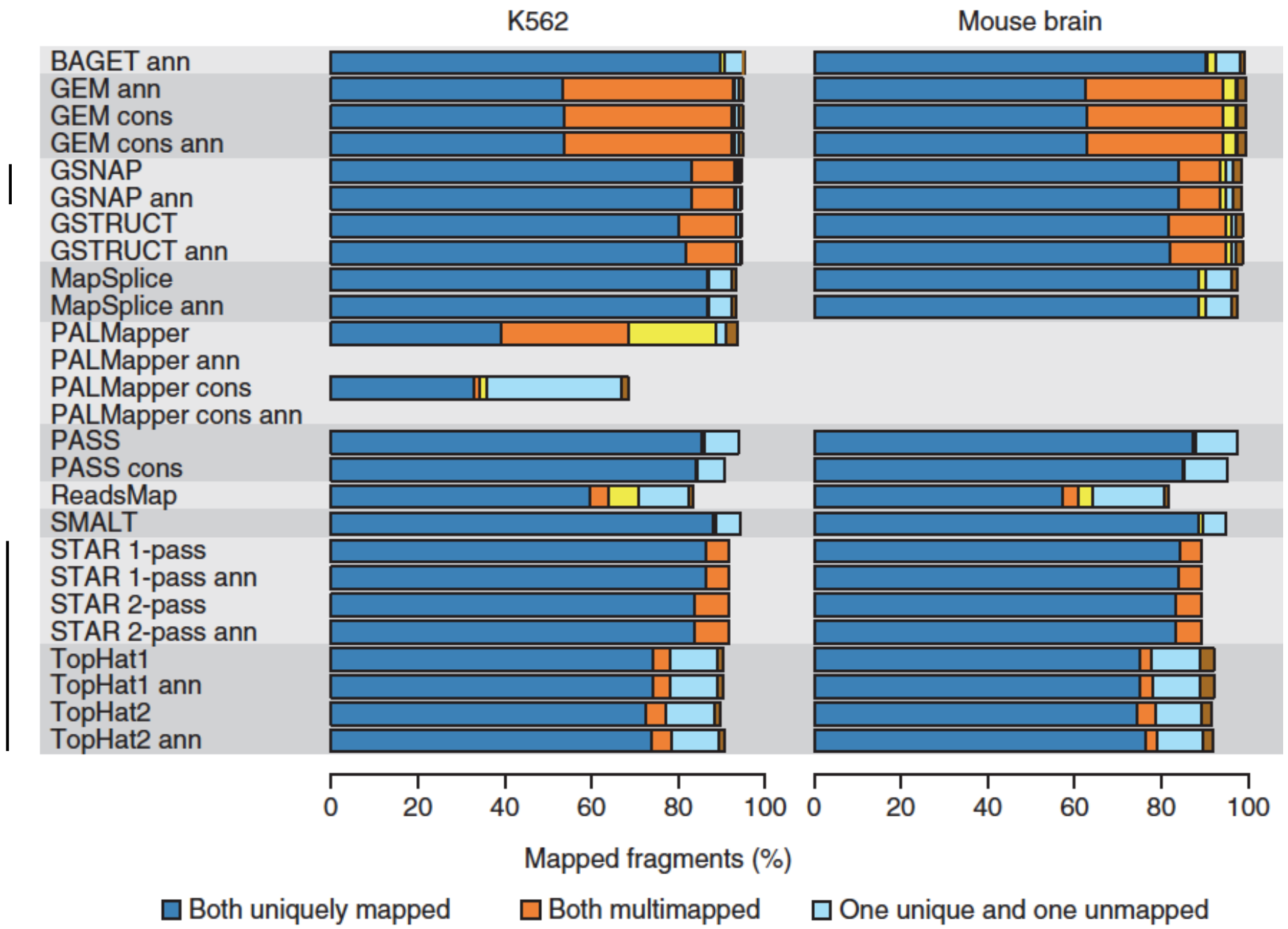
1hr~2hr/50M PE reads → STAR

6hr~over night (Tophat, GSNAP

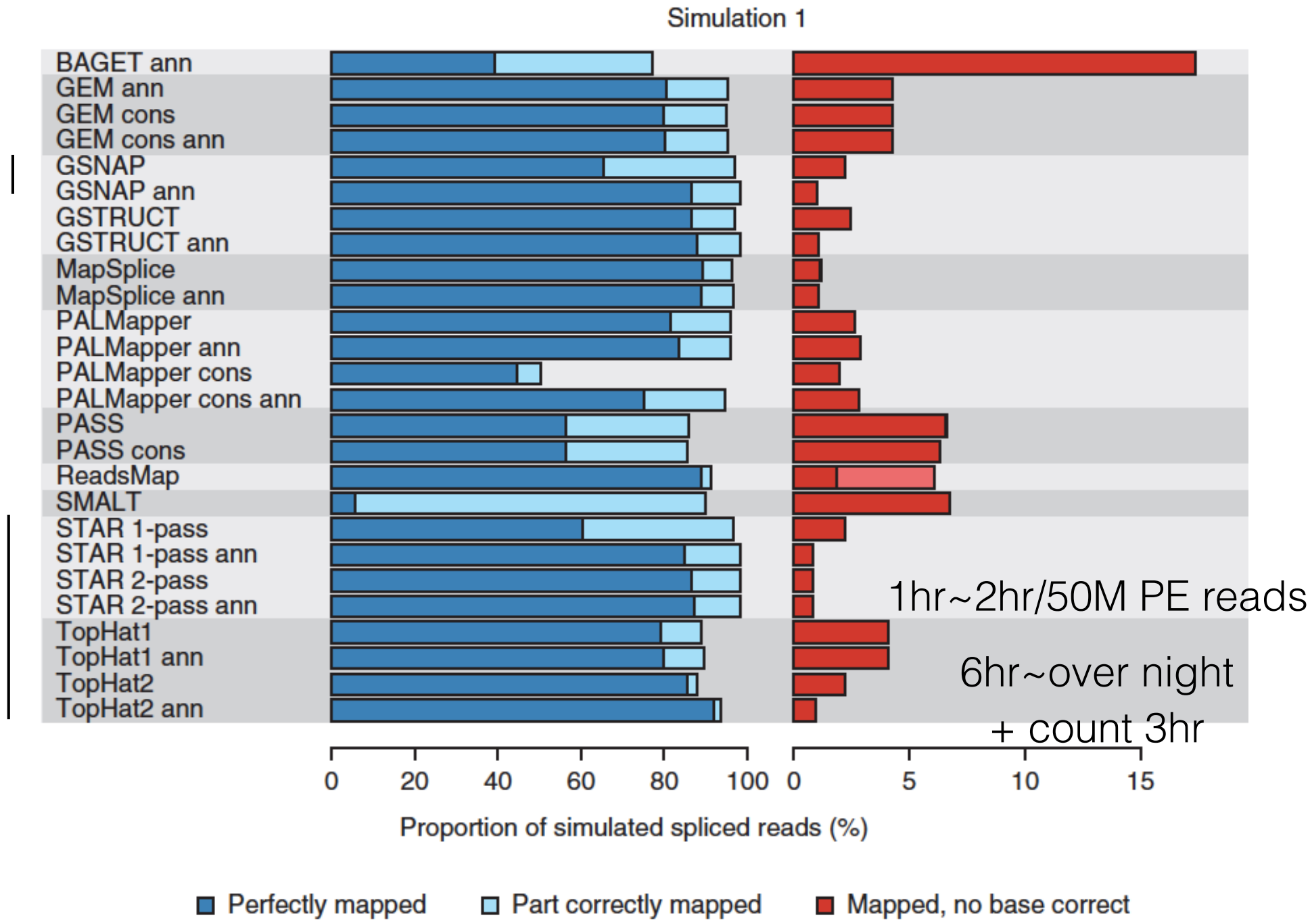
+ count 3hr + HTSeq)

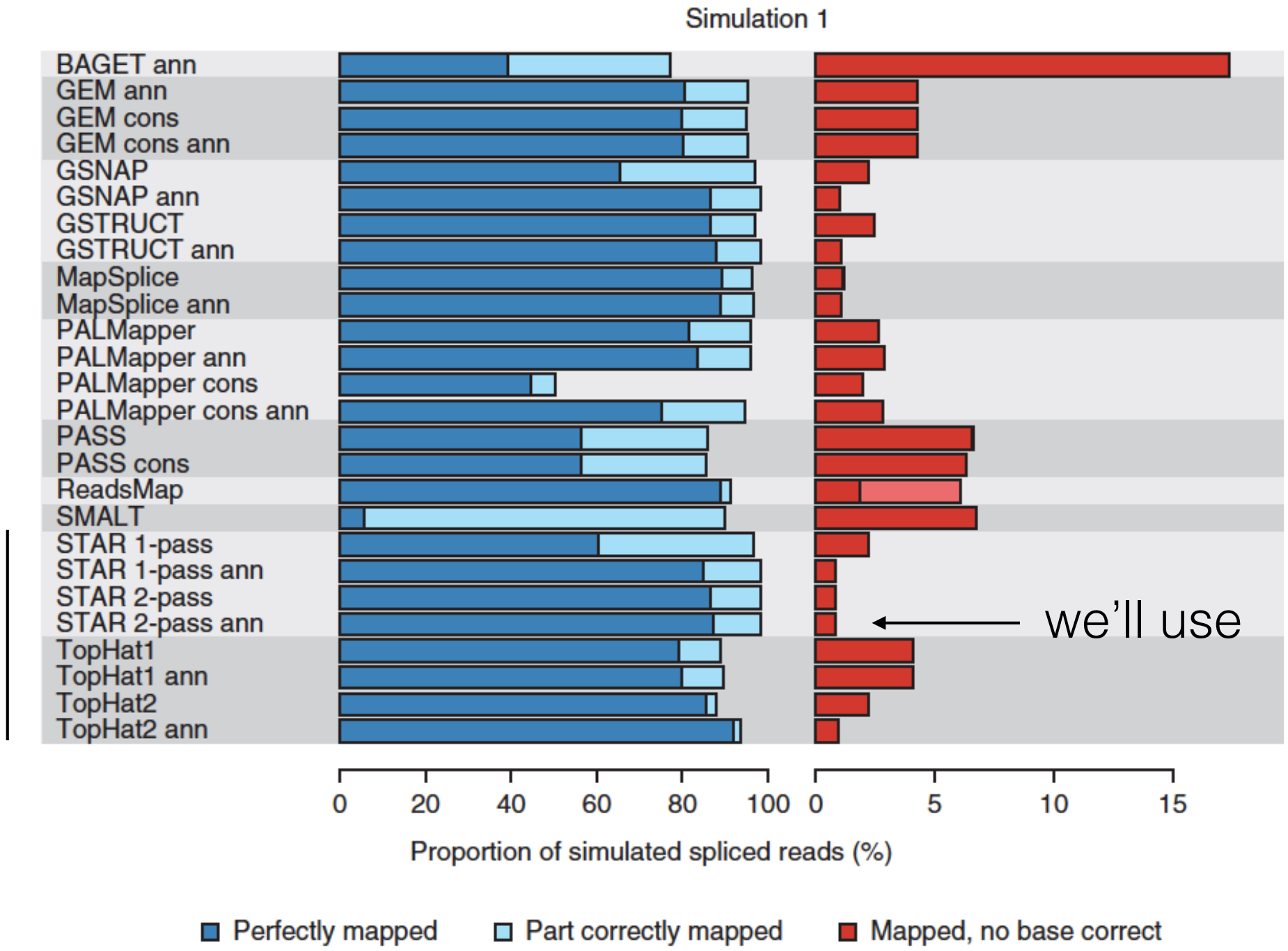
DESeq2
(EdgeR)

Systematic evaluation of spliced alignment programs for RNA-seq data, Nature Methods (2013)



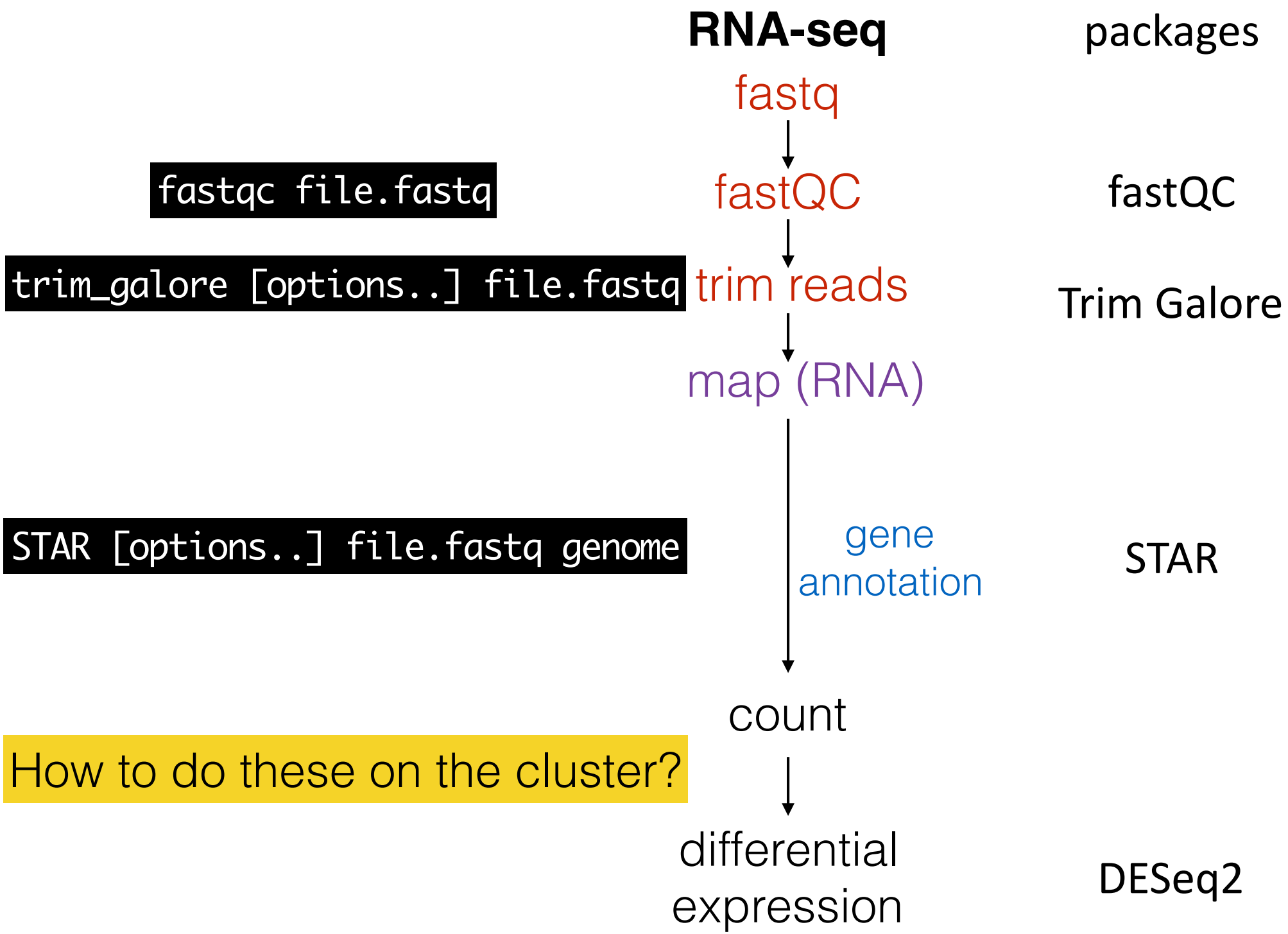
Systematic evaluation of spliced alignment programs for RNA-seq data, Nature Methods (2013)


Simulation 1
BAGET ann
GEM ann
GEM cons
GEM cons ann
GSNAP
GSNAP ann
GSTRUCT
GSTRUCT ann
MapSplice
MapSplice ann
PALMapper
PALMapper ann
PALMapper cons
PALMapper cons ann
PASS
PASS cons
ReadsMap
SMALT
STAR 1-pass
STAR 1-pass ann
STAR 2-pass
STAR 2-pass ann
TopHat1
TopHat1 ann
TopHat2
TopHat2 ann
0
20
40
60
80
100
0
5
10
15
Proportion of simulated spliced reads (%)
Perfectly mapped
Part correctly mapped
Mapped, no base correct
1hr~2hr/50M PE reads
6hr~over night
+ count 3hr


Simulation 1
BAGET ann
GEM ann
GEM cons
GEM cons ann
GSNAP
GSNAP ann
GSTRUCT
GSTRUCT ann
MapSplice
MapSplice ann
PALMapper
PALMapper ann
PALMapper cons
PALMapper cons ann
PASS
PASS cons
ReadsMap
SMALT
STAR 1-pass
STAR 1-pass ann
STAR 2-pass
STAR 2-pass ann
TopHat1
TopHat1 ann
TopHat2
TopHat2 ann
we'll use
0
20
40
60
80
100
0
5
10
15
Proportion of simulated spliced reads (%)
Perfectly mapped
Part correctly mapped
Mapped, no base correct

Keep an eye out for the new development!

- Kallisto: psuedoalignment and count reads/gene
- Take 5~10min on a LAPTOP!
- Can be more accurate than traditional methods
- https://pachterlab.github.io/kallisto/about
- https://liorpachter.wordpress.com/2015/05/10/near-optimal-rna-seq-quantification-with-kallisto/


RNA-seq
packages
fastq
fastqc file.fastq
fastQC
fastQC
trim_galore [options..] file.fastq
trim reads
Trim Galore
map (RNA)
STAR [options..] file.fastq genome
gene annotation
STAR
count
How to do these on the cluster?
differential expression
DESeq2

All you actually need to learn

(more or less)

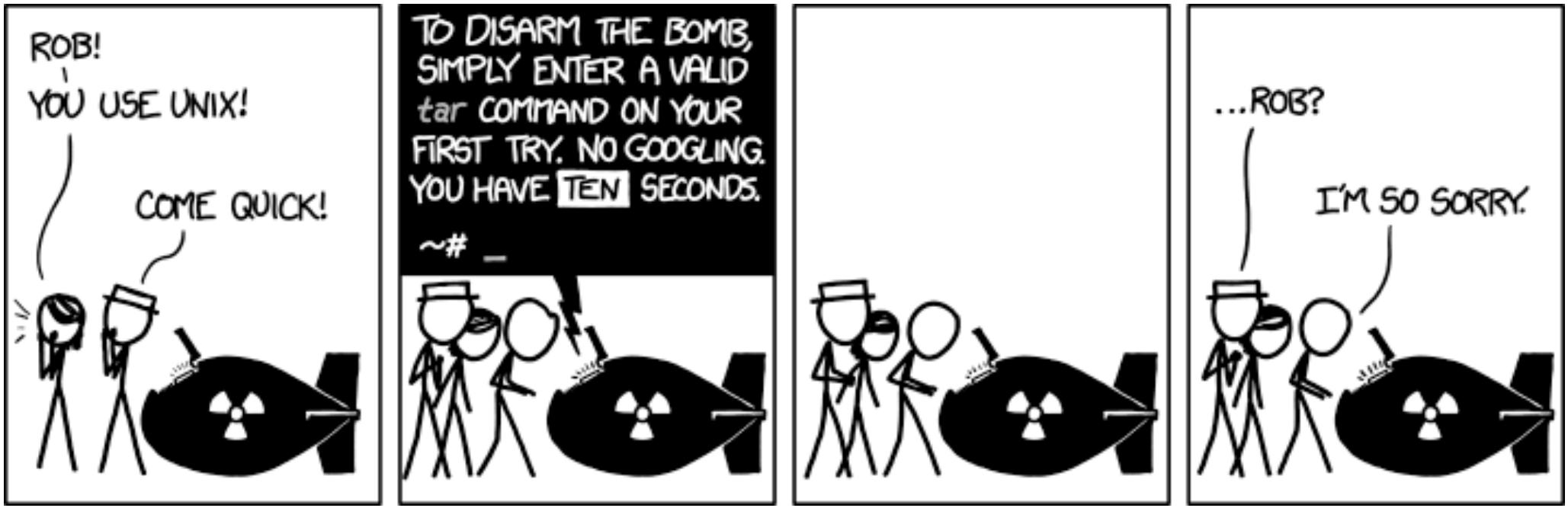
- Log in to the cluster
- With SFTP software upload data and scripts
- On command line, navigate to the folder where the data is
- Paste in the following lines, hit ‘Enter’

```
qsub Run01FastQC.sh
qsub -hold_jid FastQC -t 1-4 Run02Trim_PE.sh
qsub -hold_jid Trim_PE -t 1-4 Run03Map_pass1.sh
qsub -hold_jid Map_pass1 -t 1-4 Run04Map_pass2.sh
qsub -hold_jid Map_pass2 -t 1-4 Run05Index_TDF.sh
qsub -hold_jid Map_pass2 Run05CountTable.sh 4
```

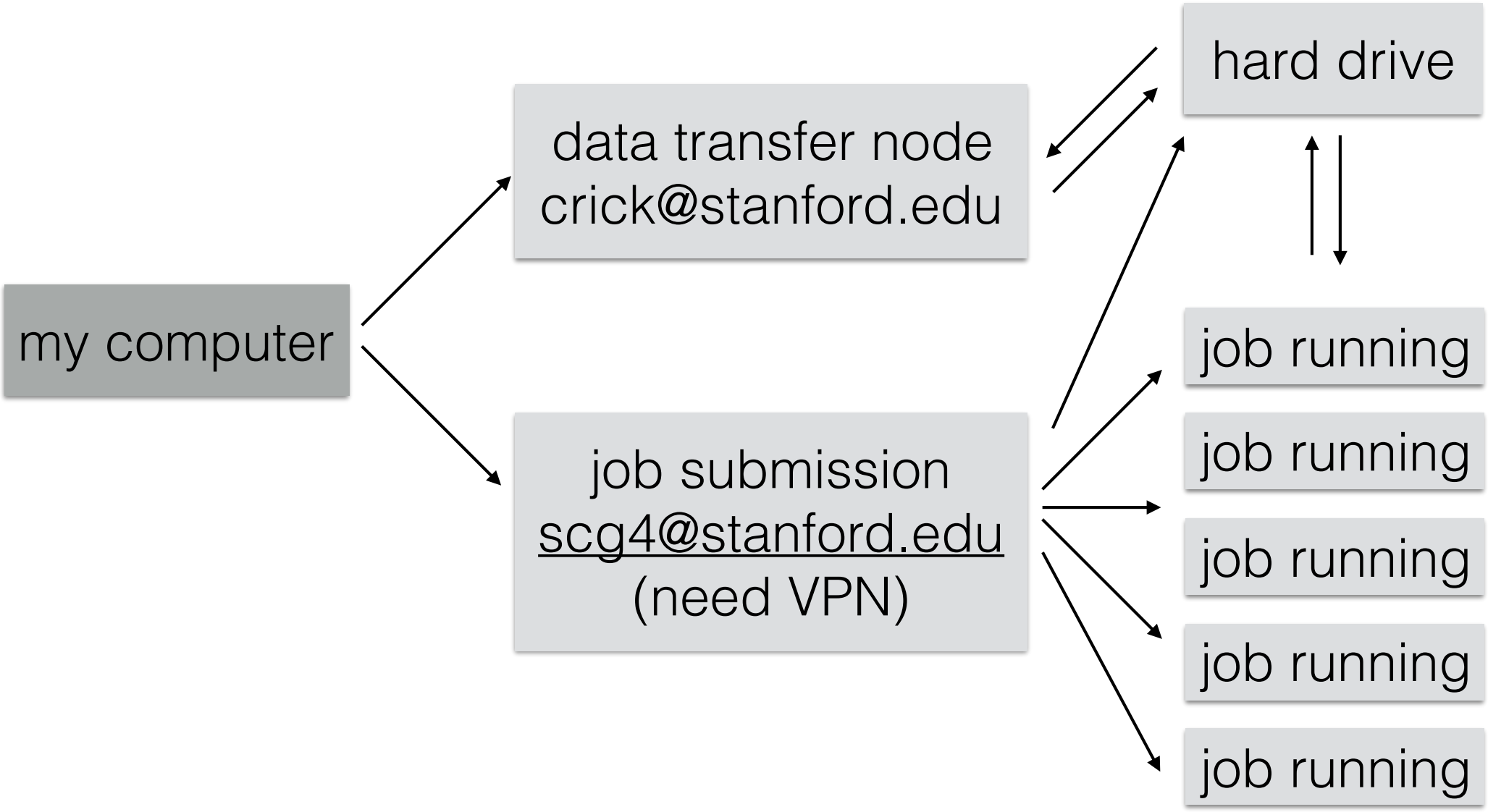
A1 | X

	A	B	C	D	E	F	G	H	I
1	X	baseMean	log2FoldChar	lfcSE	stat	pvalue	padj	V1	V3
2	FBgn003788	19671.3737	-8.5659473	0.19749093	-43.373878	0	0	scpr-B	CG17210
3	FBgn003788	17910.3814	-8.3480306	0.2108231	-39.597324	0	0	scpr-A	CG5207
4	FBgn003787	21934.1957	-6.2694633	0.17549685	-35.724079	1.67E-279	9.07E-276	scpr-C	CG5106
5	FBgn003800	4417.66542	-6.2879284	0.2091285	-30.067295	1.30E-198	5.28E-195	CG3942	CG3942
6	FBgn003396	6829.82512	-5.7727289	0.2017897	-28.607648	5.40E-180	1.76E-176	CG10202	CG10202
7	FBgn004302	15672.956	-5.8215226	0.20752302	-28.052419	3.73E-173	1.01E-169	Adgf-A2	CG32178
8	FBgn003949	6496.69201	-6.0949112	0.22516896	-27.068168	2.33E-161	5.43E-158	CG6059	CG6059
9	FBgn003825	7585.46935	-5.8324434	0.21604113	-26.996913	1.61E-160	3.27E-157	CG7362	CG7362
10	FBgn005035	6133.99344	-5.9543347	0.22170913	-26.856515	7.08E-159	1.28E-155	CG30356	CG30356
11	FBgn006636	3351.12023	6.30354362	0.23851546	26.4282388	6.49E-154	1.06E-150	dyl	CG15013
12	FBgn002856	5317.5358	-5.8280045	0.22362108	-26.061964	9.85E-150	1.46E-146	robl37BC	CG15171
13	FBgn002889	8196.40001	-6.9896726	0.26931534	-25.953489	1.66E-148	2.25E-145	CG4161	CG4161
14	FBgn005206	11224.758	-5.1861169	0.20024322	-25.899088	6.82E-148	8.54E-145	CG32061	CG32061
15	FBgn003395	6217.71484	-5.1494945	0.20133158	-25.577183	2.74E-144	3.18E-141	Adgf-E	CG10143
16	FBgn003005	4103.11368	-5.1727278	0.20273384	-25.514872	1.35E-143	1.46E-140	CG12111	CG12111
17	FBgn002908	5881.89876	-4.528828	0.17773907	-25.480205	3.27E-143	3.32E-140	gom	CG6727

- Google knows it all

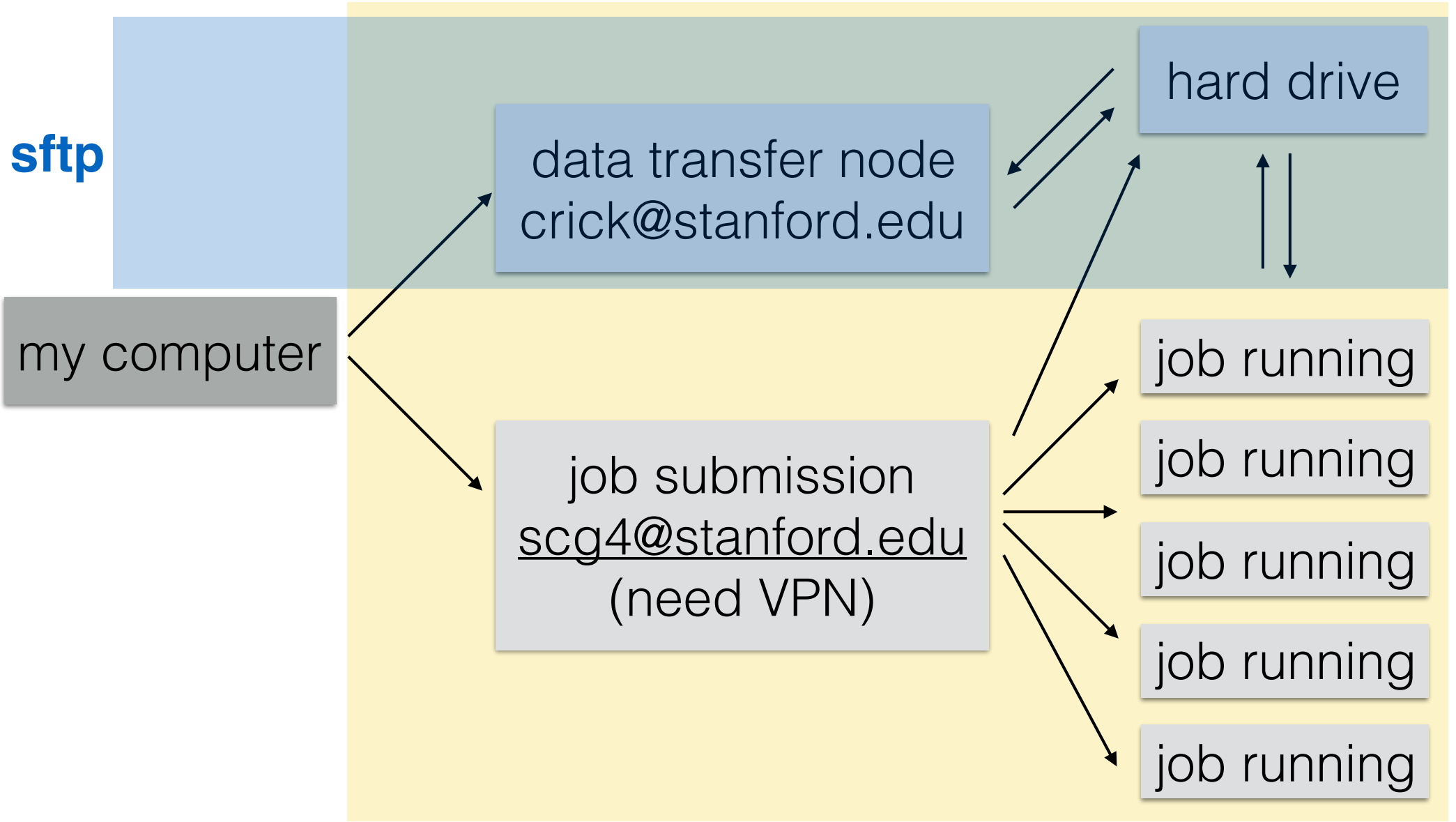


Anatomy of the cluster



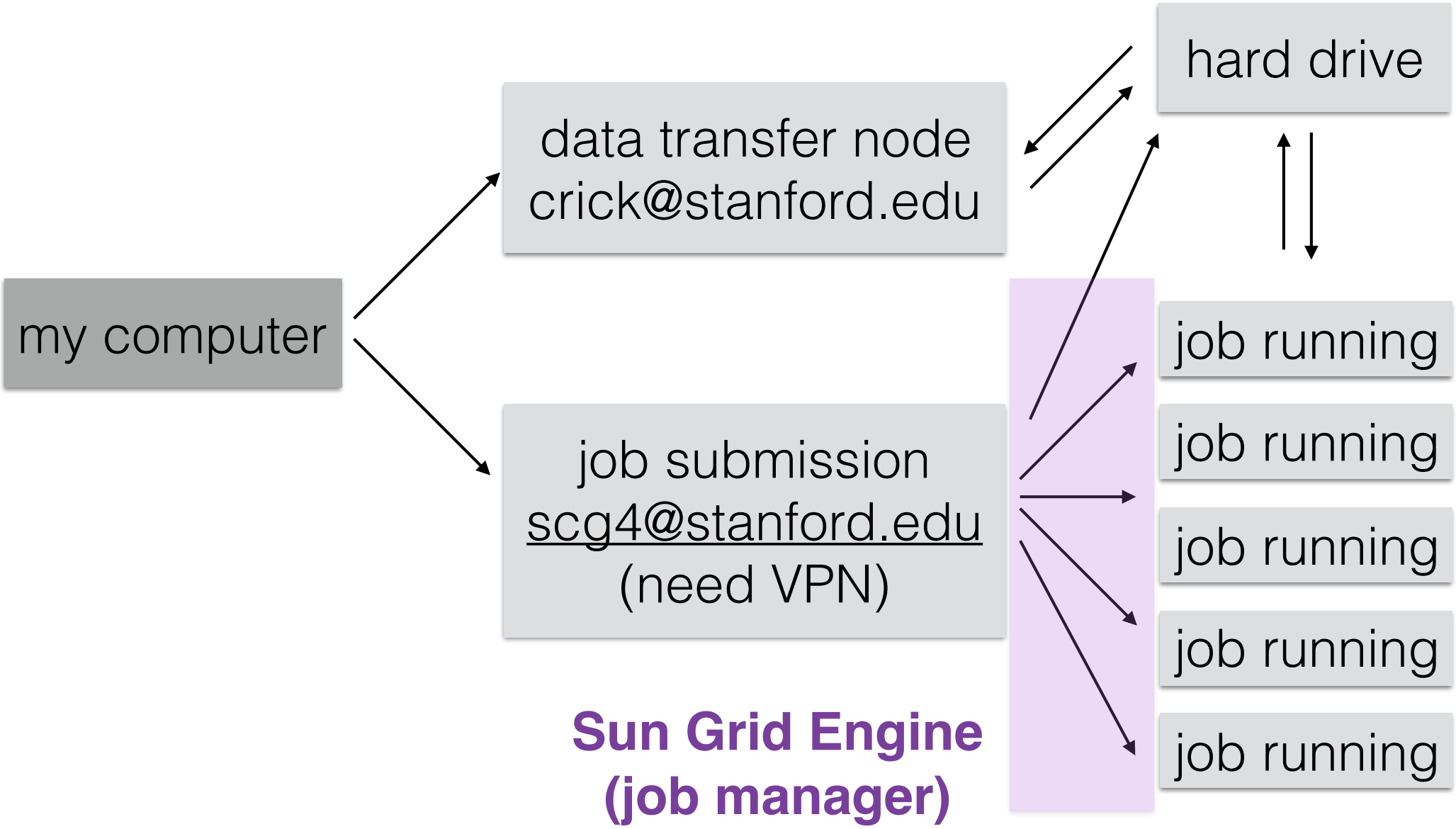
A collection of nodes: each node has many processors and lots of RAM

OS and software



Command line, Bash (for both Linux & Mac)

OS and software



hard drive

when you login, default output folder
10G space
/home/danlu

the lab directory, 10T space
do all your work here!
/srv/gsfs0/projects/fuller/

softwares are in
/srv/gsfs0/software/

Shell 101

- “**cd fuller**”: change to sub-directory ‘fuller’
- “**cd ..**” change to the parent dir
- **tab**: auto completion
- “**ls**”: list contents of the current directory
- “**pwd**”: print current directory
- space and case sensitive!

“Does the whitespace matter;
have I tattooed a syntax error on my arm?!”

http://stackoverflow.com/questions/21186724/why-is-whitespace-sometimes-needed-around-metacharacters

Log in to the cluster

```
ssh SUNID@crick.stanford.edu
```

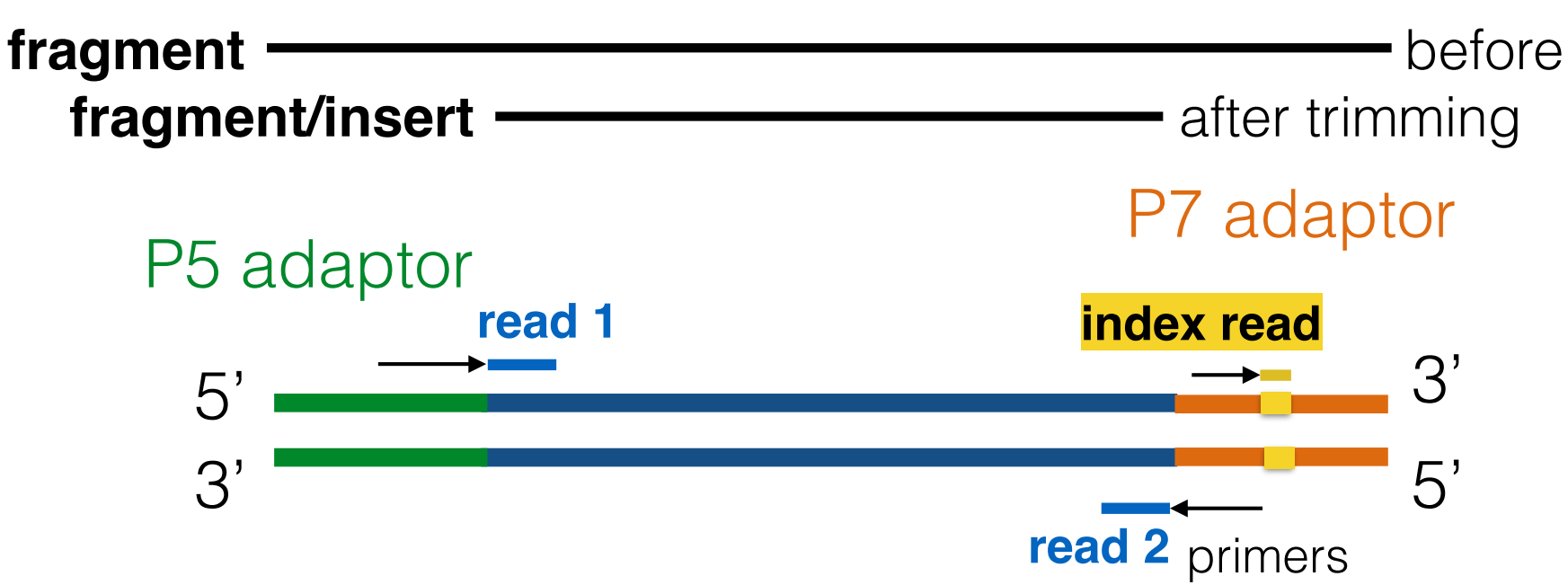
- (give password)
- (NO for the key)

```
pwd

ls
```

- * running the job needs scg4.stanford.edu, with VPN

Task 1: go to
/srv/gsfs0/projects/fuller/workshop/rna_copy
Task 2: draw out

- Where is the adaptor?
- Where is the index (barcode)?
- Where are the reads, and fragment?
- What is single-end or paired-end sequencing?
- What is a mate?
- What is a library?



- Where is the adaptor?
- Where is the index (barcode)?
- Where are the reads, and fragment?
- What is single-end or paired-end sequencing?
- What is a mate?
- What is a library?

  One sample, usually fragments with the same index

Shell 102

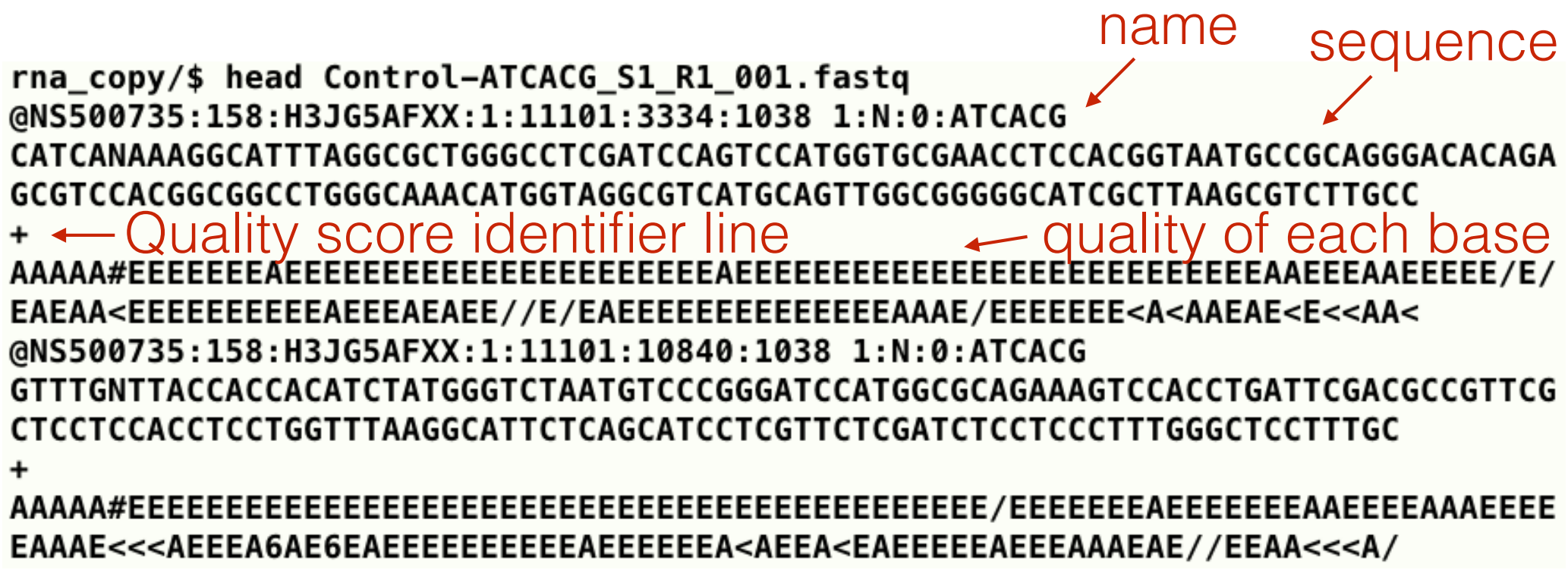
- “*****”: anything
- **[tab]**: auto-completes the unique part
- “**head**”: view the first few lines of the file
- “**tail**”: view the last few lines of the file
- “**less**” is “**more**”: view the file, “**q**” to exit

what would a .fastq file contain?

/srv/gsfs0/projects/fuller/workshop/rna_copy

```
ls

ls *.fastq

head Con[tab]1[tab]
```

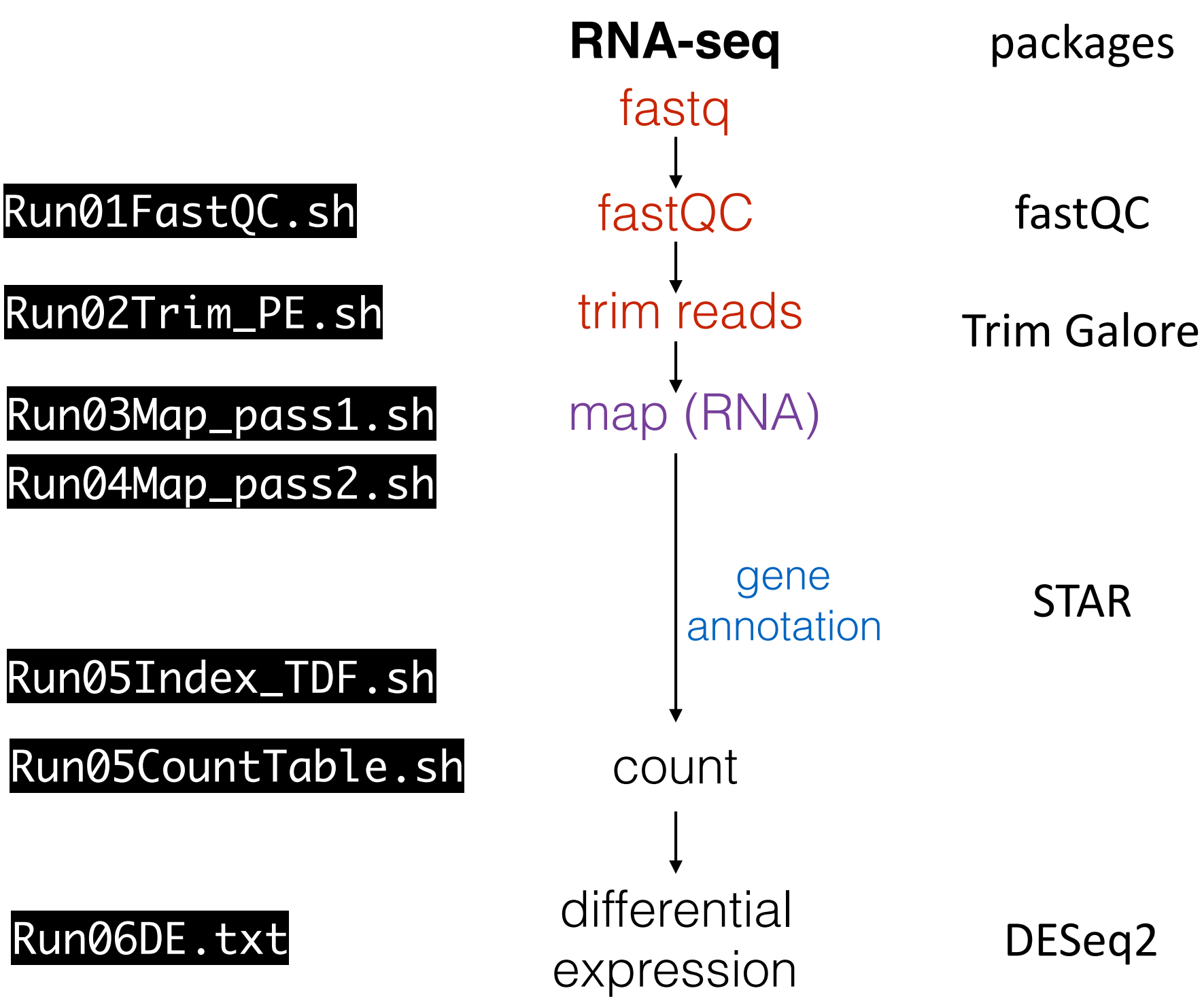
fastq file (.fastq .fq)



- compare with name of

```
head Con[tab]2[tab]
```

http://support.illumina.com/help/SequencingAnalysisWorkflow/Content/Vault/Informatics/Sequencing_Analysis/CASAVA/swSEQ_mCA_FASTQFiles.htm


RNA-seq
packages
fastq
Run01FastQC.sh
fastQC
fastQC
Run02Trim_PE.sh
trim reads
Trim Galore
Run03Map_pass1.sh
map (RNA)
Run04Map_pass2.sh
gene annotation
STAR
Run05Index_TDF.sh
Run05CountTable.sh
count
differential expression
Run06DE.txt
DESeq2

less Run01FastQC.sh

```
#!/bin/bash
```
send the script to Bash

```
#$ -N FastQC
#$ -q standard
#$ -cwd
#$ -l h_vmem=12G
#$ -w e
```
for queue manager

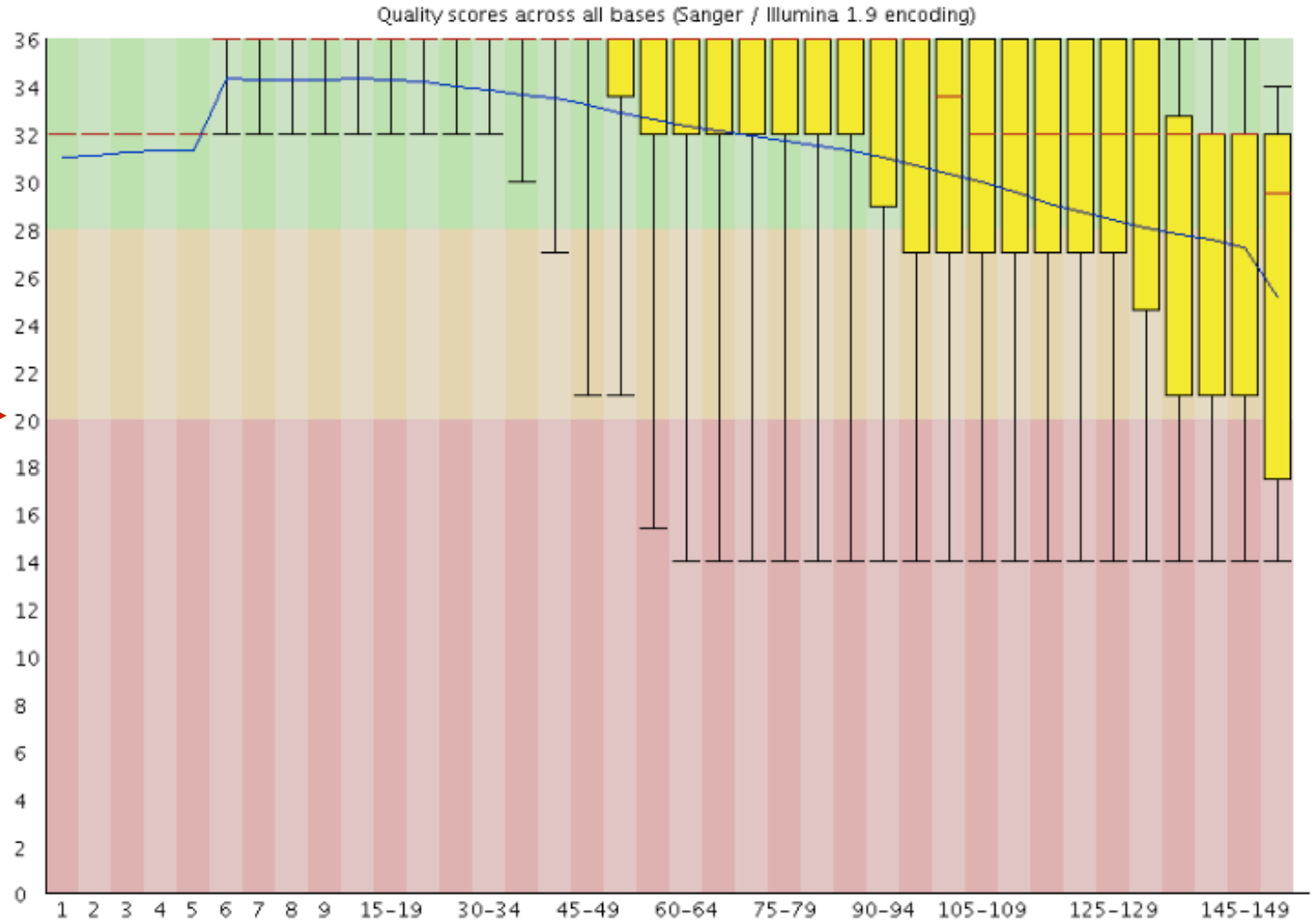
```
module add fastqc
```
tell the system where to find the software

```
mkdir -p logs
mkdir -p fastQC
```
create folders for logs and output

```
fastqc *.fastq --outdir fastQC
```
run fastQC on all fastq files, output to fastQC folder

fastqc report

Per base sequence quality



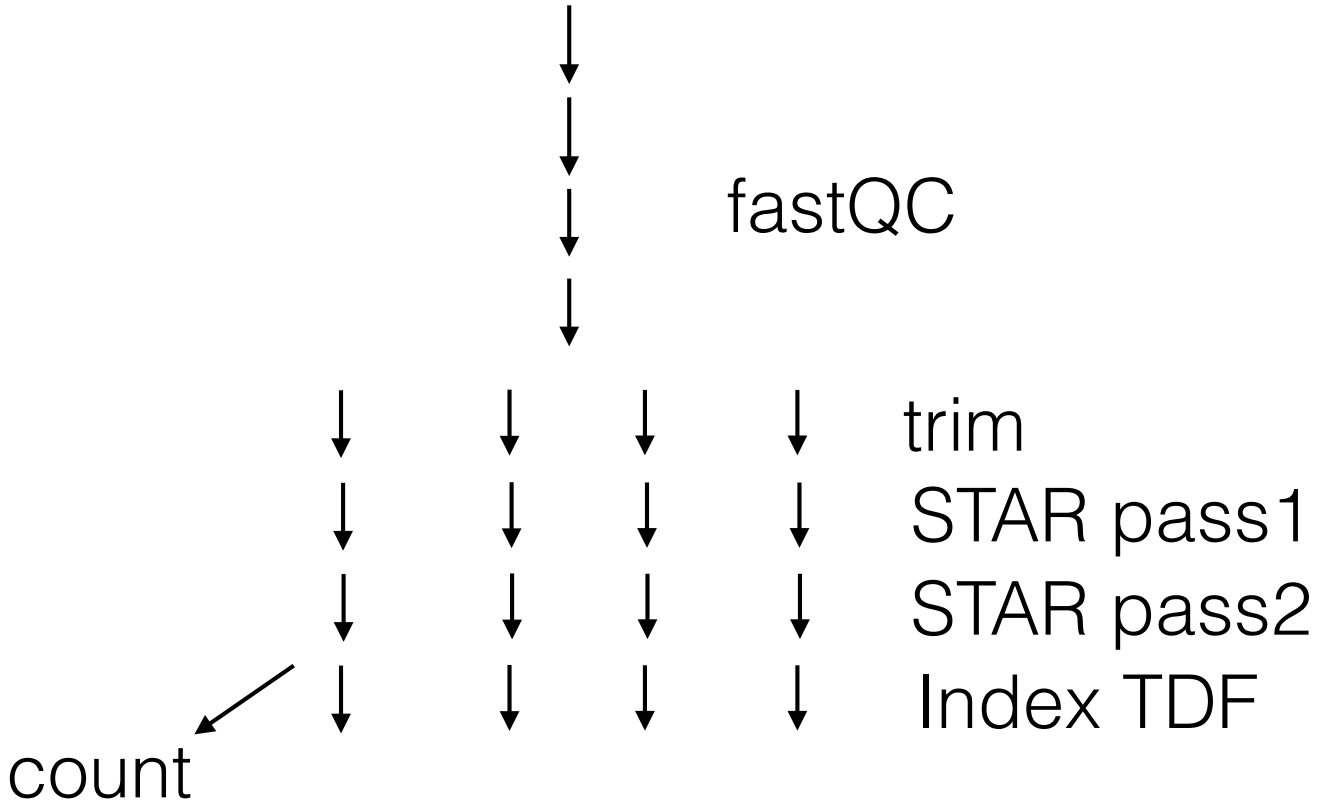
base call accuracy 99% →

http://www.illumina.com/documents/products/technotes/technote_Q-Scores.pdf

less Run02Trim_PE.sh

```
#!/bin/bash

#$ -N Trim_PE
#$ -q standard
#$ -cwd
#$ -e ./logs
#$ -o ./logs
#$ -l h_vmem=12G
#$ -w e

module add trim_galore
module add cutadapt

mkdir -p trim_PE

name_list=(*_R1*.fastq)
Input1=${name_list[$(expr $SGE_TASK_ID - 1)]}
Input2=`echo $Input1 | sed 's/_R1/_R2/'`

trim_galore --quality 20 --stringency 1 --length 30 --paired_end $Input1 $Input2
--output_dir trim_PE/
```

```
qsub Run01FastQC.sh
qsub -hold_jid FastQC -t 1-4 Run02Trim_PE.sh
qsub -hold_jid Trim_PE -t 1-4 Run03Map_pass1.sh
qsub -hold_jid Map_pass1 -t 1-4 Run04Map_pass2.sh
qsub -hold_jid Map_pass2 -t 1-4 Run05Index_TDF.sh
qsub -hold_jid Map_pass2 Run05CountTable.sh 4
```



less Run02Trim_PE.sh

```
#!/bin/bash

#$ -N Trim_PE
#$ -q standard
#$ -cwd
#$ -e ./logs
#$ -o ./logs
#$ -l h_vmem=12G
#$ -w e

module add trim_galore
module add cutadapt

mkdir -p trim_PE

name_list=(*_R1*.fastq)
Input1=${name_list[$(expr $SGE_TASK_ID - 1)]}
Input2=`echo $Input1 | sed 's/_R1/_R2/'`

trim_galore --quality 20 --stringency 1 --length 30 --paired_end $Input1 $Input2
--output_dir trim_PE/
```

in the list of all R1 files `ls *_R1*.fastq`

take the n-th

edit the name for read 2

```
cd trim_PE
head Control-ATCACG_S1_R1_001_val_1.fq
```

```
trim_PE/$ head Control-ATCACG_S1_R1_001_val_1.fq
@NS500735:158:H3JG5AFXX:1:11101:3334:1038 1:N:0:ATCACG
CATCANAAAGGCATTTAGGCGCTGGGCCTCGATCCAGTCCATGGTGCGAACCTCCACGGTAATGCCGCAGGGACACAGA
GCGTCCACGGCGGCCTGGGCAAACATGGTAGGCGTCATGCAGTTGGCGGGGGCATCGCTTAAGCGTCTTGCC
+
AAAAA#EEEEEEEAEEEEEEEEEEEEEEEEEEEEAEEEEEEEEEEEEEEEEEEEEEEEEEAAEEEAAEEEEE/E/
EAEAA<EEEEEEEEEEAEEEAEAEE//E/EAEEEEEEEEEEEEEAAAE/EEEEEEE<A<AAEAE<E<<AA<
@NS500735:158:H3JG5AFXX:1:11101:10840:1038 1:N:0:ATCACG
GTTTGNTTACCACCACATCTATGGGTCTAATGTCCCGGGATCCATGGCGCAGAAAGTCCACCTGATTCGACGCCGTTCG
CTCCTCCACCTCCTGGTTTAAGGCATTCTCAGCATCCTCGTTCTCGATCTCCTCCCTTTGGGCTCCTTTG
+
AAAAA#EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE/EEEEEEEAEEEEEEEEAAEEEEAAAEEEE
EAAAE<<<AEEEA6AE6EAEEEEEEEEEEAEEEEEEA<AEEA<EAEEEEEAEEEAAAEAE//EEAA<<<A
```

```
cd ..
less Run03Map_pass1.sh
```

```
#!/bin/bash

#$ -N Map_pass1
#$ -q standard
#$ -cwd
#$ -e ./logs
#$ -o ./logs
#$ -R y
#$ -l h_vmem=10G
#$ -w e
#$ -pe shm 4

module add STAR

cd trim_PE

star_index=/srv/gsfs0/projects/fuller/workshop/genome/STAR_indexed
gtf=/srv/gsfs0/projects/fuller/workshop/genome/Drosophila_melanogaster.BDGP6.84.gtf

name_list=(*_R1*.fq)
Input1_trimmed=${name_list[$(expr $SGE_TASK_ID - 1)]}
Input2_trimmed=`echo $Input1_trimmed | sed 's/_R1/_R2/'`

mkdir -p star_pass1
pass1_prefix=star_pass1/${Input1_trimmed%.*}_

STAR --runThreadN 4 --runMode alignReads --genomeDir $star_index --alignSJoverhangMin 10
--alignIntronMax 100000 --alignMatesGapMax 100000 --outFilterMismatchNoverLmax 0.04
--readFilesIn $Input1_trimmed $Input2_trimmed --outFileNamePrefix $pass1_prefix
--outSAMtype None
```

genome to map to

annotation of all genes

prefix for output files

go to folder fuller/workshop/genome

genome to map to

```
head Drosophila_melanogaster.BDGP6.dna.toplevel.fa
```

```
genome/$ head Drosophila_melanogaster.BDGP6.dna.toplevel.fa
>2L dna:chromosome chromosome:BDGP6:2L:1:23513712:1 REF
CGACAATGCACGACAGAGGAAGCAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCC
ATATTATAGGGAGAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTTT
GATTTTTTGGCAACCCAAAATGGTGGCGGATGAACGAGATGATAATATATTCAAGTTGCC
GCTAATCAGAAATAAATTCATTGCAACGTTAAATACAGCACAATATATGATCGCGTATGC
GAGAGTAGTGCCAACATATTGTGCTAATGAGTGCCTCTCGTTCTCTGTCTTATATTACCG
CAAACCCAAAAAGACAATACACGACAGAGAGAGAGCAGCGGAGATATTTAGATTGCCT
ATTAAATATGATCGCGTATGCGAGAGTAGTGCCAACATATTGTGCTCTCTATATAATGAC
TGCCTCTCATTCTGTCTTATTTTACCGCAAACCCAAATCGACAATGCACGACAGAGGAAG
CAGAACAGATATTTAGATTGCCTCTCATTTTCTCTCCCATATTATAGGGAGAAATATGAT
```

go to folder fuller/workshop/genome

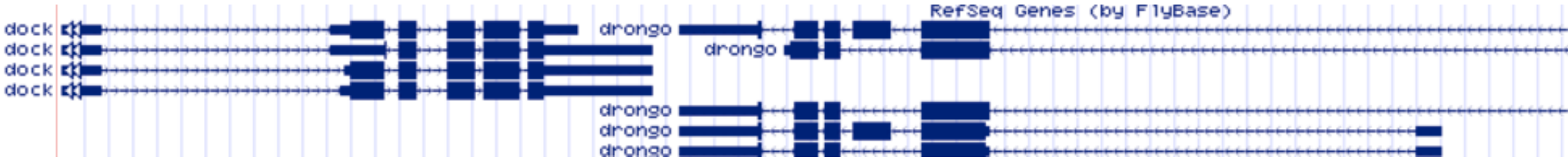
```
less Drosophila_melanogaster.BDGP6.dna.toplevel.fa.fai
```

```
2L      23513712        56      60      61
2R      25286936        23905720        60      61
3L      28110227        49614161        60      61
3R      32079331        78192948        60      61
4       1348131 110806988       60      61
X       23542271        112177642       60      61
Y       3667352 136112338       60      61
dmel_mitochondrion_genome       19517   139840912       60      61
Unmapped_Scaffold_8     88768   139860838       60      61
3Cen_mapped_Scaffold_31 87365   139951177       60      61
Unmapped_Scaffold_4     86267   140040082       60      61
rDNA    76973   140127840       60      61
3Cen_mapped_Scaffold_1  76224   140206185       60      61
Y_mapped_Scaffold_5     73091   140283763       60      61
Y_mapped_Scaffold_9     66731   140358156       60      61
Y_mapped_Scaffold_12    66439   140426085       60      61
Unmapped_Scaffold_17    62570   140493717       60      61
Unmapped_Scaffold_35    57785   140557415       60      61
XY_mapped_Scaffold_7    50625   140616249       60      61
XY_mapped_Scaffold_42   47411   140667805       60      61
Unmapped_Scaffold_28    46986   140716092       60      61
Unmapped_Scaffold_22    45120   140763947       60      61
2Cen_mapped_Scaffold_43 44411   140809910       60      61
Y_mapped_Scaffold_53    44104   140855147       60      61
Unmapped_Scaffold_52    43383   140900072       60      61
Y_mapped_Scaffold_20    39041   140944264       60      61
Unmapped_Scaffold_29    37106   140984041       60      61
3Cen_mapped_Scaffold_36 36913   141021857       60      61
Unmapped_Scaffold_11    36482   141059471       60      61
Y_mapped_Scaffold_18    34521   141096647       60      61
Y_mapped_Scaffold_21    34359   141131829       60      61
X3X4_mapped_Scaffold_6  33320   141166850       60      61
```

head Drosophila_melanogaster.BDGP6.84.gtf

annotation of all genes

```
genome/$ head Drosophila_melanogaster.BDGP6.84.gtf
#!genome-build BDGP6
#!genome-version BDGP6
#!genome-date 2014-07
#!genome-build-accession NCBI:GCA_000001215.4
#!genebuild-last-updated 2014-09
3R      FlyBase gene    722370  722621  .       -       .       gene_id "FBgn0085804";
gene_name "CR41571"; gene_source "FlyBase"; gene_biotype "pseudogene";
3R      FlyBase transcript      722370  722621  .       -       .       gene_id "FBgn00
85804"; transcript_id "FBtr0114258"; gene_name "CR41571"; gene_source "FlyBase"; gene_b
iotype "pseudogene"; transcript_name "CR41571-RA"; transcript_source "FlyBase"; transcr
ipt_biotype "pseudogene";
3R      FlyBase exon    722370  722621  .       -       .       gene_id "FBgn0085804";
transcript_id "FBtr0114258"; exon_number "1"; gene_name "CR41571"; gene_source "FlyBase
"; gene_biotype "pseudogene"; transcript_name "CR41571-RA"; transcript_source "FlyBase"
; transcript_biotype "pseudogene"; exon_id "FBtr0114258-E1";
3R      FlyBase gene    835381  2503907 .       +       .       gene_id "FBgn0267431";
gene_name "CG45784"; gene_source "FlyBase"; gene_biotype "protein_coding";
3R      FlyBase transcript      835381  2503907 .       +       .       gene_id "FBgn02
67431"; transcript_id "FBtr0346770"; gene_name "CG45784"; gene_source "FlyBase"; gene_b
iotype "protein_coding"; transcript_name "CG45784-RA"; transcript_source "FlyBase"; tra
nscript_biotype "protein_coding";
```



less Run04Map_pass2.sh

```
#!/bin/bash

#$ -N Map_pass2
#$ -q standard
#$ -cwd
#$ -e ./logs
#$ -o ./logs
#$ -R y
#$ -l h_vmem=10G
#$ -w e
#$ -pe shm 4

module add STAR

cd trim_PE

star_index=/srv/gsfs0/projects/fuller/workshop/genome/STAR_indexed
gtf=/srv/gsfs0/projects/fuller/workshop/genome/Drosophila_melanogaster.BDGP6.84.gtf

name_list=(*_R1*.fq)
Input1_trimmed=${name_list[$(expr $SGE_TASK_ID - 1)]}
Input2_trimmed=`echo $Input1_trimmed | sed 's/_R1/_R2/'`

junction=`echo star_pass1/*SJ.out.tab`
pass2_prefix=${Input1_trimmed%.*}_

STAR --runThreadN 4 --runMode alignReads --quantMode GeneCounts --genomeDir $star_index
--alignSJoverhangMin 10 --alignIntronMax 100000 --alignMatesGapMax 100000
--outFilterMismatchNoverLmax 0.04 --sjdbFileChrStartEnd $junction
--readFilesIn $Input1_trimmed $Input2_trimmed --outSAMtype BAM Unsorted SortedByCoordinate
--limitBAMsortRAM 10000000000 --outFileNamePrefix $pass2_prefix
```

← include new splice junctions found in pass 1

What would a bam/sam file contain?

go to fuller/workshop/rna_copy/trim_PE folder

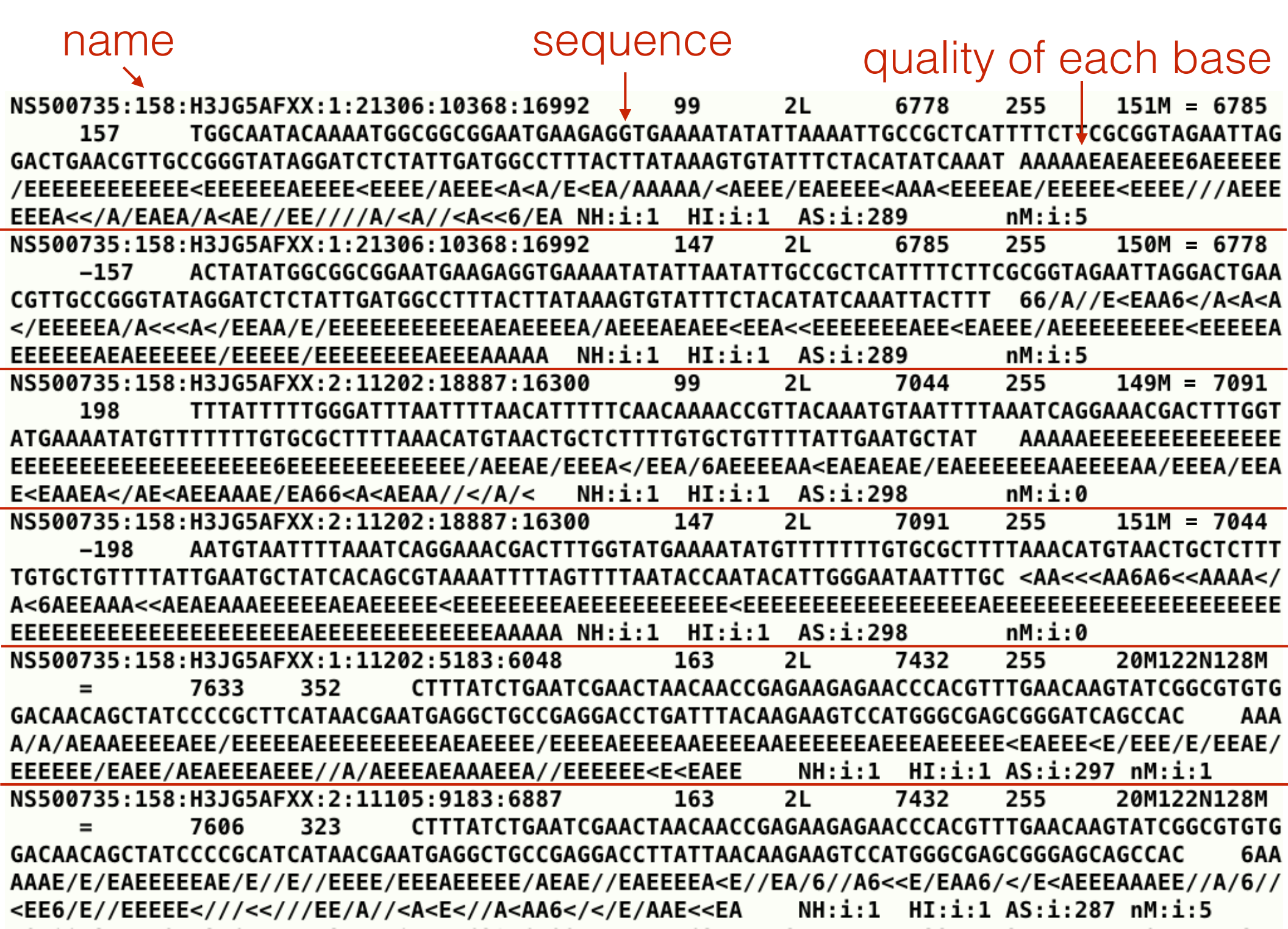
```
head Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.bam
```

[up], change the suffix

```
head Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.sam
```

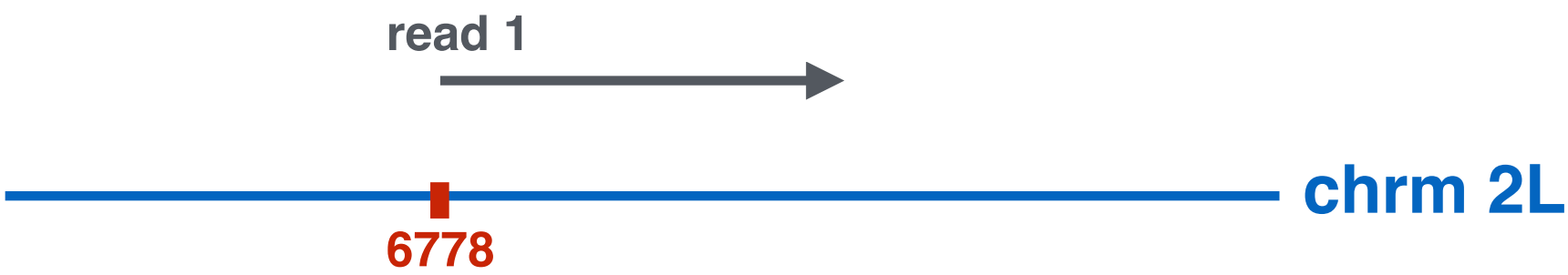
module add samtools

samtools view Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.bam > Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.sam



https://broadinstitute.github.io/picard/explain-flags.html

```
NS500735:158:H3JG5AFXX:1:21306:10368:16992      99      2L      6778    255     151M = 6785
     157    TGGCAATACAAAATGGCGGCGGAATGAAGAGGTGAAAATATATTAAAATTGCCGCTCATTTTCTTCGCGGTAGAATTAG
GACTGAACGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAAT AAAAAEAEAEEE6AEEEEE
/EEEEEEEEEEEE<EEEEEEAEEEE<EEEE/AEEE<A<A/E<EA/AAAAA/<AEEE/EAEEEE<AAA<EEEEAE/EEEEE<EEEE///AEEE
EEEA<</A/EAEA/A<AE//EE////A/<A//<A<<6/EA NH:i:1  HI:i:1  AS:i:289       nM:i:5
NS500735:158:H3JG5AFXX:1:21306:10368:16992      147     2L      6785    255     150M = 6778
     -157    ACTATATGGCGGCGGAATGAAGAGGTGAAAATATATTAATATTGCCGCTCATTTTCTTCGCGGTAGAATTAGGACTGAA
CGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAATTACTTT  66/A//E<EAA6</A<A<A
</EEEEEA/A<<<A</EEAA/E/EEEEEEEEEEEAEAEEEEA/AEEEAEAEE<EEA<<EEEEEEEAEE<EAEEE/AEEEEEEEEE<EEEEEA
EEEEEEAEAEEEEEE/EEEEE/EEEEEEEEAEEEAAAAA  NH:i:1  HI:i:1  AS:i:289       nM:i:5
```

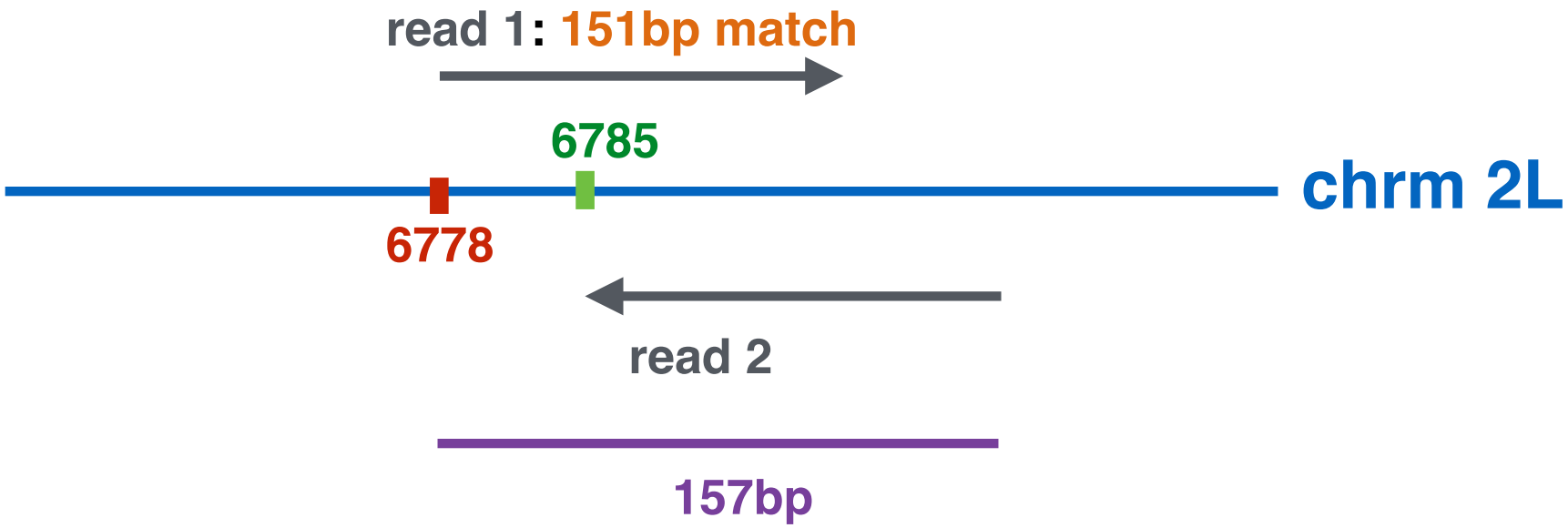


```
NS500735:158:H3JG5AFXX:1:21306:10368:16992      99      2L      6778    255     151M = 6785
     157     TGGCAATACAAAATGGCGGCGGAATGAAGAGGTGAAAATATATTAAAATTGCCGCTCATTTTCTTCGCGGTAGAATTAG
GACTGAACGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAAT AAAAAEAEAEEE6AEEEEE
/EEEEEEEEEEE<EEEEEEAEEEE<EEEE/AEEE<A<A/E<EA/AAAAA/<AEEE/EAEEEE<AAA<EEEEAE/EEEEE<EEEE///AEEE
EEEA<</A/EAEA/A<AE//EE////A/<A//<A<<6/EA NH:i:1  HI:i:1  AS:i:289       nM:i:5
NS500735:158:H3JG5AFXX:1:21306:10368:16992      147     2L      6785    255     150M = 6778
     -157    ACTATATGGCGGCGGAATGAAGAGGTGAAAATATATTAATATTGCCGCTCATTTTCTTCGCGGTAGAATTAGGACTGAA
CGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAATTACTTT  66/A//E<EAA6</A<A<A
</EEEEEA/A<<<A</EEAA/E/EEEEEEEEEEEAEAEEEEA/AEEEAEAEE<EEA<<EEEEEEEAEE<EAEEE/AEEEEEEEEE<EEEEEA
EEEEEEAEAEEEEEE/EEEEE/EEEEEEEEAEEEAAAAA  NH:i:1  HI:i:1  AS:i:289       nM:i:5
NS500735:158:H3JG5AFXX:2:11202:18887:16300      99      2L      7044    255     149M = 7091
     198     TTTATTTTTGGGATTTAATTTTAACATTTTTCAACAAAACCGTTACAAATGTAATTTTAAATCAGGAAACGACTTTGGT
ATGAAAATATGTTTTTTTGTGCGCTTTTAAACATGTAACTGCTCTTTTGTGCTGTTTTATTGAATGCTAT   AAAAAEEEEEEEEEEEEEE
EEEEEEEEEEEEEEEEEEE6EEEEEEEEEEEE/AEEAE/EEEA</EEA/6AEEEEAA<EAEAEAE/EAEEEEEEAAEEEEAA/EEEA/EEA
E<EAAEA</AE<AEEAAAE/EA66<A<AEAA//</A/<   NH:i:1  HI:i:1  AS:i:298       nM:i:0
NS500735:158:H3JG5AFXX:2:11202:18887:16300      147     2L      7091    255     151M = 7044
     -198    AATGTAATTTTAAATCAGGAAACGACTTTGGTATGAAAATATGTTTTTTTGTGCGCTTTTAAACATGTAACTGCTCTTT
TGTGCTGTTTTATTGAATGCTATCACAGCGTAAAATTTTAGTTTTAATACCAATACATTGGGAATAATTTGC <AA<<<AA6A6<<AAAA</
A<6AEEAAA<<AEAEAAAEEEEEAEAEEEEE<EEEEEEEEAEEEEEEEEEEE<EEEEEEEEEEEEEEEEEAEEEEEEEEEEEEEEEEEE
EEEEEEEEEEEEEEEEEEEEEAEEEEEEEEEEEEEAAAAA NH:i:1  HI:i:1  AS:i:298       nM:i:0
NS500735:158:H3JG5AFXX:1:11202:5183:6048        163     2L      7432    255     20M122N128M
     =       7633    352     CTTTATCTGAATCGAACTAACAACCGAGAAGAGAACCCACGTTTGAACAAGTATCGGCGTGTG
GACAACAGCTATCCCCGCTTCATAACGAATGAGGCTGCCGAGGACCTGATTTACAAGAAGTCCATGGGCGAGCGGGATCAGCCAC    AAA
A/A/AEAAEEEEAEE/EEEEEAEEEEEEEEEAEAEEEE/EEEEAEEEEAAEEEEAAEEEEEEAEEEAEEEEE<EAEEE<E/EEE/E/EEAE/
EEEEEE/EAEE/AEAEEEAEEE//A/AEEEAEAAAEEA//EEEEEE<E<EAEE    NH:i:1  HI:i:1 AS:i:297 nM:i:1
NS500735:158:H3JG5AFXX:2:11105:9183:6887        163     2L      7432    255     20M122N128M
     =       7606    323     CTTTATCTGAATCGAACTAACAACCGAGAAGAGAACCCACGTTTGAACAAGTATCGGCGTGTG
GACAACAGCTATCCCCGCATCATAACGAATGAGGCTGCCGAGGACCTTATTAACAAGAAGTCCATGGGCGAGCGGGAGCAGCCAC    6AA
AAAE/E/EAEEEEEAE/E//E//EEEE/EEEAEEEEE/AEAE//EAEEEEA<E//EA/6//A6<<E/EAA6/</E<AEEEAAAEE//A/6//
<EE6/E//EEEEE<///<<///EE/A//<A<E<//A<AA6</</E/AAE<<EA    NH:i:1  HI:i:1 AS:i:287 nM:i:5
```

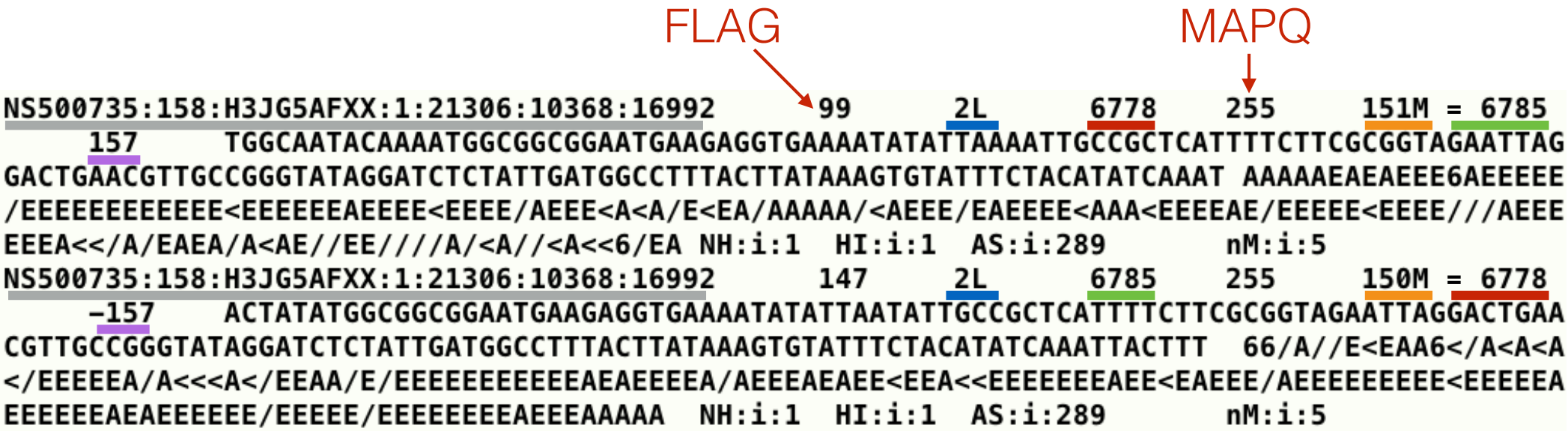
https://broadinstitute.github.io/picard/explain-flags.html

FLAG

```
NS500735:158:H3JG5AFXX:1:21306:10368:16992      99      2L      6778    255     151M = 6785
     157    TGGCAATACAAAATGGCGGCGGAATGAAGAGGTGAAAATATATTAAAATTGCCGCTCATTTTCTTCGCGGTAGAATTAG
GACTGAACGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAAT AAAAAEAEAEEE6AEEEEE
/EEEEEEEEEEE<EEEEEEAEEEE<EEEE/AEEE<A<A/E<EA/AAAAA/<AEEE/EAEEEE<AAA<EEEEAE/EEEEE<EEEE///AEEE
EEEA<</A/EAEA/A<AE//EE////A/<A//<A<<6/EA NH:i:1  HI:i:1  AS:i:289       nM:i:5
NS500735:158:H3JG5AFXX:1:21306:10368:16992      147     2L      6785    255     150M = 6778
     -157   ACTATATGGCGGCGGAATGAAGAGGTGAAAATATATTAATATTGCCGCTCATTTTCTTCGCGGTAGAATTAGGACTGAA
CGTTGCCGGGTATAGGATCTCTATTGATGGCCTTTACTTATAAAGTGTATTTCTACATATCAAATTACTTT  66/A//E<EAA6</A<A<A
</EEEEEA/A<<<A</EEAA/E/EEEEEEEEEEEAEAEEEEA/AEEEAEAEE<EEA<<EEEEEEEAEE<EAEEE/AEEEEEEEEE<EEEEEA
EEEEEEAEAEEEEEE/EEEEE/EEEEEEEEAEEEAAAAA  NH:i:1  HI:i:1  AS:i:289       nM:i:5
```



https://broadinstitute.github.io/picard/explain-flags.html



Every mapper uses their own system of MAPQ

STAR:

255 = unique mapping (50 for Tophat)

3 = maps to 2 locations in the target

2 = maps to 3 locations

1 = maps to 49 locations

0 = maps to 10 or more locations

`Run05Index_TDF.sh`

- Collect mapping statistics through Flagstat
- Index the .bam for browsing. Loading .bam directly to UCSC or IGV is possible, but slow.
- Convert .bam to .bed for UCSC genome browser
- Convert .bam to .tdf for IGV

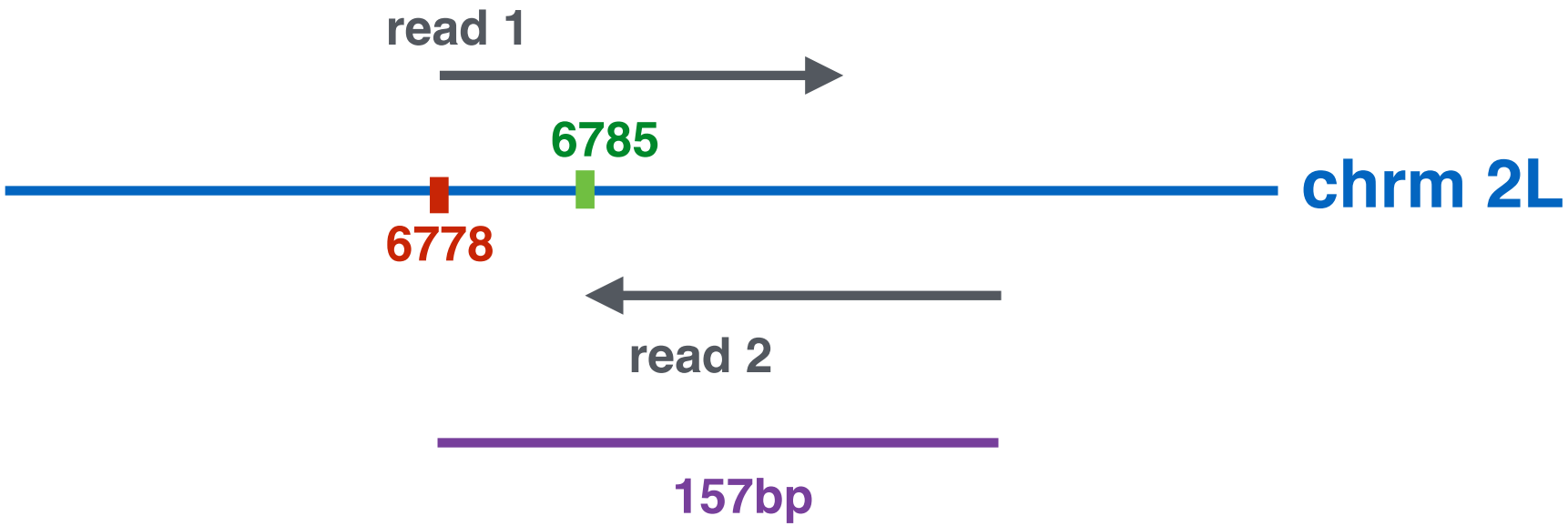
less Run05Index_TDF.sh

```
#!/bin/bash

#$ -N TDF
#$ -q standard
#$ -cwd
#$ -e ./logs
#$ -o ./logs
#$ -R y
#$ -l h_vmem=12G
#$ -w e

module add igvtools
module add samtools
module add bedtools

chrm_size=/srv/gsfs0/projects/fuller/workshop/genome/Drosophila_melanogaster.BDGP6.dna
.toplevel.fa.fai

cd trim_PE

name_list=(*.sortedByCoord.out.bam)
Input=${name_list[$(expr $SGE_TASK_ID - 1)]}

Output1=`echo $Input | sed 's/.bam/.tdf/'`
Output2=`echo $Input | sed 's/.bam/.bed/'`

samtools index $Input   # index the bam file for fast access
samtools flagstat $Input   # statistics on the bam file
igvtools count $Input $Output1 $chrm_size   # convert to tdf file
bedtools bamtobed -i $Input > $Output2    # convert to bed file
```

```
head Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.bed
```

```
trim_PE/$ head Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.bed
2L      6777    6928    NS500735:158:H3JG5AFXX:1:21306:10368:16992/1    255     +
2L      6784    6934    NS500735:158:H3JG5AFXX:1:21306:10368:16992/2    255     -
2L      7043    7192    NS500735:158:H3JG5AFXX:2:11202:18887:16300/1    255     +
2L      7090    7241    NS500735:158:H3JG5AFXX:2:11202:18887:16300/2    255     -
2L      7431    7701    NS500735:158:H3JG5AFXX:1:11202:5183:6048/2      255     +
2L      7431    7701    NS500735:158:H3JG5AFXX:2:11105:9183:6887/2      255     +
2L      7431    7579    NS500735:158:H3JG5AFXX:3:11505:15629:6188/2     255     +
2L      7433    7675    NS500735:158:H3JG5AFXX:1:11102:12511:4424/2     255     +
2L      7434    7581    NS500735:158:H3JG5AFXX:3:11505:15629:6188/1     255     -
2L      7447    7701    NS500735:158:H3JG5AFXX:3:11406:7820:13247/2     255     +
```

start from 0



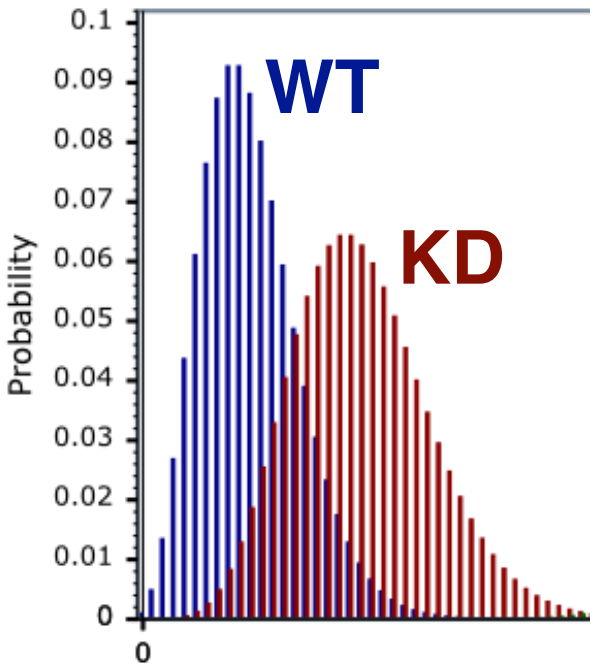
What would .tdf file contain?

```
head Control-ATCACG_S1_R1_001_val_1_Aligned.sortedByCoord.out.tdf
```

Differential gene expression analysis

- DESeq2 or EdgeR:
  - For each gene,
    - Estimate the mean and variance distribution (negative binomial)
    - Test for significant differences
  - Biological repeat is very important for properly estimating the variance

FBgc000001




```
Run05CountTable.sh
Run06DE.sh
DESeq2_annotate.R
```

```
head -n50 Control-ATCACG_S1_R1_001_val_1_ReadsPerGene.out.tab
N_unmapped      4344073 4344073 4344073
N_multimapping
N_noFeature                                    1539739
N_ambiguous
FBgn0085804     0       0       0
FBgn0267431     374     2       372
FBgn0039987     0       0       0
FBgn0058182     0       0       0
FBgn0267430     1005    29      976
FBgn0266747     171     2       169
FBgn0086917     0       0       0
FBgn0010247     1624    30      1594
FBgn0086378     1818    11      1807
FBgn0263977     5501    16      5485
FBgn0069923     573     0       573
FBgn0039955     2746    9       2737
FBgn0259821     213     10      203
FBgn0027341     363     11      362
FBgn0085812     241     147     104
FBgn0058198     5       1       4
FBgn0037213     58      2       56
FBgn0053294     8       1       7
FBgn0000500     6       0       6
FBgn0037215     527     329     231
FBgn0037217     88      17      105
FBgn0037218     1749    394     1464
FBgn0051516     513     3       516
FBgn0261436     400     17      488
FBgn0037220     2390    14      2376
FBgn0015331     2081    18      2065
```

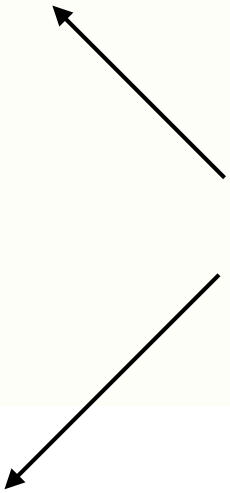
un- strnd | forward strnd | reverse strnd

count table generated by STAR

`Run05CountTable.sh` generates 2 things:

`head Control-ATCACG_S1_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt`

```
trim_PE/$ head Control-ATCACG_S1_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt
FBgn0000003     697
FBgn0000008     317
FBgn0000014     439
FBgn0000015     168
FBgn0000017     797
FBgn0000018     673
FBgn0000022     1
FBgn0000024     190
FBgn0000028     1459
FBgn0000032     1581
```



Inputs for DESeq2

`Run06DE.sh`
`DESeq2_annotate.R`

`less sampleTable.txt`

```
sampleName,fileName,condition
Control-ATCACG_S1_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,Control-ATCACG_S1_
R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,control
ctrl-02-ATCACG_S1_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,ctrl-02-ATCACG_S1_
R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,control
KD-CGATGT_S2_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,KD-CGATGT_S2_R1_001_val
_1_ReadsPerGene.out.tab.srt.4.txt,control
kmg-KD-02-CGATGT_S2_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,kmg-KD-02-CGATGT
_S2_R1_001_val_1_ReadsPerGene.out.tab.srt.4.txt,control
```

change to any name w/o space

```
head DE_results_all_annotated.txt
```

A1

	A	B	C	D	E	F	G	H	I
1	X	baseMean	log2FoldChar	lfcSE	stat	pvalue	padj	V1	V3
2	FBgn003788	19671.3737	-8.5659473	0.19749093	-43.373878	0	0	scpr-B	CG17210
3	FBgn003788	17910.3814	-8.3480306	0.2108231	-39.597324	0	0	scpr-A	CG5207
4	FBgn003787	21934.1957	-6.2694633	0.17549685	-35.724079	1.67E-279	9.07E-276	scpr-C	CG5106
5	FBgn003800	4417.66542	-6.2879284	0.2091285	-30.067295	1.30E-198	5.28E-195	CG3942	CG3942
6	FBgn003396	6829.82512	-5.7727289	0.2017897	-28.607648	5.40E-180	1.76E-176	CG10202	CG10202
7	FBgn004302	15672.956	-5.8215226	0.20752302	-28.052419	3.73E-173	1.01E-169	Adgf-A2	CG32178
8	FBgn003949	6496.69201	-6.0949112	0.22516896	-27.068168	2.33E-161	5.43E-158	CG6059	CG6059
9	FBgn003825	7585.46935	-5.8324434	0.21604113	-26.996913	1.61E-160	3.27E-157	CG7362	CG7362
10	FBgn005035	6133.99344	-5.9543347	0.22170913	-26.856515	7.08E-159	1.28E-155	CG30356	CG30356
11	FBgn006636	3351.12023	6.30354362	0.23851546	26.4282388	6.49E-154	1.06E-150	dyl	CG15013
12	FBgn002856	5317.5358	-5.8280045	0.22362108	-26.061964	9.85E-150	1.46E-146	robl37BC	CG15171
13	FBgn002889	8196.40001	-6.9896726	0.26931534	-25.953489	1.66E-148	2.25E-145	CG4161	CG4161
14	FBgn005206	11224.758	-5.1861169	0.20024322	-25.899088	6.82E-148	8.54E-145	CG32061	CG32061
15	FBgn003395	6217.71484	-5.1494945	0.20133158	-25.577183	2.74E-144	3.18E-141	Adgf-E	CG10143
16	FBgn003005	4103.11368	-5.1727278	0.20273384	-25.514872	1.35E-143	1.46E-140	CG12111	CG12111
17	FBgn002908	5881.89876	-4.528828	0.17773907	-25.480205	3.27E-143	3.32E-140	gom	CG6727
18	FBgn003670	9850.10739	-6.6787357	0.26286298	-25.40767	2.07E-142	1.99E-139	CG13725	CG13725
19	FBgn003526	7798.18779	-4.9252317	0.19587057	-25.145338	1.59E-139	1.44E-136	CG18171	CG18171
20	FBgn003711	6598.40498	-4.5441228	0.18163412	-25.018003	3.89E-138	3.34E-135	CG11249	CG11249
21	FBgn005245	1989.24751	-7.2346906	0.29333329	-24.663722	2.62E-134	2.13E-131	CG32457	CG32457
22	FBgn026281	4000.43284	-5.6783901	0.23119866	-24.560653	3.33E-133	2.58E-130	CG43183	CG43183
23	FBgn002883	4382.96851	-4.7132717	0.19219814	-24.522982	8.40E-133	5.95E-130	CSN1a	CG4697
24	FBgn003807	9128.51586	-5.0740231	0.20689984	-24.524055	8.18E-133	5.95E-130	CG14391	CG14391
25	FBgn003813	2617.28974	7.57013122	0.30969547	24.4437908	5.86E-132	3.97E-129	Osi22	CG8644
26	FBgn026581	4660.50441	-4.4785532	0.18456129	-24.26594	4.49E-130	2.92E-127	CR44601	CR44601
27	FBgn026747	2421.2467	-5.7131279	0.23619369	-24.188317	2.95E-129	1.85E-126	CR45821	CR45821
28	FBgn003497	4468.5994	-5.0223547	0.20784927	-24.163447	5.39E-129	3.25E-126	CG13564	CG13564

head DE_results_500_summary.txt

X	baseMean	log2FoldChar	lfcSE	stat	pvalue	padj	V1	V3	V2.y
FBgn003788	19697.6408	-8.5703111	0.19602982	-43.719425	0	0	scpr-B	CG17210	The gene SCP-containing protein
FBgn003788	17935.0381	-8.3522964	0.2098653	-39.798368	0	0	scpr-A	CG5207	The gene SCP-containing protein
FBgn003787	21964.0795	-6.2737519	0.17386547	-36.083944	4.05E-285	2.12E-281	scpr-C	CG5106	The gene SCP-containing protein
FBgn003800	4423.47013	-6.2921111	0.20789323	-30.266071	3.21E-201	1.26E-197	CG3942	CG3942	This gene is referred to in FlyBas
FBgn003396	6838.58174	-5.7768724	0.2005146	-28.810234	1.60E-182	5.01E-179	CG10202	CG10202	This gene is referred to in FlyBas
FBgn004302	15692.852	-5.8255974	0.20658665	-28.199292	5.97E-175	1.56E-171	Adgf-A2	CG32178	The gene Adenosine deaminase-
FBgn003949	6505.09154	-6.0989701	0.22452229	-27.164208	1.72E-162	3.85E-159	CG6059	CG6059	This gene is referred to in FlyBas
FBgn003825	7595.12061	-5.8365035	0.21521891	-27.118916	5.89E-162	1.15E-158	CG7362	CG7362	This gene is referred to in FlyBas
FBgn005035	6141.68565	-5.9583613	0.22092123	-26.970524	3.28E-160	5.71E-157	CG30356	CG30356	This gene is referred to in FlyBas
FBgn006636	3346.82437	6.30019881	0.23764193	26.51131	7.18E-155	1.13E-151	dyl	CG15013	The gene dusky-like is referred t
FBgn002856	5324.1614	-5.8320182	0.22284769	-26.170422	5.77E-151	8.23E-148	robl37BC	CG15171	The gene robl37BC is referred to
FBgn005206	11238.9281	-5.1902177	0.19911581	-26.066326	8.79E-150	1.15E-146	CG32061	CG32061	This gene is referred to in FlyBas
FBgn002889	8206.37614	-6.9934054	0.26950547	-25.94903	1.86E-148	2.25E-145	CG4161	CG4161	This gene is referred to in FlyBas
FBgn002908	5889.37631	-4.5330157	0.17597879	-25.758874	2.56E-146	2.87E-143	gom	CG6727	The gene gomdanji is referred to
FBgn003395	6225.51831	-5.1536	0.20009105	-25.756274	2.74E-146	2.87E-143	Adgf-E	CG10143	The gene Adenosine deaminase-
FBgn003005	4108.23806	-5.1768394	0.20138346	-25.706378	9.92E-146	9.72E-143	CG12111	CG12111	This gene is referred to in FlyBas
FBgn003670	9862.03422	-6.6824868	0.26297624	-25.410991	1.91E-142	1.76E-139	CG13725	CG13725	This gene is referred to in FlyBas
FBgn003526	7808.17416	-4.9293758	0.1946421	-25.325332	1.68E-141	1.46E-138	CG18171	CG18171	This gene is referred to in FlyBas
FBgn003711	6606.55466	-4.5482666	0.17987251	-25.286057	4.55E-141	3.75E-138	CG11249	CG11249	This gene is referred to in FlyBas

All you actually need to do

- Log in to the cluster
- With FTP software upload data and scripts
- On command line, navigate to the folder where the data is
- Paste in the following lines, hit ‘Enter’

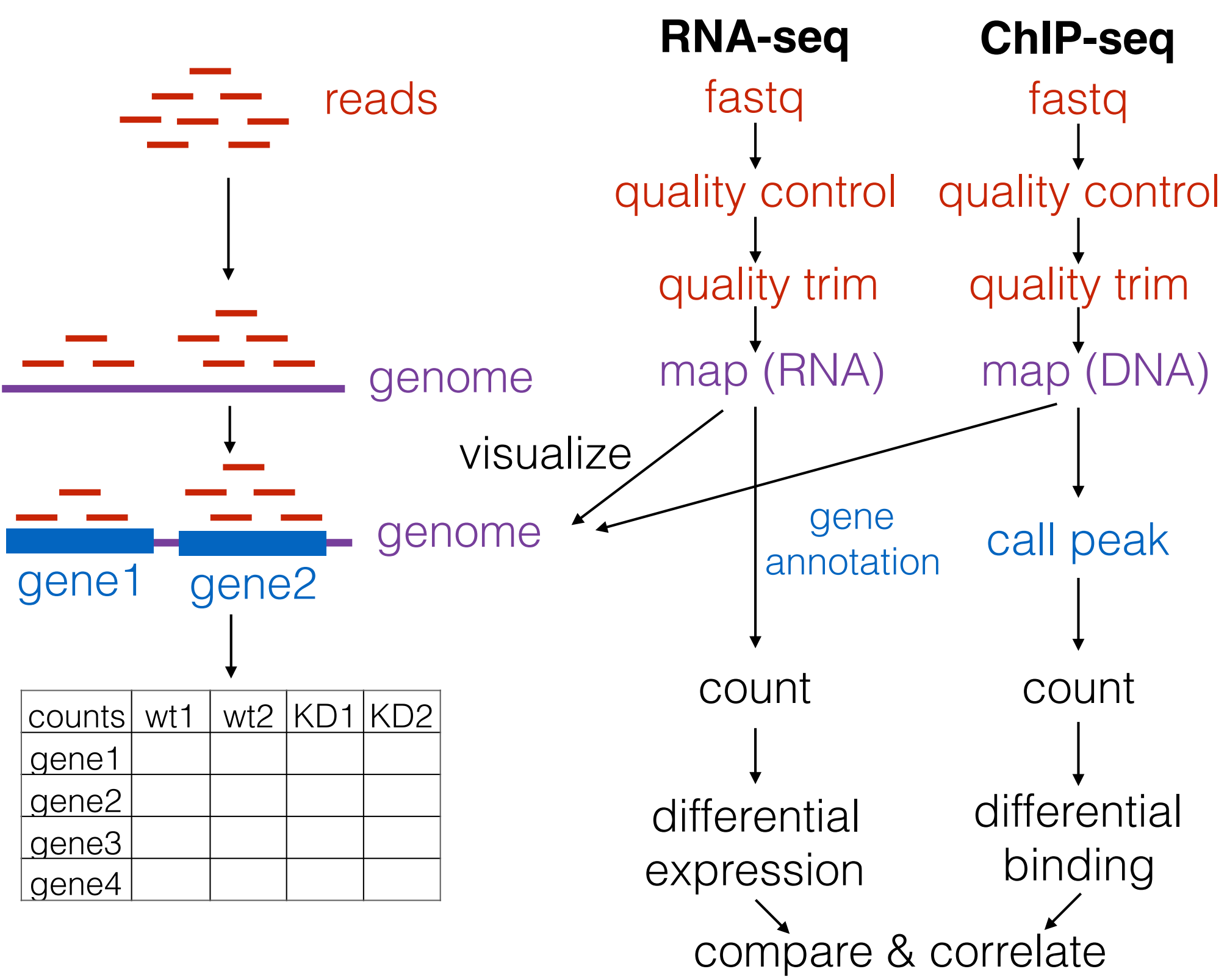
```
qsub Run01FastQC.sh
qsub -hold_jid FastQC -t 1-X Run02Trim_PE.sh
qsub -hold_jid Trim_PE -t 1-X Run03Map_pass1.sh
qsub -hold_jid Map_pass1 -t 1-X Run04Map_pass2.sh
qsub -hold_jid Map_pass2 -t 1-X Run05Index_TDF.sh
qsub -hold_jid Map_pass2 Run05CountTable.sh Y
(modify sampleTable.txt)
qsub Run06DE.sh
```

* X is total number of library you have
* Y is the strandedness of the RNA-seq
* **For paired-end sequencing only**

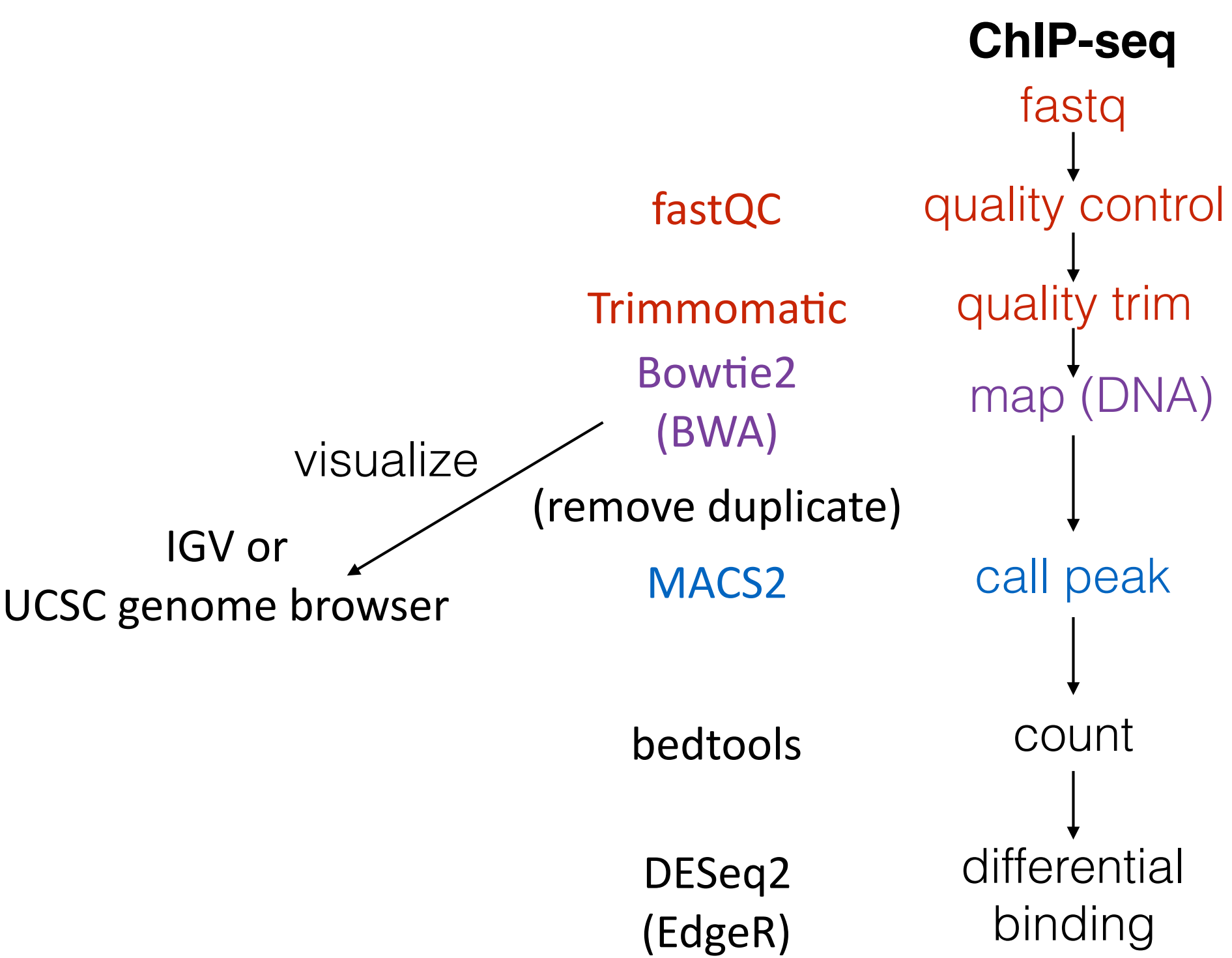
Jamie's pipeline

- For RNA-seq and ChIP-seq
- Have options for different sequencing types
- With added quality control steps along the way
- Put all file names in meta.txt, and run:

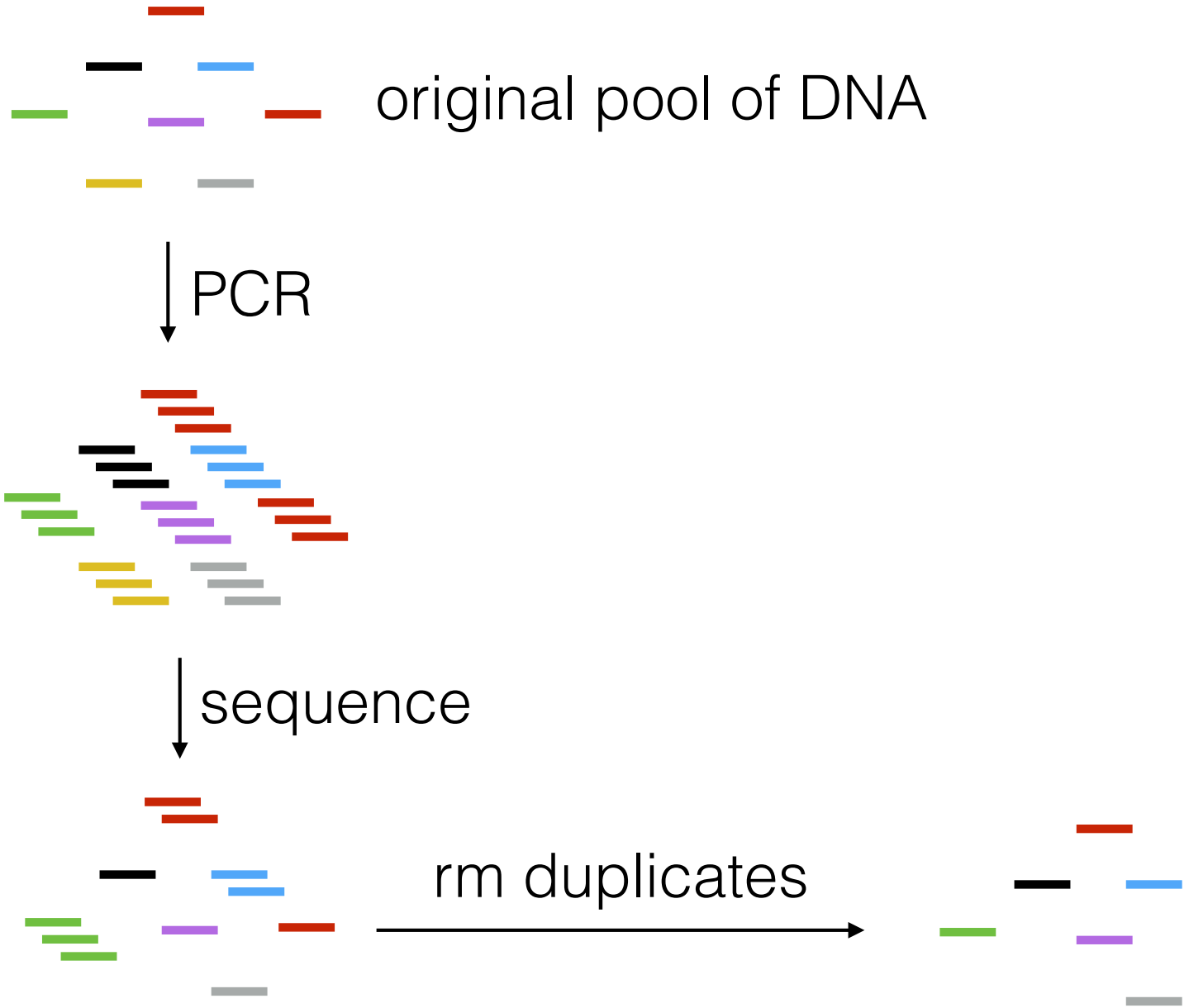
```
seq_pipelines-master/$ python RNAseq_pipeline.py -h
usage: RNAseq_pipeline.py [-h] -m META [-o OUTPUT] [-d DIRECTORY]
                          [-c {scg3,cho_oro}] [-g GENOME] [-p] [-l READLEN]
                          [--no_trim] [-t TEMP]
                          [--bowtie_mismatch BOWTIE_MISMATCH] [-n]

batch pipeline for mapping ChIP-seq data

optional arguments:
  -h, --help            show this help message and exit
  -m META, --meta META  table containing new_filename<tab>index, the index can
                        be any text used to uniquely identify the fastq files
  -o OUTPUT, --output OUTPUT
                        directory to create new files in, default is current
                        working directory
  -d DIRECTORY, --directory DIRECTORY
                        directory storing fastq files, default current working
                        directory
  -c {scg3,cho_oro}, --machine {scg3,cho_oro}
                        speficies the paths for genome.fa, chr sizes and
                        bowtie2 index for a given system
  -g GENOME, --genome GENOME
                        specifies the genome to be used for mapping
  -p, --paired          flag specifies paired end, without flag assumes single
                        end reads
  -l READLEN, --readlen READLEN
                        specify the length of reads, default 50 bp
  --no_trim             flag to prevent trimming reads using Trimmomatic prior
                        to mapping, default trimming
  -t TEMP, --temp TEMP  specifies the temporary folder path, otherwise will
                        use current working directory
  --bowtie_mismatch BOWTIE_MISMATCH
                        number of mismatches for bowtie2
  -n, --norun           only generate the bash files but do not run them
```



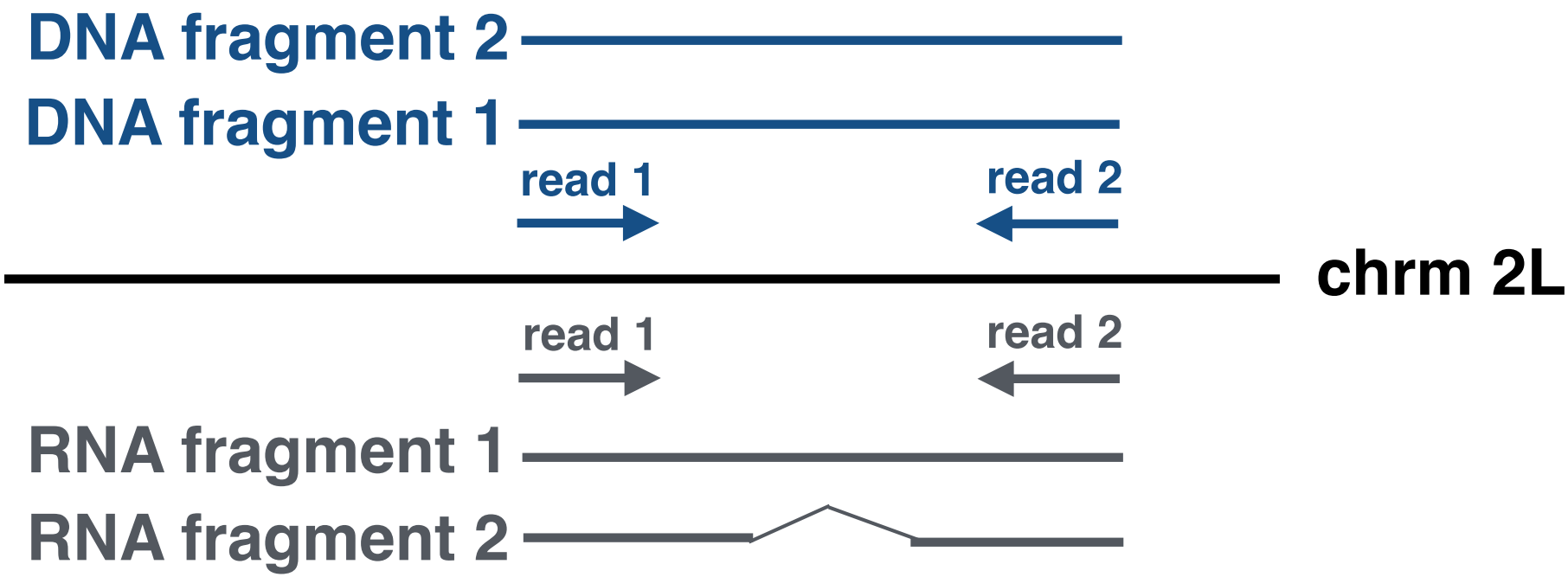
counts	wt1	wt2	KD1	KD2
gene1				
gene2				
gene3				
gene4				


ChIP-seq
fastq
fastQC
quality control
Trimmomatic
quality trim
Bowtie2
(BWA)
map (DNA)
visualize
(remove duplicate)
IGV or
UCSC genome browser
MACS2
call peak
bedtools
count
DESeq2
(EdgeR)
differential
binding

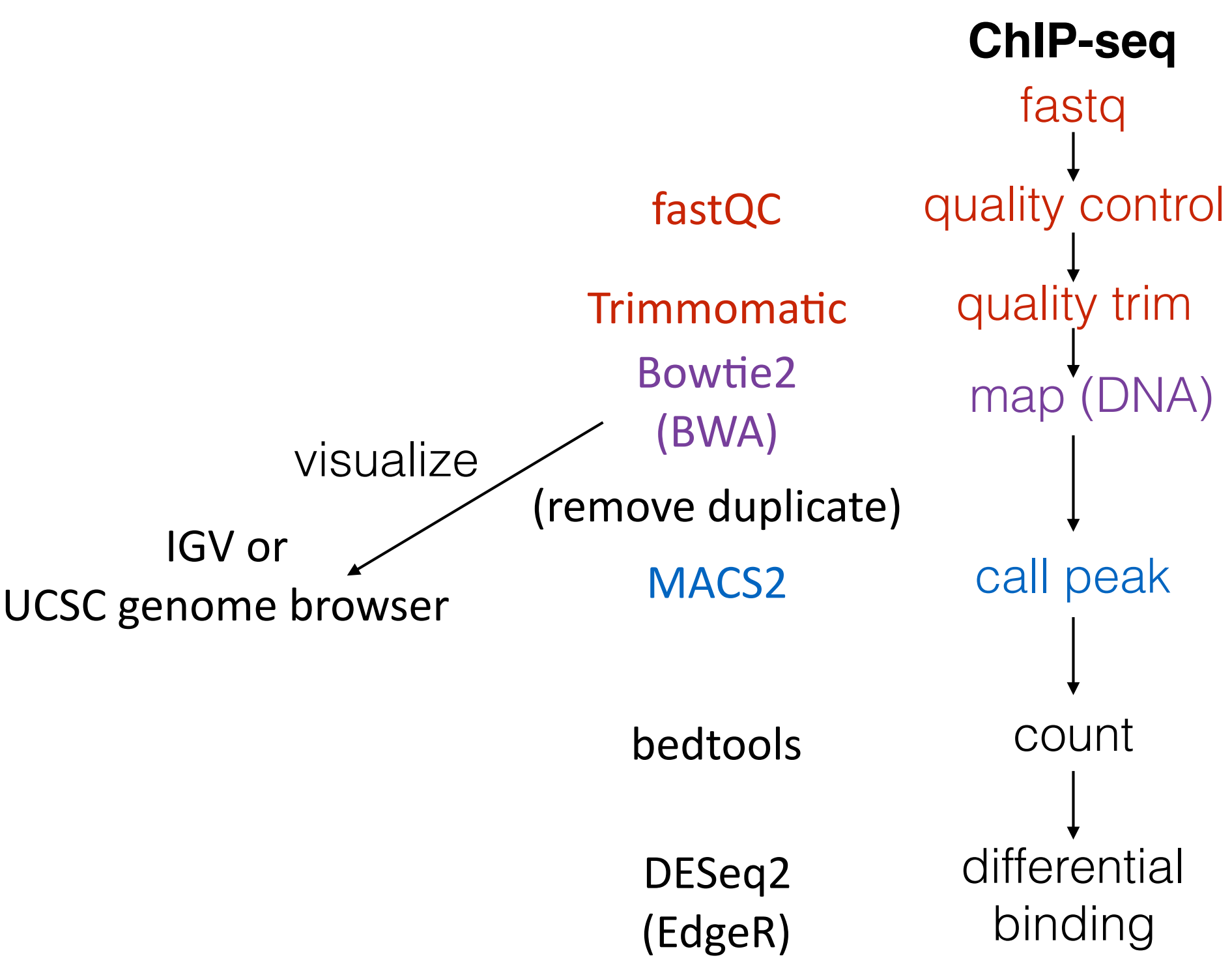
PCR duplicates



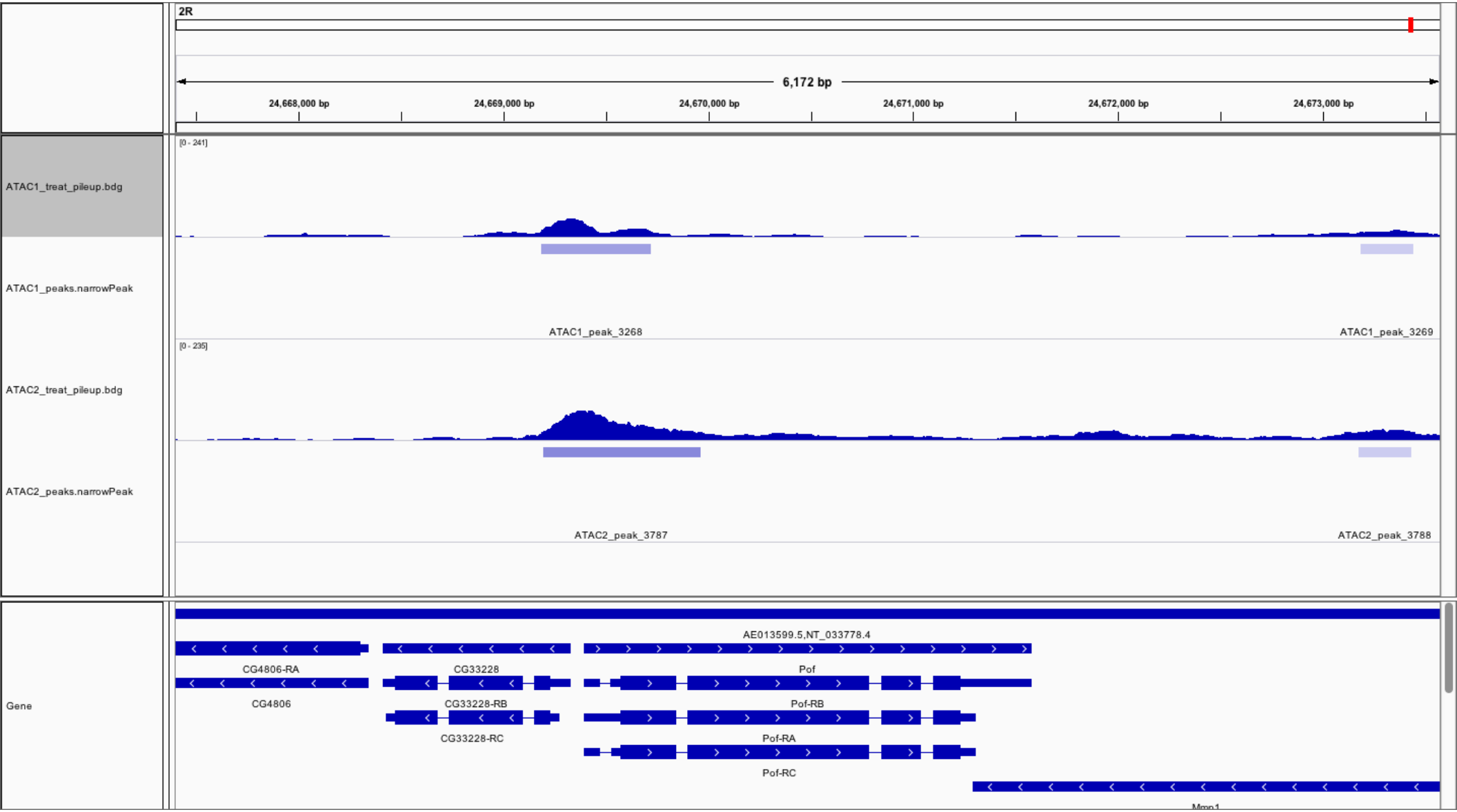
Remove duplicate?



- Still some debate on best practice
- Suggested: do it for DNA-seq, not for RNA-seq
- MACS2 does this by default
- Bottom line: do as few PCR cycles as possible

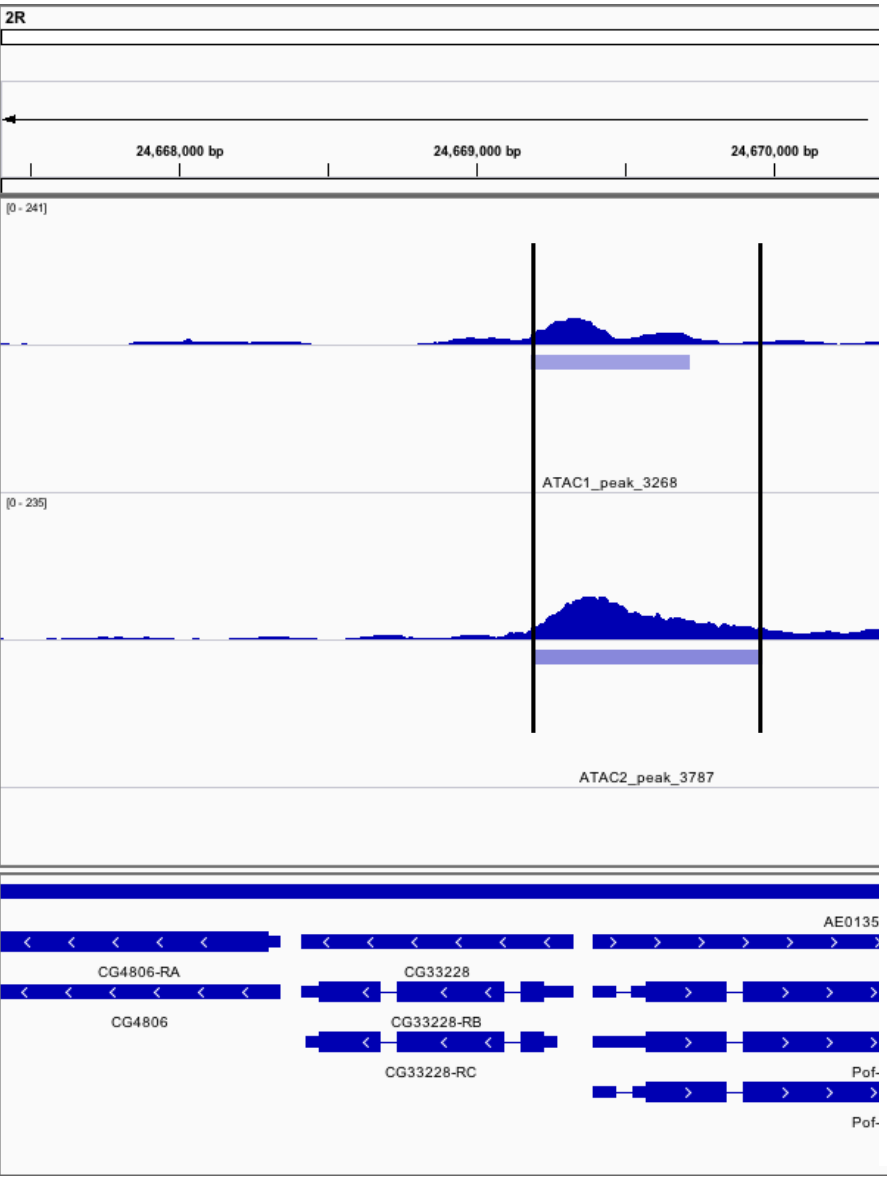

ChIP-seq
fastq
fastQC
quality control
Trimmomatic
quality trim
Bowtie2
(BWA)
map (DNA)
visualize
(remove duplicate)
IGV or
UCSC genome browser
MACS2
call peak
bedtools
count
DESeq2
(EdgeR)
differential
binding

MACS2: identify peaks



The peak files are .bed files

```
macs2/$ head ATAC1_peaks.narrowPeak
2L      5497    6046    ATAC1_peak_1    433     .       11.46617        46.78581        43.37067        358
2L      18828   19059   ATAC1_peak_2    74      .       4.04531 9.48651 7.42410 143
2L      21661   21872   ATAC1_peak_3    94      .       4.76190 11.58204        9.41230 117
2L      66661   67542   ATAC1_peak_4    1136    .       13.82784        118.74139       113.64207       220
2L      72161   73588   ATAC1_peak_5    263     .       6.40177 29.24053        26.35966        1049
2L      87322   87837   ATAC1_peak_6    247     .       7.61589 27.56848        24.74527        122
2L      94625   95059   ATAC1_peak_7    149     .       5.90278 17.43558        14.99410        65
2L      102209  102465  ATAC1_peak_8    112     .       4.85893 13.51786        11.24909        172
2L      106425  106749  ATAC1_peak_9    50      .       2.87129 6.93193 5.01890 101
2L      107790  108316  ATAC1_peak_10   202     .       5.49569 22.89202        20.23101        166
```



- “Merge” peaks (Bedtools)
- Count reads/peak (Bedtools)

- Associate peaks to genes
- Count table of counts for each gene

- DESeq2 or EdgeR for difference

Questions?