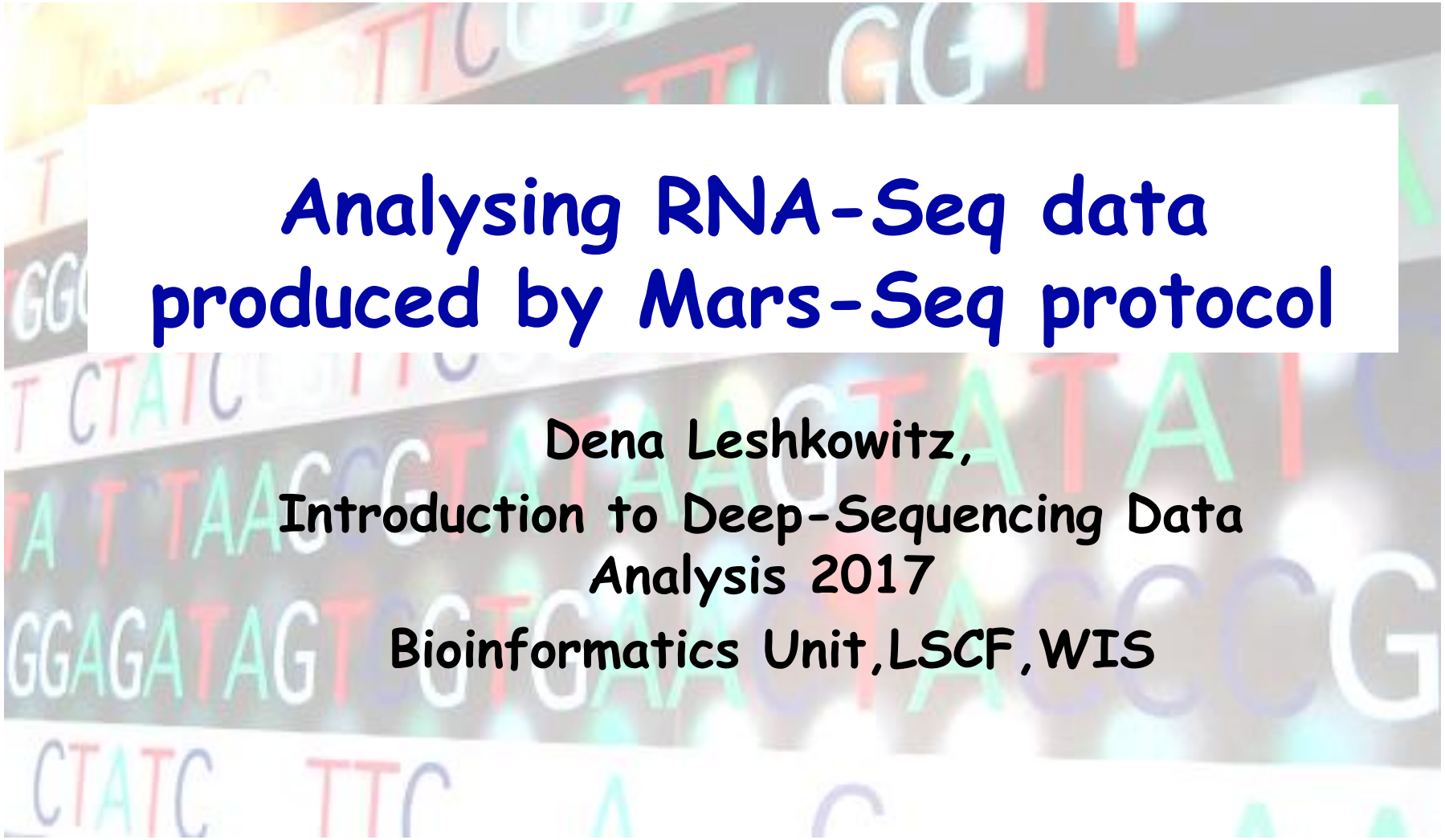
מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE



Analysing RNA-Seq data produced by Mars-Seq protocol

Dena Leshkowitz,

Introduction to Deep-Sequencing Data Analysis 2017

Bioinformatics Unit,LSCF,WIS

Directions for Using NextSeq at LSCF
http://susanc.weizmann.ac.il/ngs/howto.html

NextSeq Workflow

- The steps for the NextSeq Workflow
- Getting the NextSeq data
- Test your SampleSheet

Notice - It is your responsibility to delete the sequence data on our server. We will not store the data for beyond three months. Click **here** to check how much NextSeq data you currently have on storage.

Step 1: Get a userID on susanc

Before starting your work with the NextSeq, you will need to get a userID on susanc.

If you do not have a userID, call Irit Orr at 934-2470 and get one.

Step 2: Preparing your SampleSheet

In order to run analysis on the NextSeq data you will need to prepare a file with the details about your samples.
It's recommended to **test in advance** your SampleSheet or MARSseq file CLICK HERE
The file should be named and formatted according to the protocol you will be using.
In the SampleSheet file you need to enter who should receive the sequences. You may specify for selected samples to be delivered to separated users.

- ☑ If you prepared your samples following the Illumina protocol, then you need to provide a SampleSheet.csv file (sample)
- ☑ If you prepared your samples following the MARSeq protocol, then you need to provide a Mars-seq_users.xlsx file (sample)
- ☑ If you prepared your samples following the 10X Genomic protocol, then you need to provide a SampleSheet.csv file (sample)

10X Note: Leave the Lane column empty (*unless you know what your'e doing*)

Do Not Mix Protocols On A Run

Next-Seq initial pipeline at LSCF

- Enter userid info
- Bcl2fastq is done automatically once sequencing has been completed (on a server named stephan)
- If you provide one of the sample sheet formats required – multiplexing will be performed (no mismatch)
- When run is done you will receive an email including instructions on how to download the sequences. Storage is temporarily

	A	B	C	D	E	F	G	H	I	J	K	L
1	#	Sample ID	rd1 barcode	rd1 barcode name	rd2 barcode name	rd2 barcode	i7 index barc		i5 index barcode	i5 index name	userid	
2	1	POS_CP_WT-A			v3_gr1A3	TGTCACG					idoamit	
3	2	POS_CP_WT-B			v3_gr1B3	TTCCTGA					sorek	
4	3	POS_CP_AD-A			v3_gr1C3	GGATCTA					hadask	
5	117											

Information required for demultiplexing

Dimultiplexing

Multiplexing allows to pool samples together and sequence them simultaneously

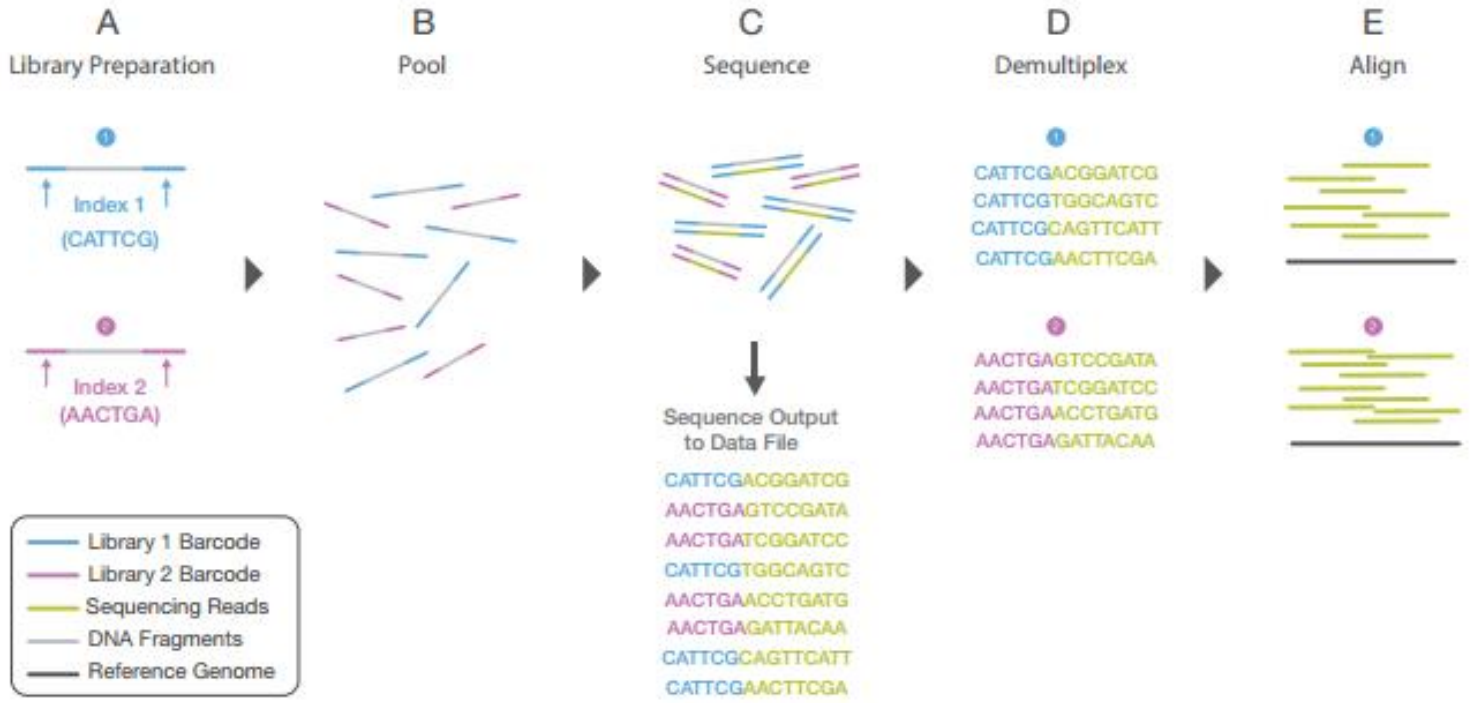


Figure 5: Library Multiplexing Overview.

a. Two distinct libraries are attached to unique index sequences. Index sequences are attached during library preparation.

b. Libraries are pooled together and loaded into the same flow cell lane.

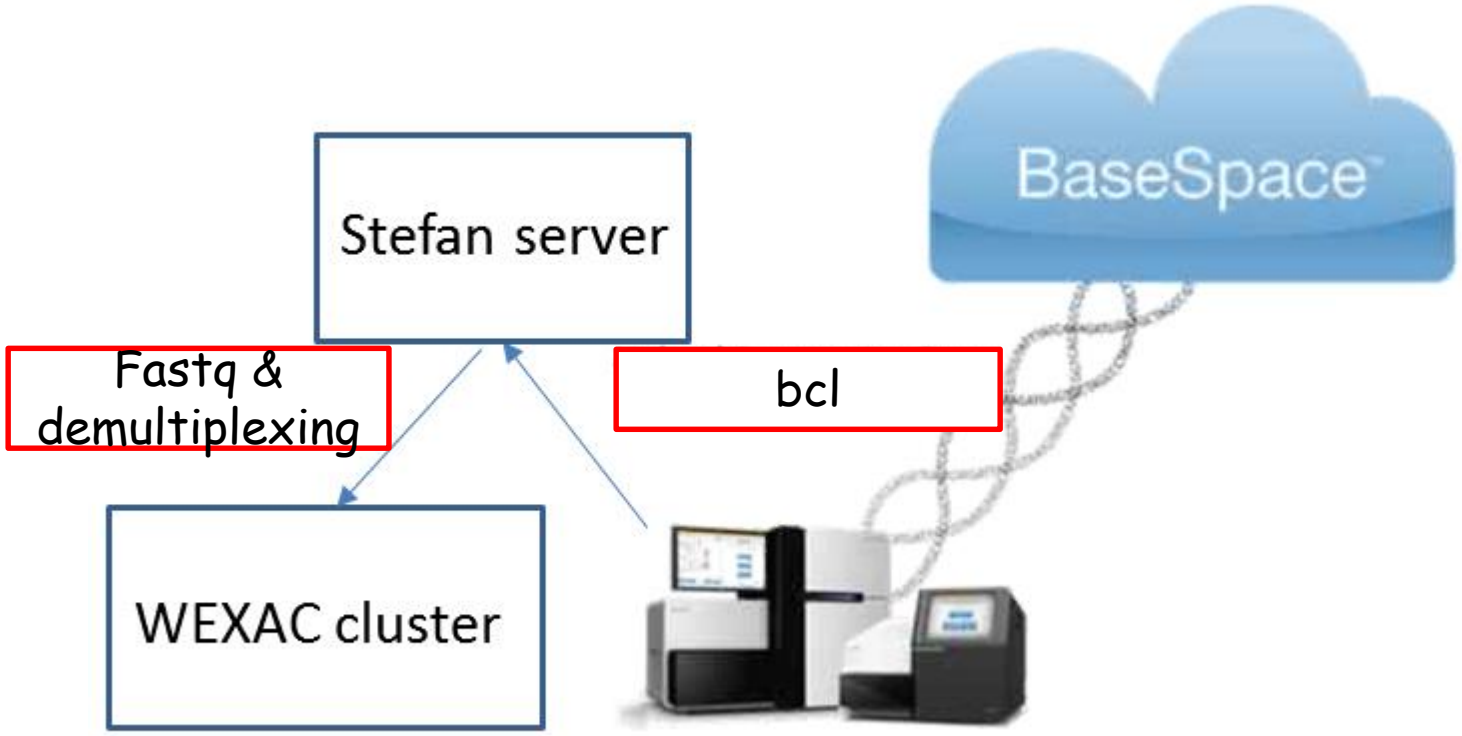
c. Libraries are sequenced together during a single instrument run. All sequences are exported to a single output file.

d. A demultiplexing algorithm sorts the reads into different files according to their indexes.

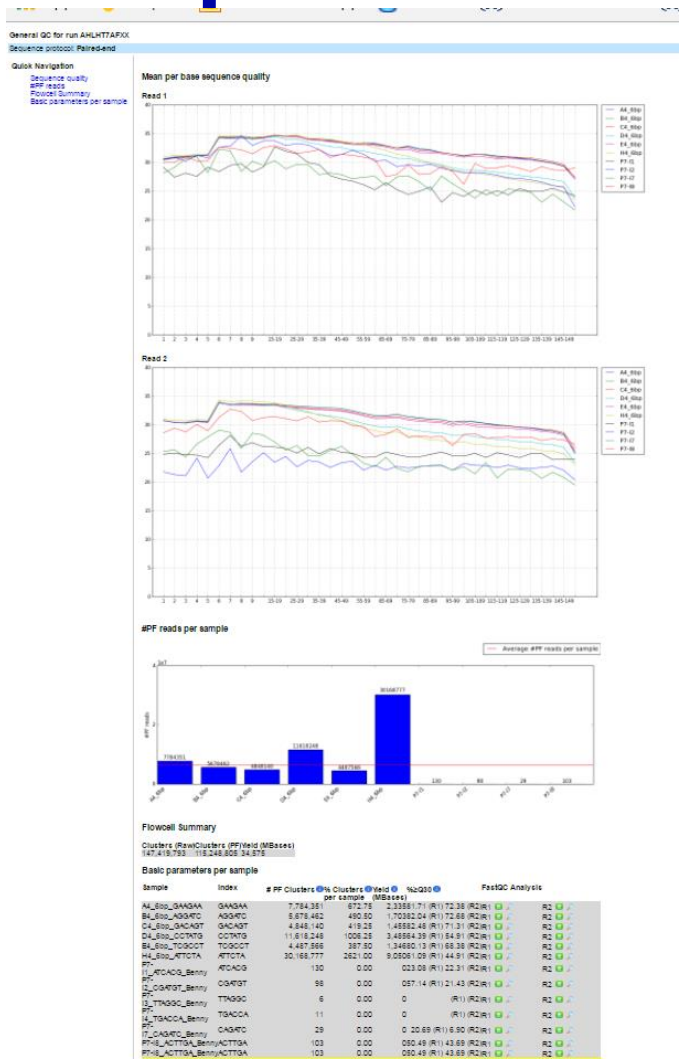
e. Each set of reads is aligned to the appropriate reference sequence.

http://www.illumina.com/content/dam/illumina-marketing/documents/products/illumina_sequencing_introduction.pdf

Fastq files & demultiplexing run on Stefan



QC reports for Next-seq runs at LSCF



Flowcell Summary

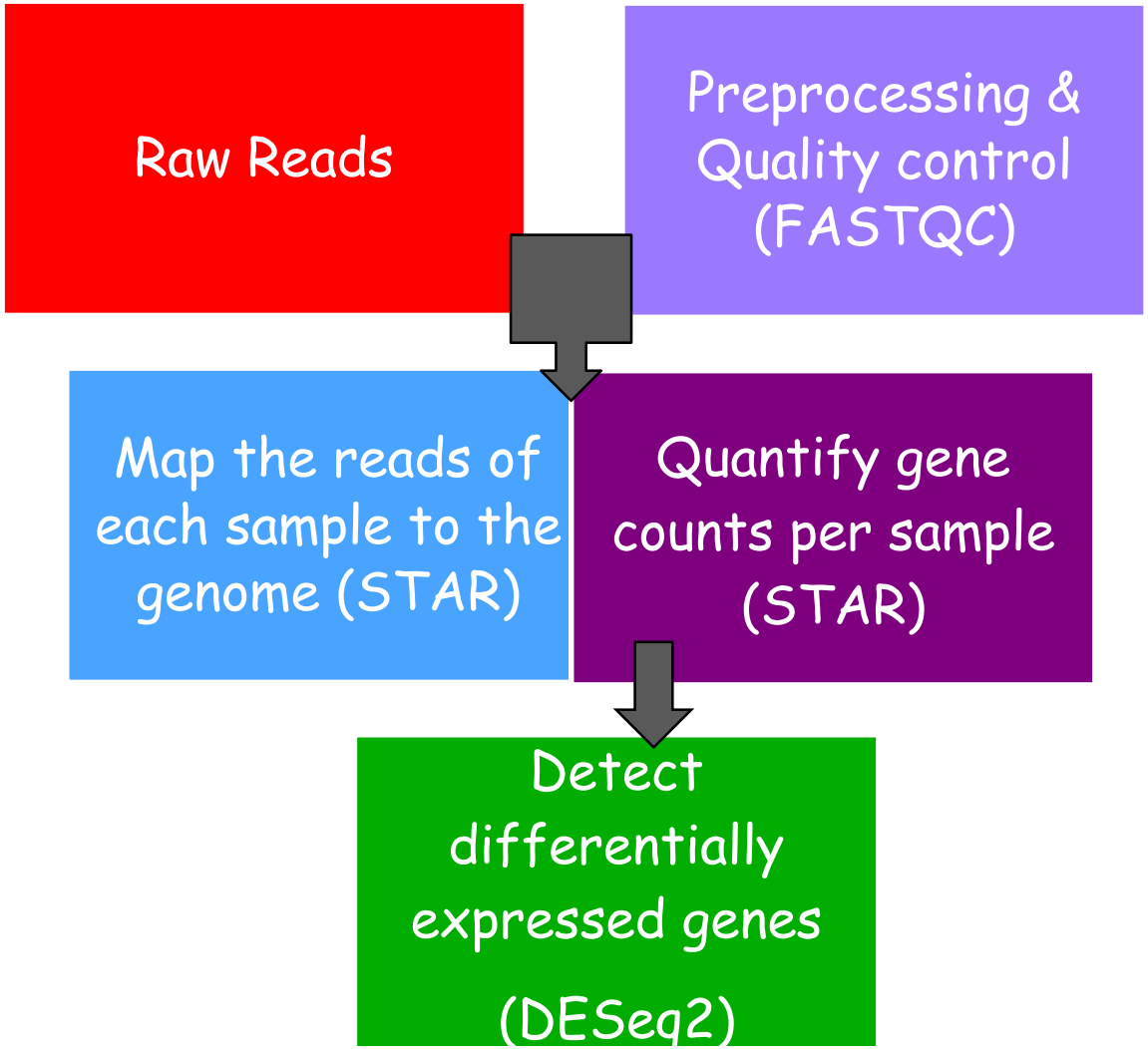
Clusters (Raw)	Clusters (PF)	Yield (MBases)
147,419,793	115,248,805	34,575

Basic parameters per sample

Sample	Index	# PF Clusters	% Clusters per sample	Yield (MBases)	%≥Q30	FastQC Analysis	
A4_6bp_GAAGAA	GAAGAA	7,784,351	672.75	2,335	81.71 (R1) 72.38 (R2)	R1	R2
B4_6bp_AGGATC	AGGATC	5,678,462	490.50	1,703	82.04 (R1) 72.68 (R2)	R1	R2
C4_6bp_GACAGT	GACAGT	4,848,140	419.25	1,455	82.48 (R1) 71.31 (R2)	R1	R2
D4_6bp_CCTATG	CCTATG	11,618,248	1006.25	3,485	64.39 (R1) 54.91 (R2)	R1	R2
E4_6bp_TCGCCT	TCGCCT	4,487,566	387.50	1,346	80.13 (R1) 68.38 (R2)	R1	R2
H4_6bp_ATTCTA	ATTCTA	30,168,777	2621.00	9,050	61.09 (R1) 44.91 (R2)	R1	R2
P7-I1_ATCACG_Benny	ATCACG	130	0.00	0	23.08 (R1) 22.31 (R2)	R1	R2
P7-I2_CGATGT_Benny	CGATGT	98	0.00	0	57.14 (R1) 21.43 (R2)	R1	R2
P7-I3_TTAGGC_Benny	TTAGGC	6	0.00	0	(R1) (R2)	R1	R2
P7-I4_TGACCA_Benny	TGACCA	11	0.00	0	(R1) (R2)	R1	R2
P7-I7_CAGATC_Benny	CAGATC	29	0.00	0	20.69 (R1) 6.90 (R2)	R1	R2
P7-I8_ACTTGA_Benny	ACTTGA	103	0.00	0	50.49 (R1) 43.69 (R2)	R1	R2
P7-I8_ACTTGA_Benny	ACTTGA	103	0.00	0	50.49 (R1) 43.69 (R2)	R1	R2
Undetermined Indices	Undetermined	50,662,884	4402.25	15,200			

http://stefan.weizmann.ac.il

RNA-Seq Workflow



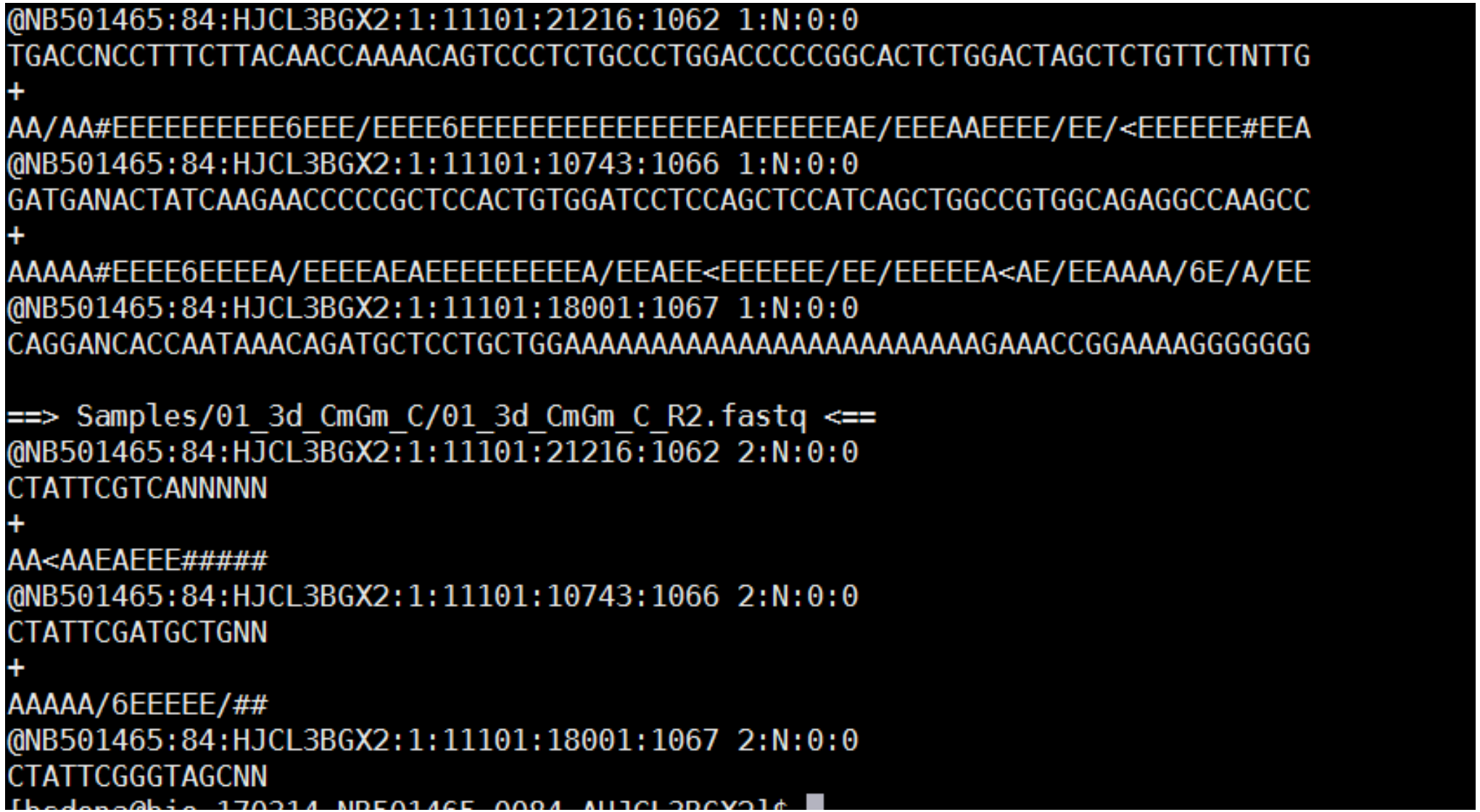
Getting from Fastq files to gene counts:

1. Editing R1 Sequence header to contain UMI

Step 1 -

R1

R2


@NB501465:84:HJCL3BGX2:1:11101:21216:1062 1:N:0:0
TGACCNCCTTTCTTACAACCAAAACAGTCCCTCTGCCCTGGACCCCCGGCACTCTGGACTAGCTCTGTTCTNTTG
+
AA/AA#EEEEEEEEE6EEE/EEEE6EEEEEEEEEEEEEEEAEEEEEEAE/EEEAAEEEE/EE/<EEEEEE#EEA
@NB501465:84:HJCL3BGX2:1:11101:10743:1066 1:N:0:0
GATGANACTATCAAGAACCCCCGCTCCACTGTGGATCCTCCAGCTCCATCAGCTGGCCGTGGCAGAGGCCAAGCC
+
AAAAA#EEEE6EEEEA/EEEEAEAEEEEEEEEEA/EEAEE<EEEEEE/EE/EEEEEA<AE/EEAAAA/6E/A/EE
@NB501465:84:HJCL3BGX2:1:11101:18001:1067 1:N:0:0
CAGGANCACCAATAAACAGATGCTCCTGCTGGAAAAAAAAAAAAAAAAAAAAAAAGAAACCGGAAAAGGGGGGG
==> Samples/01_3d_CmGm_C/01_3d_CmGm_C_R2.fastq <==
@NB501465:84:HJCL3BGX2:1:11101:21216:1062 2:N:0:0
CTATTCGTCANNNNN
+
AA<AAEAEEE#####
@NB501465:84:HJCL3BGX2:1:11101:10743:1066 2:N:0:0
CTATTCGATGCTGNN
+
AAAAA/6EEEEE/##
@NB501465:84:HJCL3BGX2:1:11101:18001:1067 2:N:0:0
CTATTCGGGTAGCNN

Step 1 - UMI moved from R2 to header

```
[redacted] head analysis/1_combined_fastq/17/17.R1.combined.fastq
@NB501465:88:HJHMFBGX2:1:11101:24963:1074_RX:Z:AACATAGT_QX:Z:14,14,36,36,36,14,14,36 1:N:0:0
CGTGCCACACACCCTGGAGCATAGCAGAGCTGTGCTACTGGAGATGTATAATCCGTTTTGATATGCAAAAGAATA
+
/AAAA//EE6/AE/E/E/EE/6//EE//EEAE///EE/E66/EEA6//AE//EE/EEE////A//6A/EE/E///
@NB501465:88:HJHMFBGX2:1:11101:12669:1074_RX:Z:GTTAACGT_QX:Z:32,36,36,36,36,36,27,36 1:N:0:0
GTTTGTTTTATTTTGTTCTAAAATTAAAAGTATGCAAAAAAAAAAAAAAAAAAAAAACGTTAACCGGGGGCCGG
```

Getting from Fastq files to gene counts:

1. Editing R1 Sequence header to contain UMI
2. Trim first 3 bases, adapter and poly A/T bases using cutadapt
3. Map the edited fastq using STAR
4. Modify BAM file - move UMI from read name to a FLAG

Move UMI

```
[bsdena@bio 4_mapping]$ samtools view 17Aligned.sortedByCoord.out.bam | head
NB501465:88:HJHMFBGX2:3:21411:23237:15991_RX:Z:ACGTTTGG_QX:Z:14,32,36,14,14,36,36,32    256     chr1    3083902 0       34M     *       0
0       ACCCAAAAAAAAAAAAAAAAAAAAAAAACACA      AAAAAEEEEEAEEEEEEEEEEEEAEEEE//A//A      NH:i:5  HI:i:3  AS:i:31 nM:i:1
```

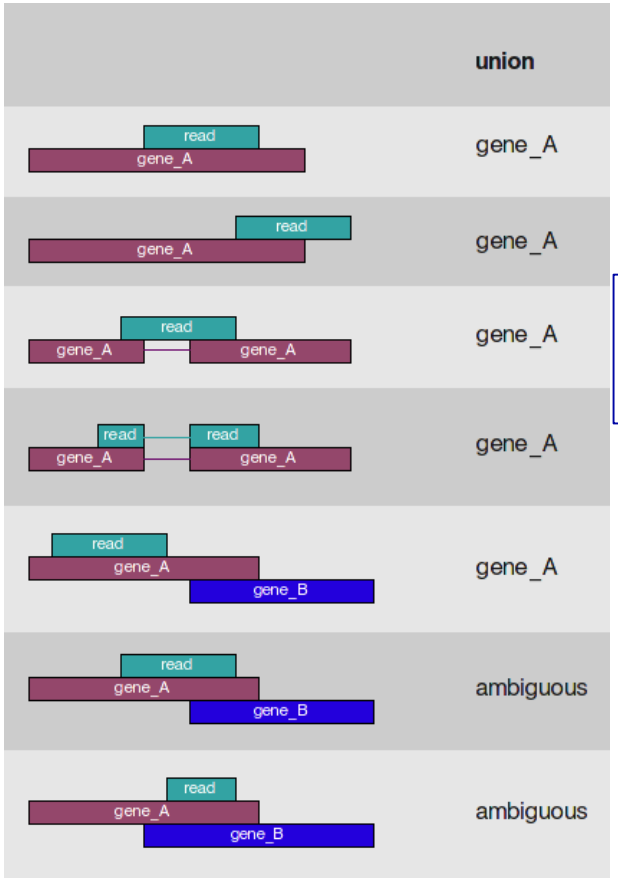
```
NB501465:88:HJHMFBGX2:3:21411:23237:15991       256     chr1    3083902 0       34M     *       0       0       ACCCAAAAAAAAAAAAAAAAAA
AAAAAAAACACA    AAAAAEEEEEAEEEEEEEEEEEEAEEEE//A//A      NH:i:5  HI:i:3  AS:i:31 nM:i:1  RX:Z:ACGTTTGG   QX:Z:/AE//EEA
```

Getting from Fastq files to gene counts:

1. Editing R1 Sequence header to contain UMI
2. Trim first 3 bases, adapter and poly A/T bases using cutadapt
3. Map the edited fastq using STAR
4. Modify BAM file - move UMI from read name to a FLAG
5. Sort and index the resulting bam using 'samtools sort' and 'samtools index'
6. Run HTSeq on modified GTF file to produce raw counts

HTSeq

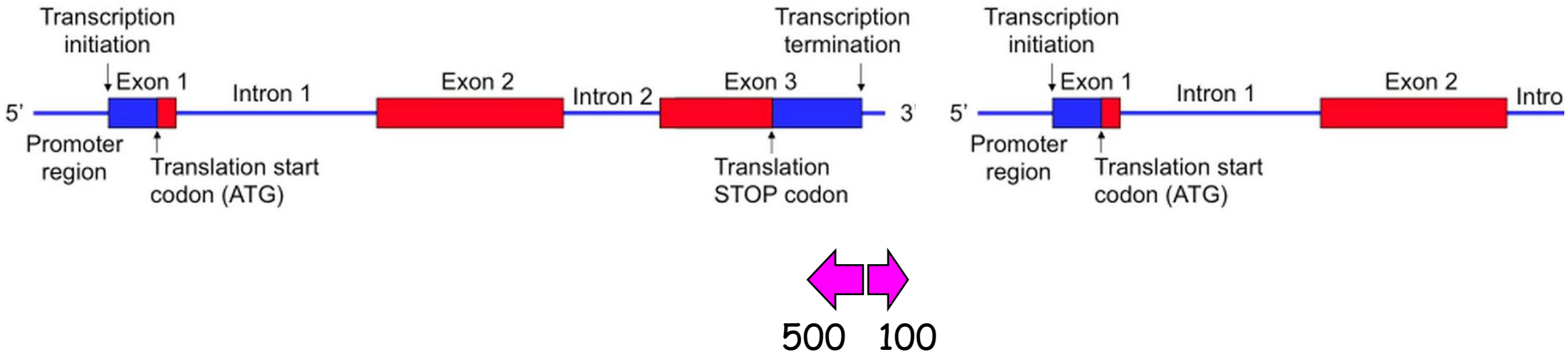
A gene is quantified by counting the number of fragments/reads which align to all its exons – in Mars-seq only to the 3 end of the genes



Discard a read if it is non-uniquely mapped to a gene

http://www-huber.embl.de/users/anders/HTSeq/doc/count.html

Create 3UTR GTF file



Getting from Fastq files to gene counts:

1. Editing R1 Sequence header to contain UMI
2. Trim first 3 bases, adapter and poly A/T bases using cutadapt
3. Map the edited fastq using STAR
4. Modify BAM file - Move UMI from read name to a FLAG
5. Sort and index the resulting bam using 'samtools sort' and 'samtools index'
6. Run HTSeq on modified GTF file to produce raw counts
7. Add feature tag to bams (gene)
8. Mark duplicate Reads

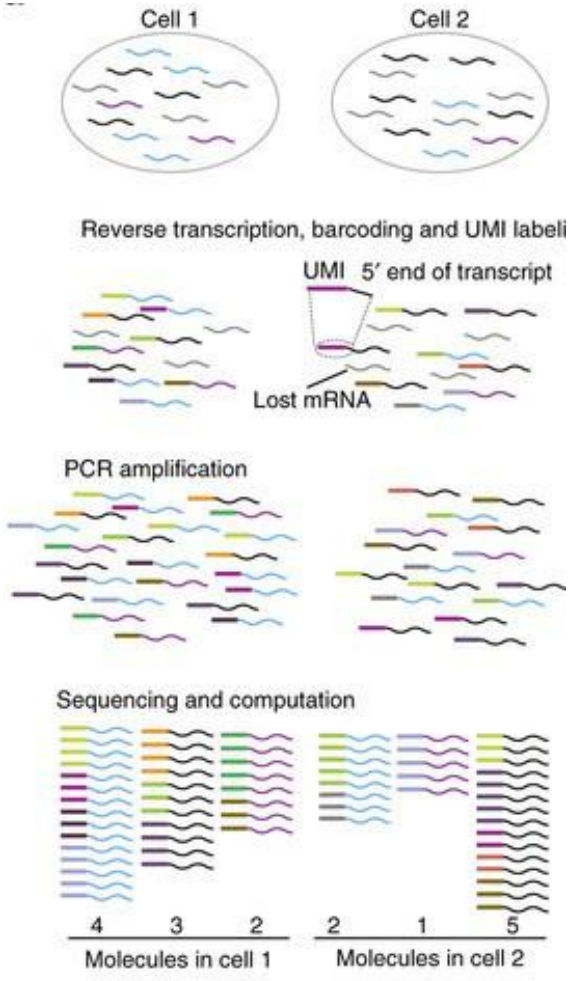
Add Feature Tag and Mark Duplicates

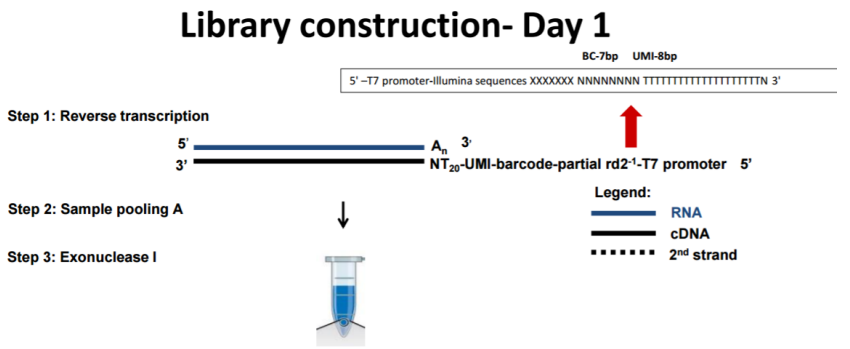
```
K00279:102:HJK2GBBXX:1:1104:3995:7222   0       chr1    164437130       255     101M    *       0       0
GTCCGTTGTGTTAACCTGACTGACACAAGATCAACAAGAGCACTGTACTCCTGGCAATTATTACATATGCTAGAACATGGATTTGCACTGTAGACAACACT
-AFFFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFFJJFJJFJJFFJ    NH:i:1  HI:i:1  AS:i:99 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8


K00279:102:HJK2GBBXX:1:1102:23399:31734 1024    chr1    164437180       255     100M    *       0       0
CTGGCAATTATTACATATGCTAGAACATGGATTTGCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATA
<A-FFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ    NH:i:1  HI:i:1  AS:i:98 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1102:30919:42724 1024    chr1    164437198       255     101M    *       0       0
GCTAGAACATGGATTTGCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATAAAGATTTTTATTTTCAAAC
AAAFFJJJJJJJJJJJJJJJJJJJJFJJJJJFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFJFJJAJJJFJFJJJJJF<JFJJJJFJJJJJJJJ    NH:i:1  HI:i:1  AS:i:99 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1103:31994:17438 1024    chr1    164437199       255     100M    *       0       0
CTAGAACATGGATTTGCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATAAAGATTTTTATTTTCAAAC
AAFAFJJJJJJJJJJFJJJJJJJJJJJJJJJJJJJJJJJJJJFFJJJJJJJJJJJFJJJF<JJFJJJAF7<-AFFF77A<FJJF-7FJJJJJJJJJ<FFJF    NH:i:1  HI:i:1  AS:i:98 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1105:22343:7468  1024    chr1    164437210       255     98M     *       0       0
ATTTGCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATACAGATTTTTATTTTCAAACAAGTGACTT
AA<A-<A7A-FJ<FFJA-FJFAFAJJJ-FA-7---F--A--77F-77<FFFJ<7--JF7FF<-<FJA7<--<A<-<7AF-FA-7FJJFAAF7-7F7FA    NH:i:1  HI:i:1  AS:i:94 nM:i:1  RX:Z:ATTTTTTT
QX:Z:JJFJFJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1102:4564:36200  1024    chr1    164437211       255     101M    *       0       0
TTTGCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATAAAGATTTTTATTTTCAAACAAGTGACTTTGGT
AAFFFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ    NH:i:1  HI:i:1  AS:i:99 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1103:13869:49089 1024    chr1    164437214       255     101M    *       0       0
GCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATAAAGATTTTTATTTTCAAACAAGTGACTTTGGTTTT
AAFFFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFJFJFJFF    NH:i:1  HI:i:1  AS:i:99 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
K00279:102:HJK2GBBXX:1:1104:13362:2563  1024    chr1    164437214       255     101M    *       0       0
GCACTGTAGACAACACTTAACACCAGTCTATGGGGTACTGCATTGCTTTTTATAAAGTTCAAAATAAAGATTTTTATTTTCAAACAAGTGACTTTGGTTTT
AAFFFJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJFJJJJJJJJJAJJJJJFJJJJJJJJJJJFFAJFAFFJFAFF    NH:i:1  HI:i:1  AS:i:99 nM:i:0  RX:Z:ATTTTTTT
QX:Z:JJJJJJJJ   XF:Z:Nme7       rp:Z:K00279:102:HJK2GBBXX:1:1104:3995:7222      nd:i:8
```

Using UMI to identify correct number of reads-fragments

In the analysis of Mars-Seq data reads are considered duplicated if they map to the same gene and have the same UMI (errors are not considered as duplicates)





This figure is adapted from Islam et al (2014)

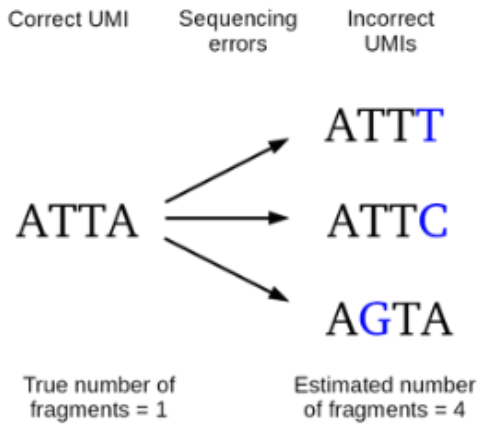
Getting from Fastq files to gene counts:

1. Editing R1 Sequence header to contain UMI
2. Trim first 3 bases, adapter and poly A/T bases using cutadapt
3. Map the edited fastq using STAR
4. Modify BAM file - Move UMI from read name to a FLAG
5. Sort and index the resulting bam using 'samtools sort' and 'samtools index'
6. Run HTSeq on modified GTF file to produce raw counts
7. Add feature tag to bams (gene)
8. Mark duplicate Reads
9. Run HTSeq to produce deDuplicated counts
10. Correct deDuplicated counts

Barcode clashing correction

UMI barcodes are assigned to fragments (semi-) randomly, so it might happen that two independent fragments get assigned the same UMI barcode. When the number of fragments mapping to a gene are low, there are few such clashes, and the effect on the counts is negligible. However, when the counts are of the same order of magnitude as the number of barcode options (4^bcLength) clashing becomes significant, and a correction should be incorporated.

The problem with the use of UMI



Sequencing errors inflate the apparent numbers of unique fragments sequenced

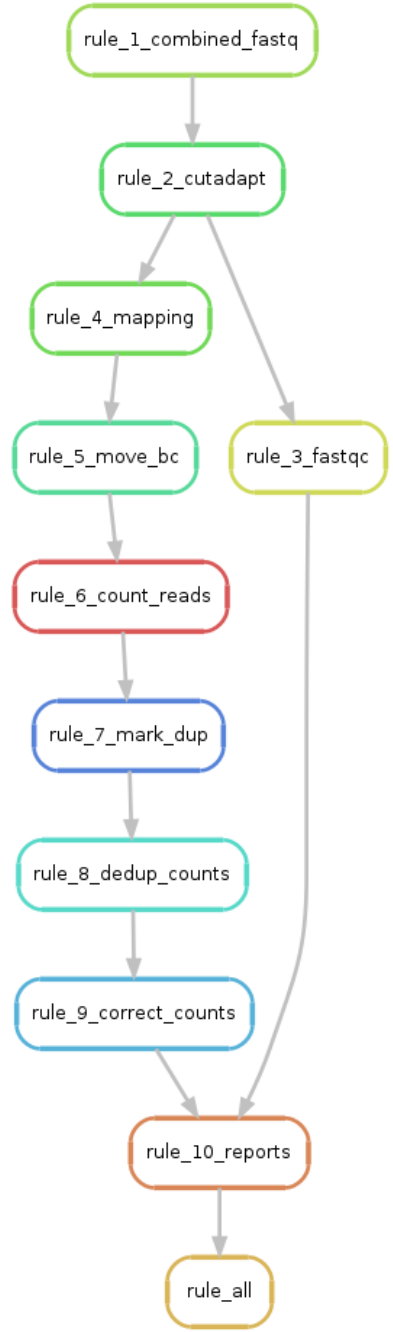
https://cgatoxford.wordpress.com/2015/08/14/unique-molecular-identifiers-the-problem-the-solution-and-the-proof/

Pipeline

Public Tools

- Python
- STAR
- HTSeq
- Picard
- Samtools
- Snakemake
- R (DESeq2)

Initial Pipeline Developers: Barak Markus, Jonathan Barlev & Gil Hornung , Bioinformatics Unit, INCPM, WIS


rule_1_combined_fastq
rule_2_cutadapt
rule_4_mapping
rule_5_move_bc
rule_3_fastqc
rule_6_count_reads
rule_7_mark_dup
rule_8_dedup_counts
rule_9_correct_counts
rule_10_reports
rule_all

Link to GUI pipeline and manual

- Pipeline interface

  http://ngsbio.wexac.weizmann.ac.il
- Manual

  https://bbcunit.atlassian.net/wiki/pages/viewpage.action?pageId=56524801

- Pipeline, GUI and more by Rafael Kohen, Bioinformatics Unit, LSCF

  (Dena Leshkowitz and Ester Feldmesser design)

Thanks

Questions??