Overview of Assembly Techniques for Next Generation Sequencing Data

Noushin Ghaffari, PhD
Bioinformatics Scientist, Genomics and Bioinformatics, Texas A&M AgriLife Research
Research Scientist, Texas A&M High Performance Research Computing

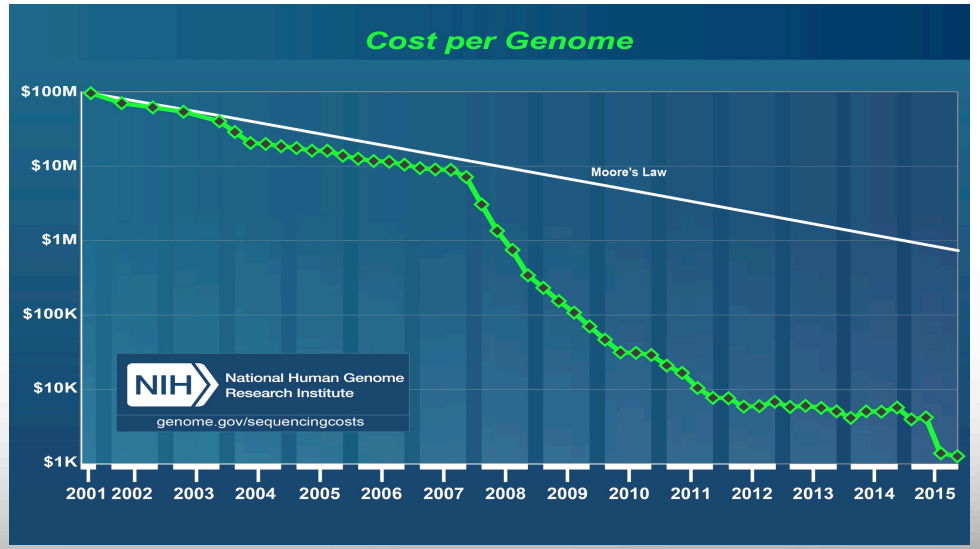
DIVISION OF RESEARCH
TEXAS A&M UNIVERSITY

Outline

- Background
  - Sequencing
- Application of Next Generation Sequencing in Research
  - Transcriptome assembly
  - Evaluations
  - Hands-on experiments

Why sequencing?

- Determining the sequence of nucleotides within a DNA (or RNA) fragment
- Gene recognition
- Ultimately completing the genome of non-model organisms, e.g. *Pacific whiteleg shrimp*
- Human genome project
  - $3.8 Billion
  - 13 years to complete
  - 1990-2003
  - 8-9x coverage
  - 27 GBases



Sanger

Classic Sequencing

Third Generation Sequencing Platforms

PacBio

Next Generation Sequencing Platforms

Roche GS-FLX

Life Technologies SOLiD

Illumina HiSeq

Life Technologies Ion Torrent

MinION

Primary NGS Applications

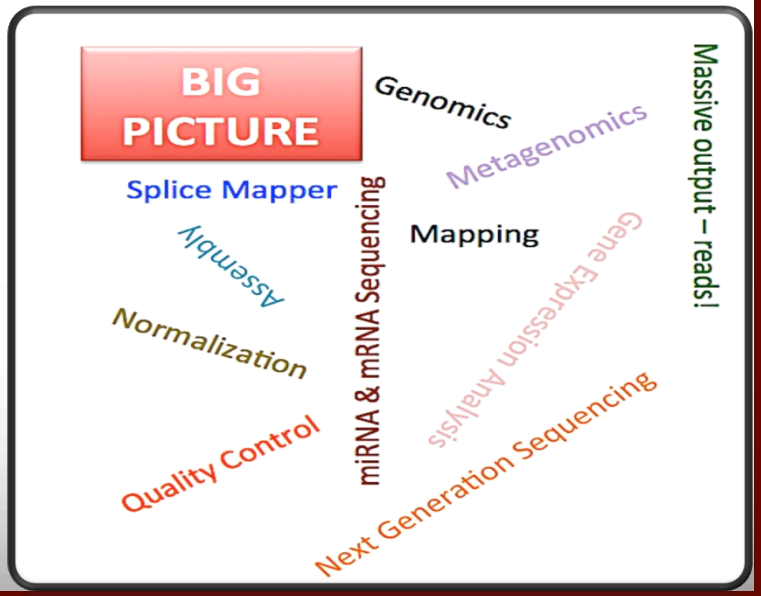
1. Alignment
2. **Assembly**(no reference/with a reference)
   - **Genome**
   - **Transcriptome**
3. RNA-Seq
4. Metagenomics
5. ChIP-Seq
6. RADSeq

Today

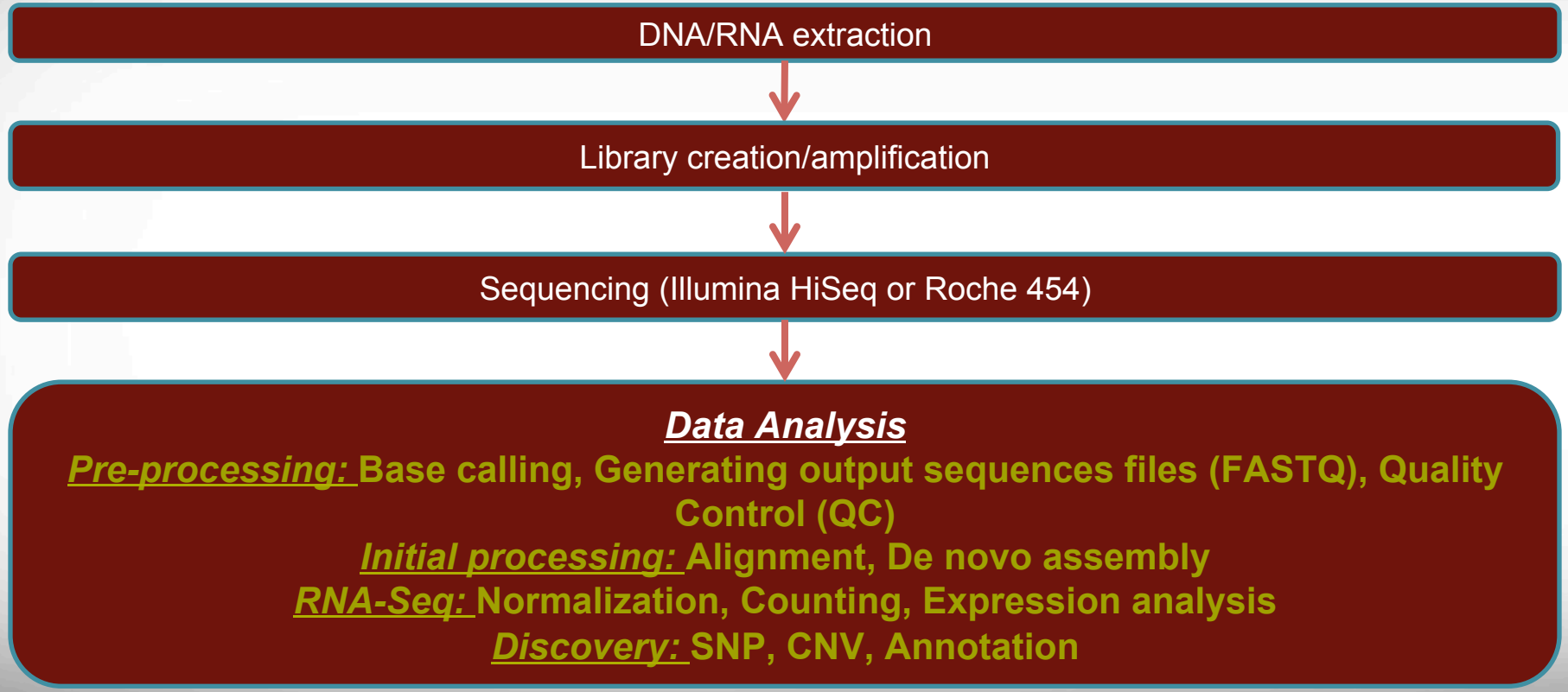
Last Week

Next Week

Next Month



NGS Sequencing Workflow

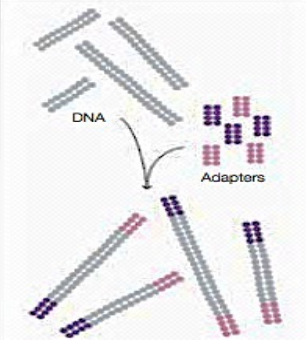


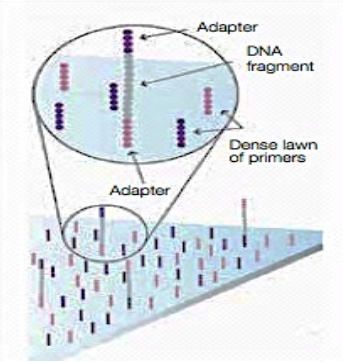
Illumina next-generation sequencing - 1

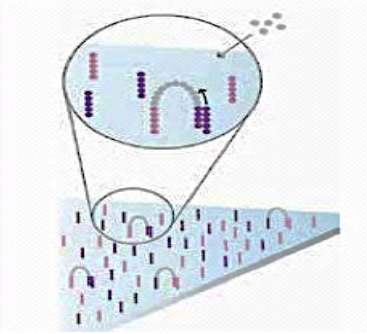
Sequencing by Synthesis (SBS) Technology

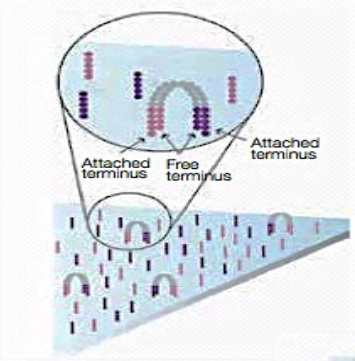
- Randomly shearing DNA
- Attaching DNA fragments to the flowcell surface
- Cluster generation
  - Duplicating single stranded fragments by “Bridge Amplification”
  - Denaturing the double-stranded DNA
- Adding four labelled *reversible terminators*, primers, and DNA polymerase
- Determining the attached nucleotide, based on the emitted fluorescence

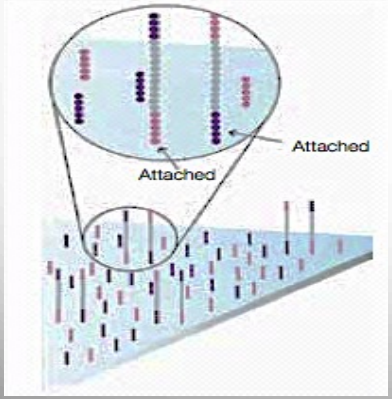
Illumina next-generation sequencing - 2

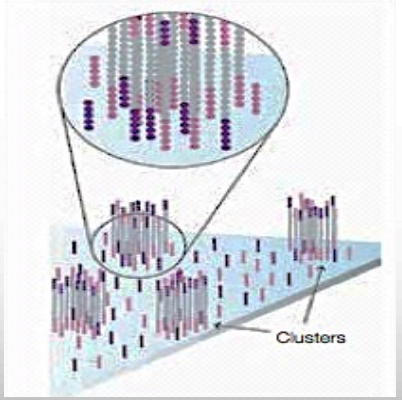


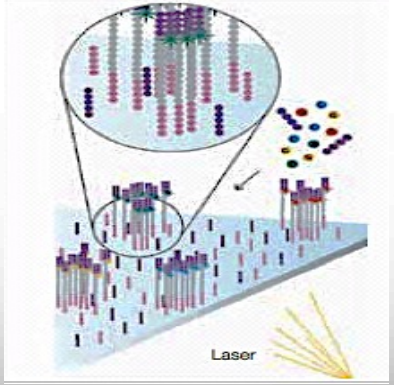


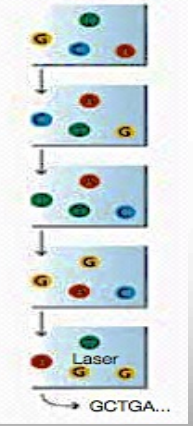












Sequence and Quality Scores

Quality scores measure the probability that a base is called incorrectly.

Quality Score

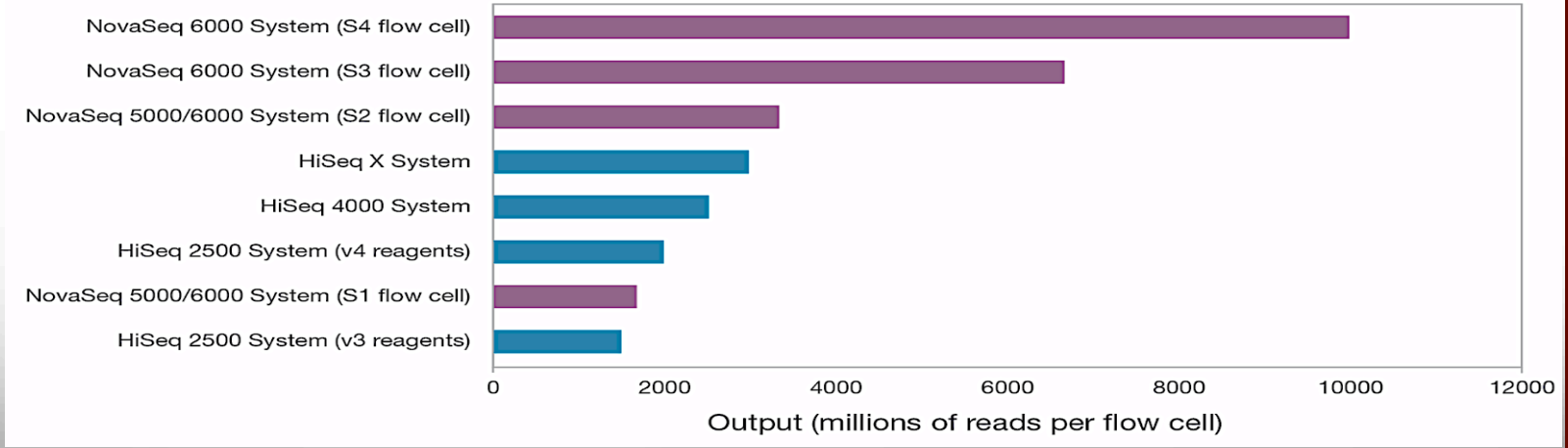
- Illumina Quality Score
  - Phred-like algorithm: similar to scoring for Sanger sequencing
  - Quality score of a given base, Q, is defined as:
  - e: estimated probability of the base call being wrong

$$Q = -10\log_{10}(e)$$

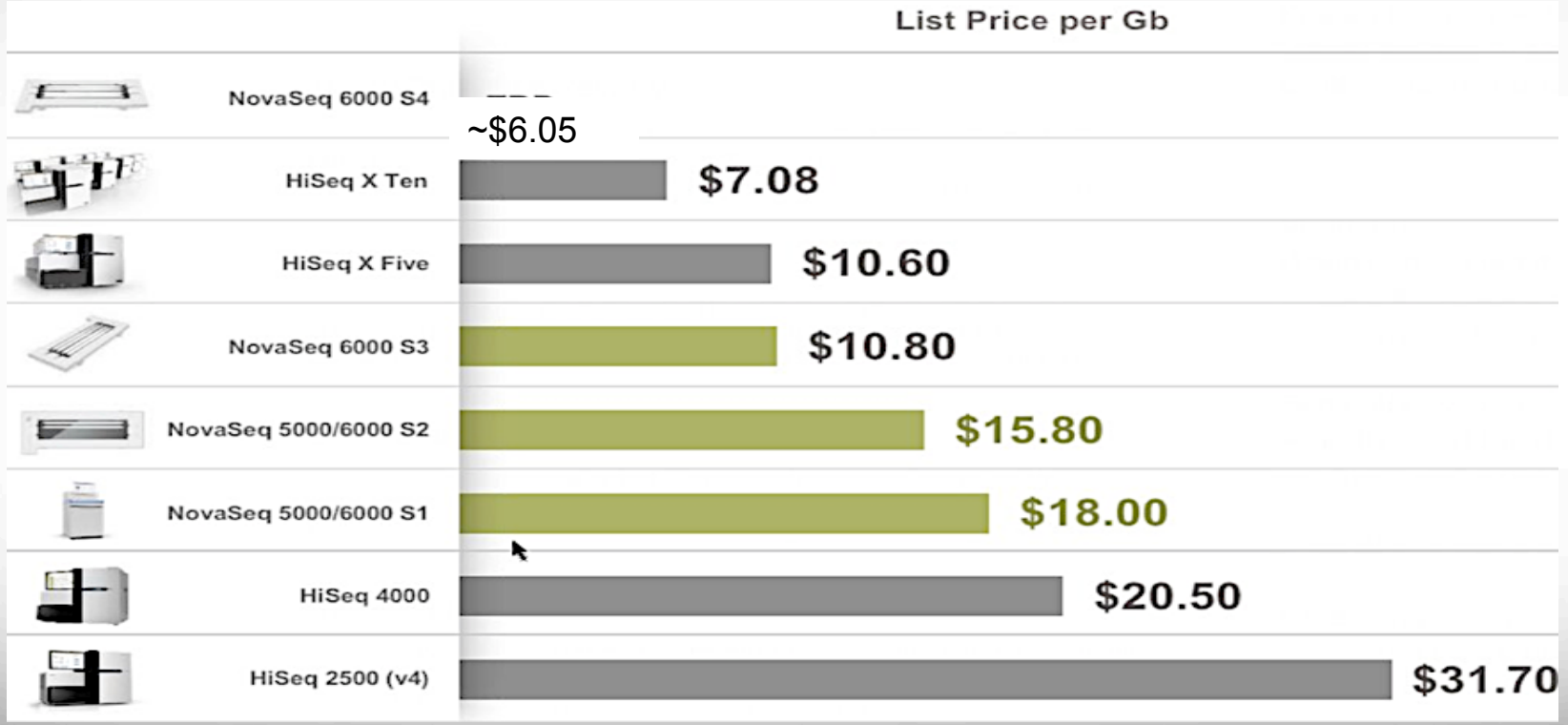
Quality Score	Probability of Incorrect Base Call	Inferred Base Call Accuracy
10 (Q10)	1 in 10	90%
20 (Q20)	1 in 100	99%
30 (Q30)	1 in 1000	99.90%

NovaSeq Sepecifications

- Each NovaSeq S4 will generate the equivalent of 32 lanes on the HiSeq 4000
- 96 samples per flow cell -> 384 samples, 200TB of data generated per run, 10,560 reads
- 150PE price drop RNASeq 207 -> $83 / sample
- One NovaSeq = 8 HS4000 = 11 HS2500
- 60% cost reduction



More affordable to Sequence!



Long reads

Pacific Biosciences (PacBio)

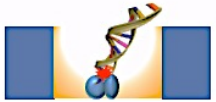
- Single Molecule Real Time Sequencing (SMRT) Methodology
- Fluorescent dyes
- Zero Mode Waveguide
- https://www.youtube.com/watch?v=NHCJ8PtYCFc

Oxford Nanopore Technologies (MinION)

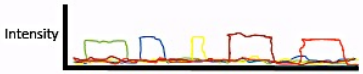
- Nanopore: a small hole (nanometer)
  - used to identify DNA sequence, passing through nanopore
- Single DNA molecule is sequenced
- https://www.youtube.com/watch?v=GUb1TZvMWsw

PacBio SMRT seq

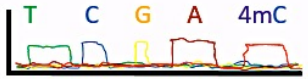
DNA passes thru polymerase in an illuminated volume



Raw output is fluorescent signal of the nucleotide incorporation, specific to each nucleotide

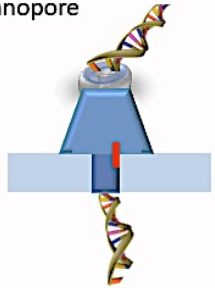


A,C,T,G have known pulse durations, which are used to infer methylated nucleotides

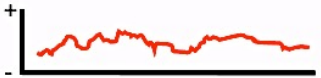


Oxford Nanopore

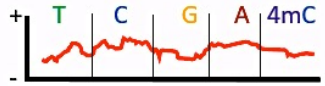
DNA passes thru nanopore



Raw output is electrical signal caused by nucleotide blocking ion flow in nanopore



Each nucleotide has a specific electric "signature"

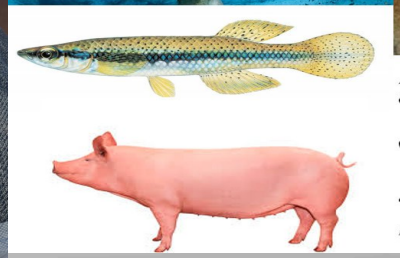


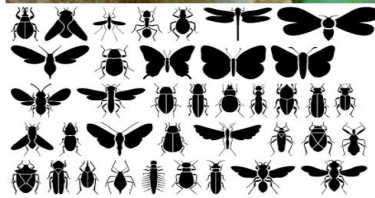
Nivretta Thatra, PacBio SMRT technology and Oxford Nanopore can use unaltered DNA to detect methylation

PacBio Sequel

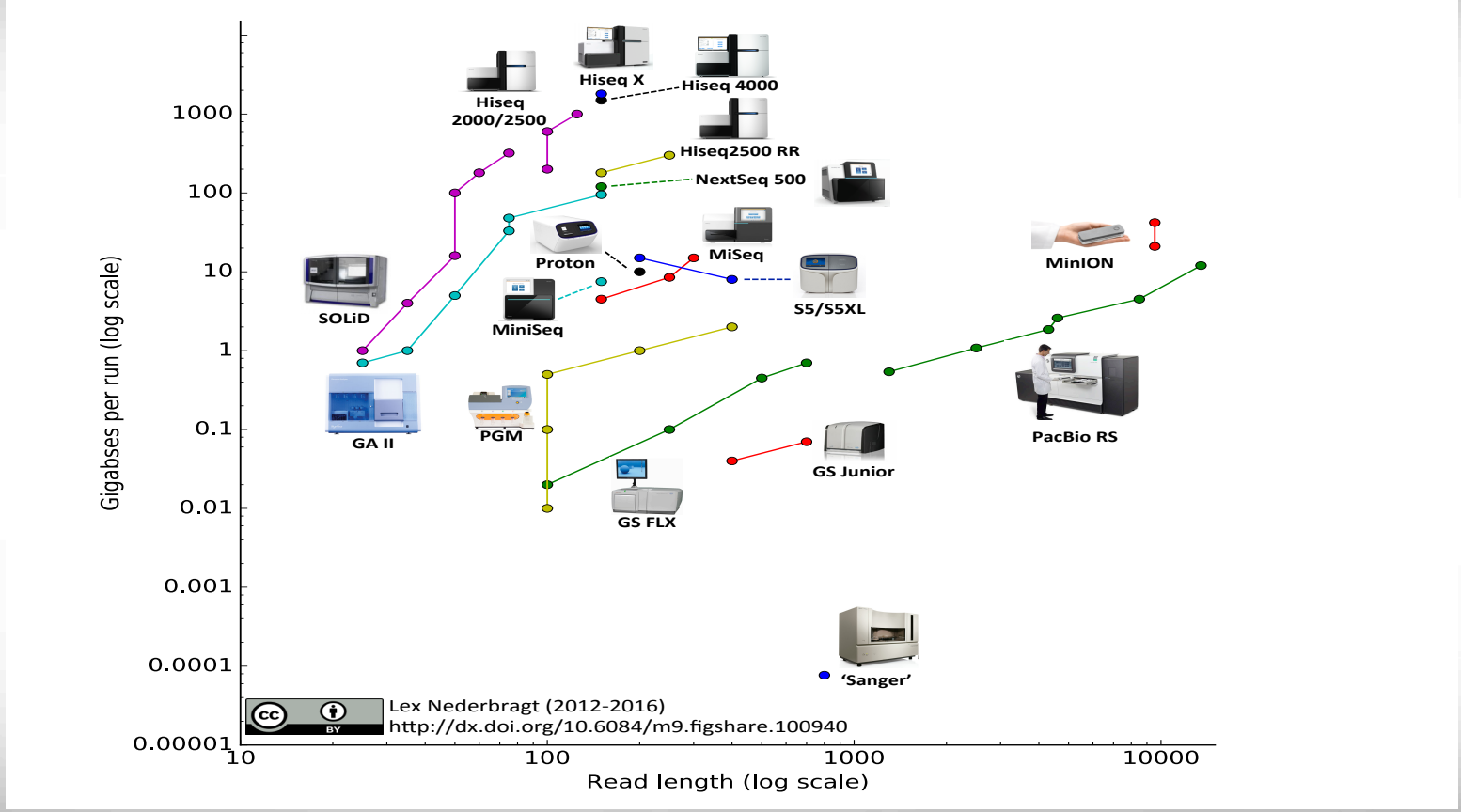
- 5 GB per SMRT Cell
- 1M ZMW/SMRT Cell
- Up 16 SMRT/week
- 10 hour run time/SMRT
- Avg. read 10-15kb
- ~10x jump over RSII

- Real time genomic sequencing is possible with portable **MinION - Oxford Nanopore Technologies (ONT)**
- Texas A&M AgriLife PoreCampUSA 2017 consisted of sample preparation, sequencing, basecalling, data pre-processing, quality control and bioinformatics analysis.
- On-site sequencing at **Jester King Brewery** in Austin!





NGS Read Specifications



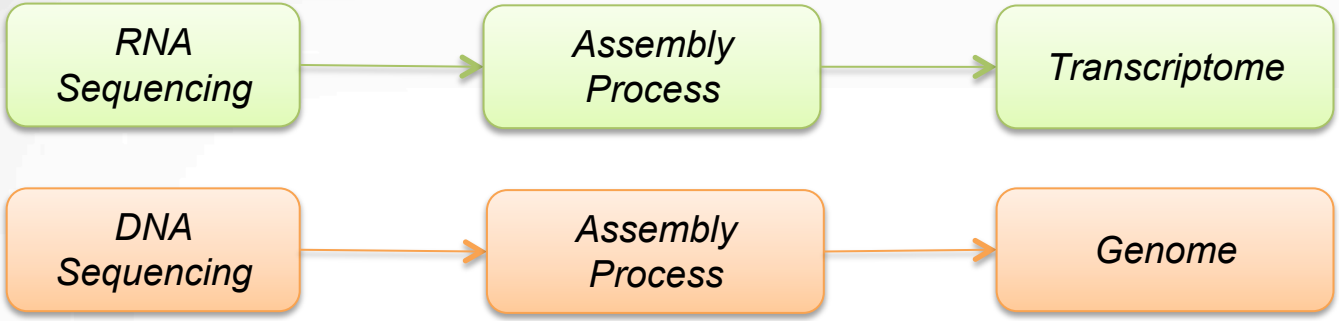
Lex Nederbragt blog: https://flxlexblog.wordpress.com/2016/07/08/developments-in-high-throughput-sequencing-july-2016-edition/

Why assembly?

Generating the consensus of transcriptome or genome of non-model species

Reconstructing the genome and transcriptome of non-model species are essential steps in expanding our understanding of the organism and developing therapeutic approaches to fight disease

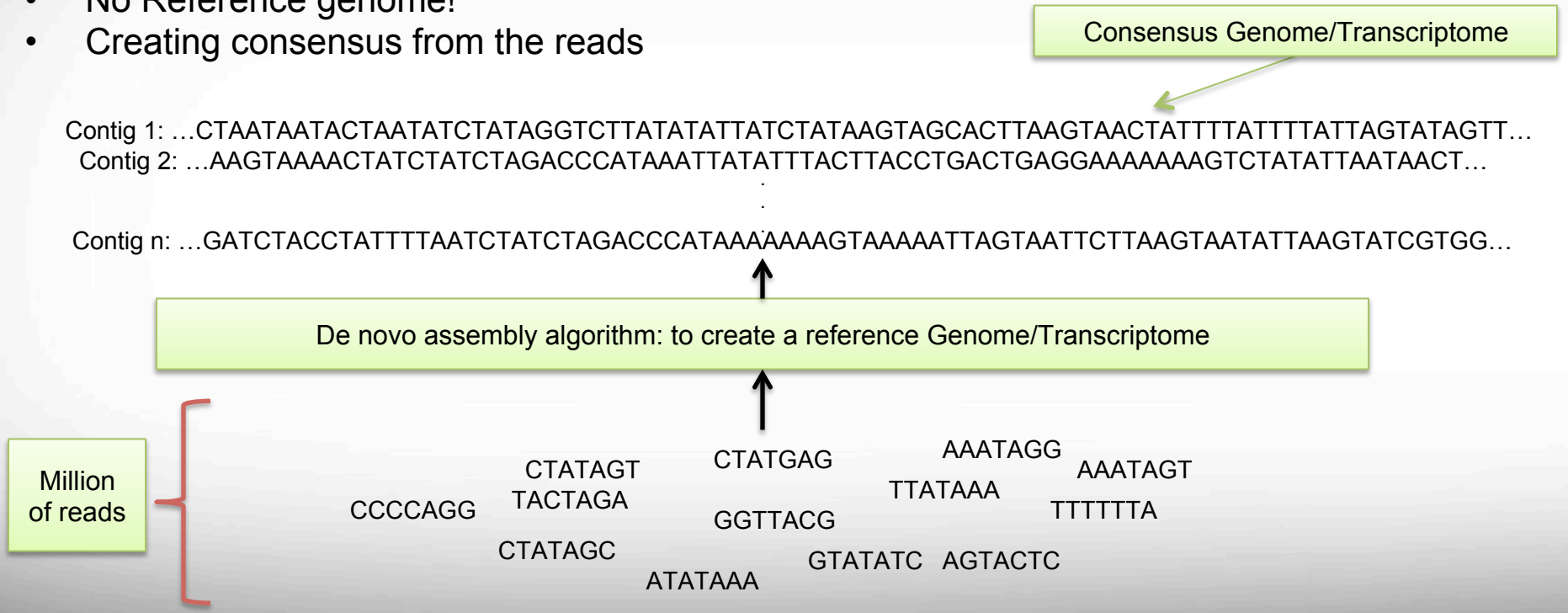
Genome versus Transcriptome Assembly



- Transcriptome assembly can be challenging:
  - *Uneven coverage*
  - *Splicing, multiple contigs per locus*
  - *Numerous transcripts*

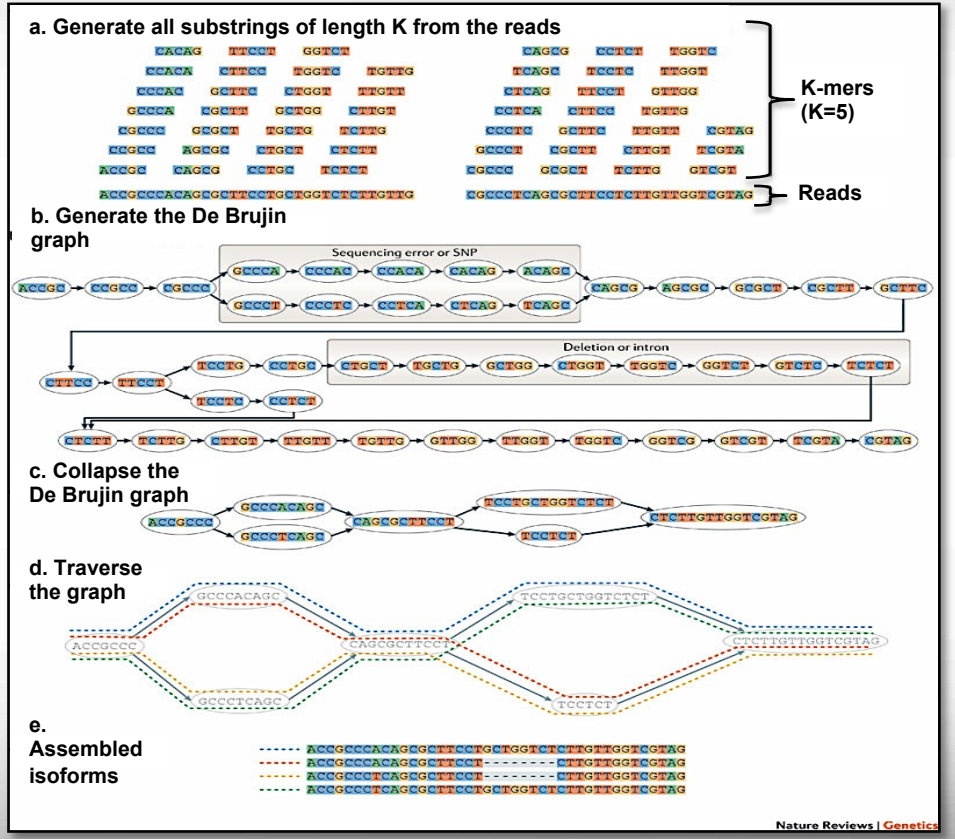
De novo Assembly

- Pool of reads
- No Reference genome!
- Creating consensus from the reads



De novo Assembly - 2

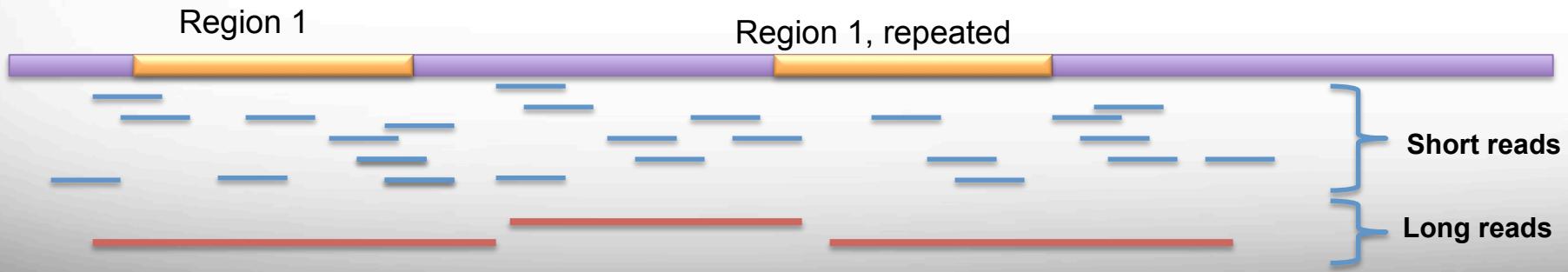
- Connection reads by finding common sections of kmers
  - Kmers are made from reads!
- Resolving conflicts
- Complicated process!
- Highly computational resource demanding!



Miller JR, Koren S, Sutton G. Assembly algorithms for next-generation sequencing data. Genomics. 2010 Jun;95(6):315-27. doi: 10.1016/j.ygeno.2010.03.001. Epub 2010 Mar 17. PMID: 20211242; PMCID: PMC2874646.
Compeau PE, Pevzner PA, Tesler G. How to apply de Bruijn graphs to genome assembly. Nat Biotechnol. 2011 Nov 15;29(11):987-91. doi: 10.1038/nbt.2023. PMID: 22068540; PMCID: PMC5531759.
Nagarajan N, Pop M. Sequence assembly demystified. Nat Rev Genet. 2013 Mar;14(3):157-67. doi: 10.1038/nrg3367. Epub 2013 Jan 22. PMID: 23358380.
Martin JA and Wang Z. Next-generation transcriptome assembly. Nature Reviews Genetics 12, 671-682, 2011

Comparing sequencing technologies - Repeats

Sanger	Next Generation Sequencing
Low coverage depth	High coverage depth
High cost for large genomes	Relatively low cost, even for large genomes
Slow	Fast
Handles repeats well	Need long reads for repeated regions (e.g. PacBio, Illumina Mate-Pair)



Genome Assembly Tools:

- ALLPATHS
- ALLPATHS-LG (Special recipe: fragments + jumping libraries)
- ABySS
- EULER-SR
- SOAPDenovo
- VCAKE
- Velvet
- CLC Bio Genomics Workbench

Transcriptome Assembly Tools:

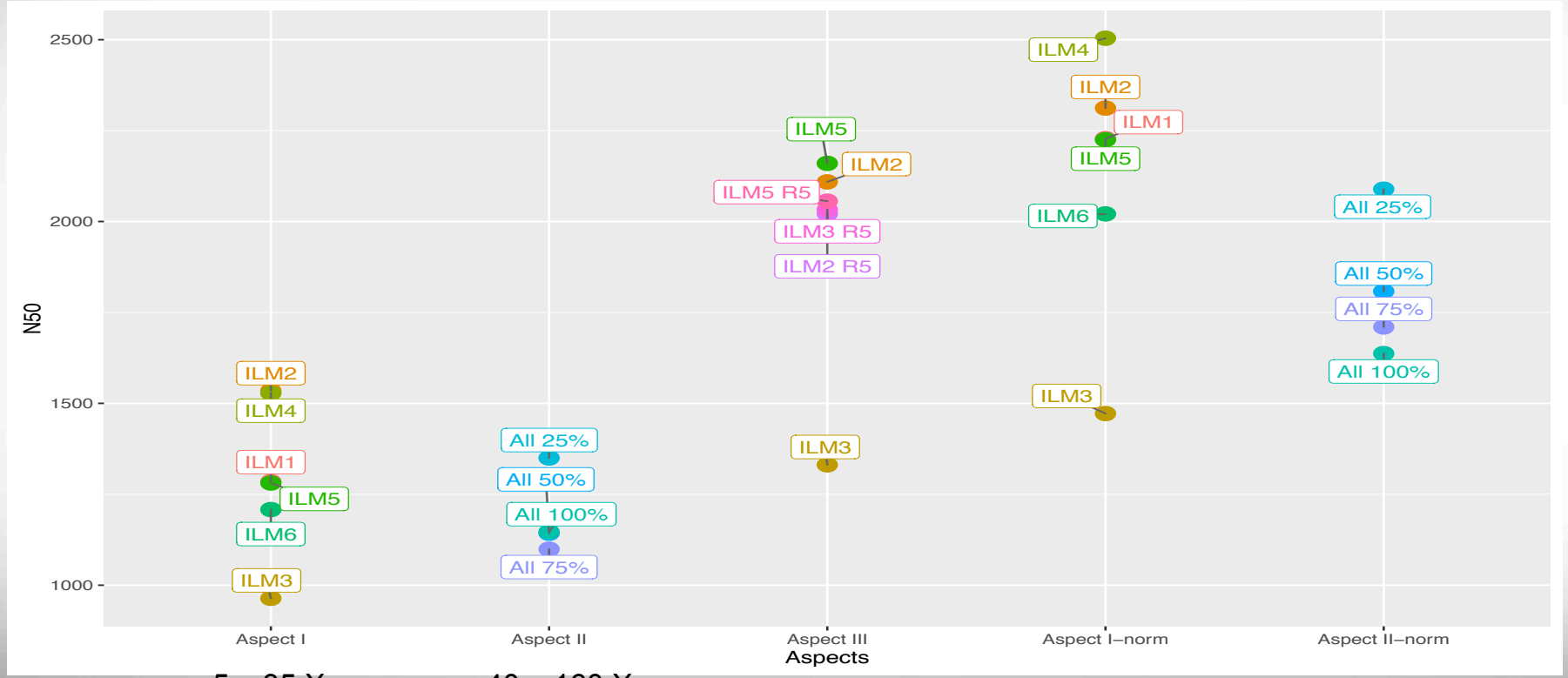
- SOAPdenovo-Trans
- Trans-ABySS
- Velvet + Oases
- Trinity
- Rnnotator
- CLC Bio Genomics Workbench

High Quality Assembly

- Hybrid Approach
  - Short reads with high coverage and high quality + long reads with lower quality score, but capable of covering repeats
- High Coverage
  - Ideally with 50+X long reads
- Merging
  - Metassembler, **Metassembler: merging and optimizing de novo genome assemblies**

Reasonable Coverage!

SEQC Evaluation Study by Ghaffari et. al.



5 – 35 X

40 – 160 X

2.7 – 4.6 X

Long Read/Hybrid Assembly Tools

- HGAP: PacBio only – Celera Based
- PBcR: PacBio & ONT – Celera Based
- LQS : ONT – Celera Based
- Falcon: PacBio & ONT – Celera Based
- Canu: PacBio & ONT – Celera Based
- Miniasm: PacBio & ONT – NO error correction!
- ALLPATHS-LG: Hybrid Assembly, using PacBio and Illumina
- SPAdes: Hybrid Assembly, using PacBio or ONT and Illumina

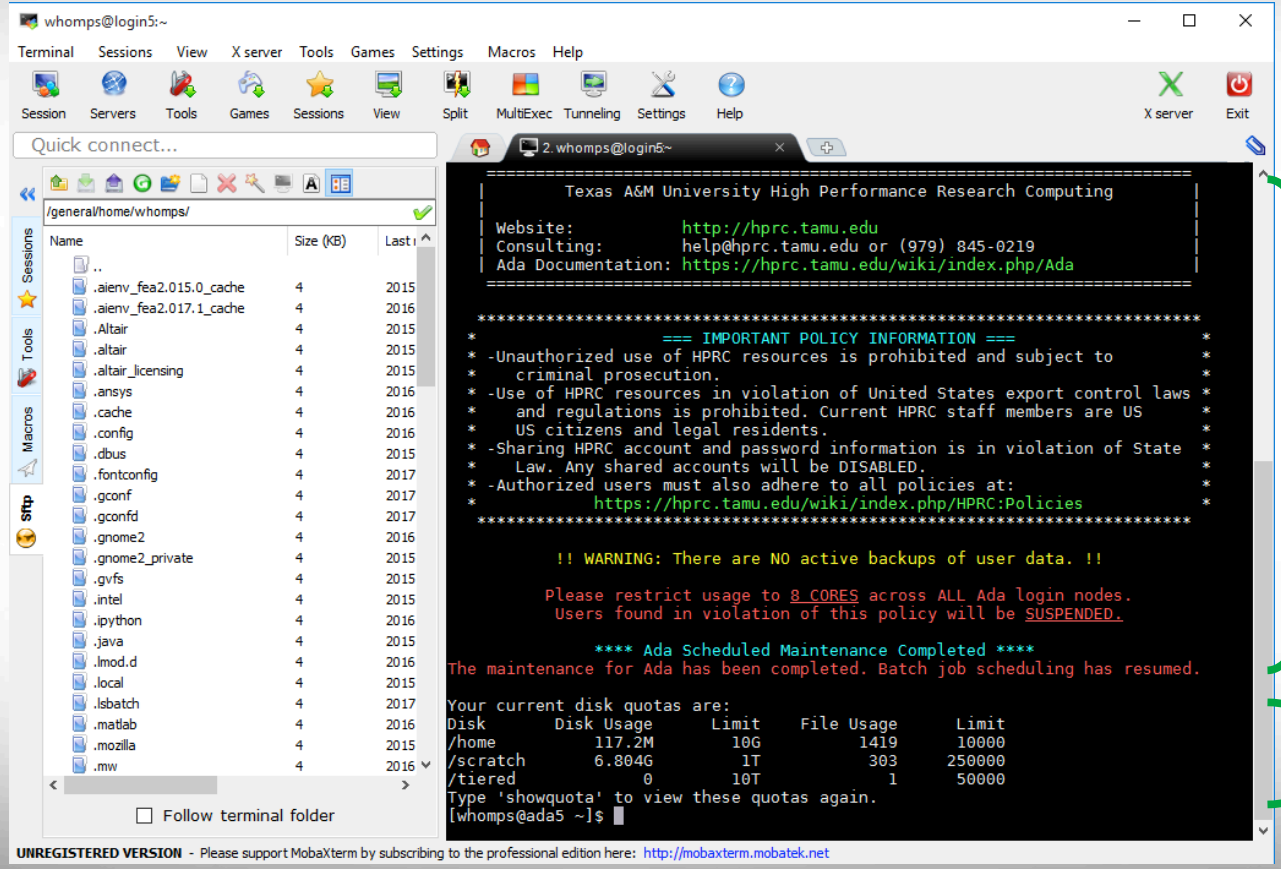
Practical Portion

Logging in to the system

- SSH (secure shell)
  - The only program allowed for remote access; encrypted communication; freely available for Linux/Unix and Mac OS X hosts;
- For Microsoft Windows PCs, use *MobaXterm*
    - https://hprc.tamu.edu/wiki/HPRC:MobaXterm
      - You are able to view images and use GUI applications with MobaXterm
  - or *Putty*
    - https://hprc.tamu.edu/wiki/HPRC:Access#Using_PuTTY
      - You can not view images or use GUI applications with PuTTY

- Both state of Texas law and TAMU regulations prohibit the sharing and/or illegal use of computer passwords and accounts
- Don't write down passwords
- Don't choose easy to guess/crack passwords
- Change passwords frequently

Using SSH - MobaXterm (on Windows)



message of the day

your quotas

Using SSH to Access Ada

```
ssh user_NetID@ada.tamu.edu
```

https://hprc.tamu.edu/wiki/Ada:Access

You may see something like the following the first time you connect to the remote machine from your local machine:

```
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)?
```

Type yes, hit enter and you will then see the following:

```
Host 'ada.tamu.edu' added to the list of known hosts.
user_NetID@ada.tamu.edu's password:
```

Transcriptome Assembly Practice

- The material are based on Trinity developers' workshop series, available on Github
  - https://github.com/trinityrnaseq/BerlinTrinityWorkshop2017
- RNA-Seq paired-end experiment consists of 3 conditions of growing Candida glabrata (yeast)
  - Wild type (WT)
  - Alkaline (ph8)
  - Nitosative challenge (GSNO)
- 2M total sampled read, with 3 biological replicates per sample

Defining the transcriptomic landscape of Candida glabrata by RNA-Seq". Linde et al. Nucleic Acids Res. 2015

Login and Set up

- Login to Ada using SSH or MobaXterm
- Let's take a look at the path and create appropriate directories

```
echo $SCRATCH
cd $SCRATCH
Pwd
mkdir NGS_assembly_Oct17
mkdir NGS_assembly_Oct17/Data
mkdir NGS_assembly_Oct17/Scripts
mkdir NGS_assembly_Oct17/Outputs
```

Inspecting the data

- The data is available on a shared folder, accessible to all attendees
- Copying the data to user's local space

```
cp /scratch/training/NGS_assembly/Data/Fastq_files/*.fastq \
$SCRATCH/NGS_assembly_Oct17/Data
```

Let's take a look at the data

```
cd $SCRATCH/NGS_assembly_Oct17/Data
ls -l
head -n 16 GSNO_rep1_1.fastq
```

Data processing

```
wc -l GSNO_rep1_1.fastq
```

There are 2 million reads in each file. Assembly will take long, thus, we will select portions of 4 replicates from 2 samples for the exercise.

```
head -n400000 GSNO_rep1_1.fastq > left_GSNO.100k.fastq
head -n400000 GSNO_rep1_2.fastq > right_GSNO.100k.fastq

head -n400000 ph8_rep1_1.fastq > left_ph8.100k.fastq
head -n400000 ph8_rep1_2.fastq > right_ph8.100k.fastq
```

Running the Assembly

- We are using Trinity software for assembling the sample data
  - Trinity: reconstructing a full-length transcriptome without a genome from RNA-Seq data
- Here are tips on how to run Trinity on TAMU HPRC Ada system
  - https://hprc.tamu.edu/wiki/Ada:NGS:RNA-seq#Example_Trinity_Tutorials
- You may use the GCATemplates tool on Ada to copy a sample code for

```
module load GCATemplates
gcatemplates
```

Sample Trinity Script - 1

```
#BSUB -L /bin/bash              # uses the bash login shell to initialize the job's execution environment.
#BSUB -J trinity_wo_ref_genome  # job name
#BSUB -n 20                     # assigns 20 cores for execution
#BSUB -R "span[ptile=20]"       # assigns 20 cores per node
#BSUB -R "rusage[mem=2700]"     # reserves 2700MB memory per core
#BSUB -M 2700                   # sets to 2700MB (~2.7GB) per process enforceable memory limit. (M * n)
#BSUB -W 48:00                  # sets to 48 hours the job's runtime wall-clock limit.
#BSUB -o stdout.%J              # directs the job's standard output to stdout.jobid
#BSUB -e stderr.%J              # directs the job's standard error to stderr.
```

Sample Trinity Script - 2

```
module load Trinity/2.2.0-intel-2015B

<<README
    - Trinity: assembles transcript sequences from Illumina RNA-Seq data.
    - Trinity manual: https://github.com/trinityrnaseq/trinityrnaseq/wiki
README

##############################################################################
# TODO Edit these variables as needed:
se_1='c_reinhardtii_rna_seq_SRR1179643_1.fasta'
seqType='fa'                    # fa, fq

threads=20                      # make sure this is <= your BSUB -n value

##############################################################################
```

Sample Trinity Script - 3

```
# Assemble RNA-seq data; Find assembled transcripts as: 'trinity_out_dir/Trinity.fasta'
Trinity --seqType $seqType --max_memory 53G --single $se_1 --CPU $threads --no_version_check --
inchworm_cpu 6

<<CITATION
    - Acknowledge TAMU HPRC: https://hprc.tamu.edu/wiki/index.php/HPRC:AckUs

    - Trinity citation:
        Full-length transcriptome assembly from RNA-Seq data without a reference genome.
        Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L,
        Raychowdhury R, Zeng Q, Chen Z, Mauceli E, Hacohen N, Gnirke A, Rhind N, di Palma F,
        Birren BW, Nusbaum C, Lindblad-Toh K, Friedman N, Regev A.
        Nature Biotechnology 29, 644–652 (2011)
CITATION
```

Trinity Script

- Trinity script that will be used for this session is provided on the training directory. We will copy that to the users account and then submit it to the scheduler system.

```
cp /scratch/training/NGS_assembly/Scripts/Trinity_* \
$SCRATCH/NGS_assembly_Oct17/Scripts

cd $SCRATCH/NGS_assembly_Oct17/Scripts
ls –l
cat Trinity_GSNO_ph8_100K.sh
```

Submitting the Script

- After the script is copied, users can submit the job to Ada scheduler for running, and monitor its progress

```
bsub < Trinity_GSNO_ph8_100K.sh

bjobs
```

Completed Assembly

- The data and the script that we used in the practical session, are based on a very small subset of data. We have the assembly based on the complete dataset and we will use it in the next steps.

```
cd $SCRATCH/NGS_assembly_Oct17/Outputs
mkdir All_Data
cd All_Data
cp /scratch/training/NGS_assembly/Data/workshop_shared/shared/Trinity.fasta  .
```

Assembly Evaluation

There are multiple quality control metrics to ensure the quality of the assembly.

Quality Control

- After the assembly job is completed, the basic statistics of the assembly can be accessed using a script provided by Trinity tool
  - % $TRINITY_HOME/util/TrinityStats.pl Trinity.fasta
  - Next slide shows the output of the TrinityStats script for the whole dataset

Assembly Stats - 1

- ##################################
- ## Counts of transcripts, etc.
- ##################################
- Total trinity 'genes': 7648
- Total trinity transcripts: 7719
- Percent GC: 38.88
- ##
- Stats based on ALL transcript contigs:
- ##
- Contig N10: 4318
- Contig N20: 3395
- Contig N30: 2863
- Contig N40: 2466
- ***Contig N50: 2065***

N50: at least half of the contigs are this length or longer

Assembly Stats - 2

- Median contig length: 1038
- Average contig: 1354.26
- Total assembled bases: 10453524
- ##
- ## Stats based on ONLY LONGEST ISOFORM per 'GENE':
- ##
- Contig N10: 4317
- Contig N20: 3375
- Contig N30: 2850
- Contig N40: 2458
- Contig N50: 2060
- Median contig length: 1044
- Average contig: 1354.49
- Total assembled bases: 10359175

Read Representation - 1

- Assembly algorithms use subset of reads, called k-mers to reconstruct the transcripts. Since the k-mers are shorter than reads, aligning reads to the resulting contigs will not provide a perfect mapping. However, high read representation is an indicator of high quality assembly.
- The Bowtie tool can be used for mapping reads to the contigs. Here are the two steps needed to run the mapping. We will use the already ran output for examination since mapping is a time consuming process.

1) Building a bowtie2

```
% bowtie2-build Trinity.fasta Trinity.fasta
```

2) Aligning the reads:

```
% bowtie2 --local --no-unal -x Trinity.fasta -q -1 left.100k.fastq -2 right.100k.fastq \ | samtools view -Sb - | samtools sort -no - - > bowtie2.nameSorted.bam
```

Read Representation - 2

SAM header is present: 7719 sequences.

100000 reads; of these:

100000 (100.00%) were paired; of these:

1396 (1.40%) aligned concordantly 0 times

92183 (92.18%) aligned concordantly exactly 1 time

6421 (6.42%) aligned concordantly >1 times

1396 pairs aligned concordantly 0 times; of these:

372 (26.65%) aligned discordantly 1 time

1024 pairs aligned 0 times concordantly or discordantly; of these:

2048 mates make up the pairs; of these:

1483 (72.41%) aligned 0 times

314 (15.33%) aligned exactly 1 time

251 (12.26%) aligned >1 times

99.26% overall alignment rate

> 70% reads mapped as pairs is desirable

Useful QC Matrics

- ExN50: the concept is similar to N50, but it is based on highly expressed transcripts.
- Steps for calculating ExN50
  - Assembling all the data into one single fasta file
  - Mapping each sample data to the fasta file
  - Finding expression levels for each transcript
  - Normalizing the expression levels using edgeR TMM method
  - Running contig_ExN50_statistic.pl
- Copy the sample script

```
cd /scratch/training/NGS_assembly/Scripts
cp Bowtie_RSEM.sh $SCRATCH/NGS_assembly_Oct17/Scripts
cat Bowtie_RSEM.sh
```

Mapping reads and ExN50 Script - 1

```
#BSUB -L /bin/bash
#BSUB -J Bowtie_RSEM
#BSUB -o stdout.%J
#BSUB -e stderr.%J
#BSUB -n 2
#BSUB -R "span[ptile=2]"
#BSUB -R "rusage[mem=2700]"
#BSUB -M 2700
#BSUB -W 4:00

module load Bowtie2/2.2.6-intel-2015B
module load SAMtools/1.3-intel-2015B
module load Trinity/2.2.0-intel-2015B
module load RSEM/1.2.29-intel-2015B
module load R_tamu/3.3.1-intel-2015B-default-mt
```

Mapping reads and ExN50 Script - 2

```
left='left_GSNO.100k.fastq'
right='right_GSNO.100k.fastq'
fasta_file='/scratch/user/noushin/NGS_assembly_Oct17/Outputs/Trinity_Output_GSNO_ph8_100K/
Trinity.fasta'

$TRINITY_HOME/util/align_and_estimate_abundance.pl  --transcripts $fasta_file --seqType fq --left
$SCRATCH/NGS_assembly_Oct17/Data/$left --right $SCRATCH/NGS_assembly_Oct17/Data/$right --
est_method RSEM --aln_method bowtie2 --trinity_mode --prep_reference --output_prefix GSNO_100K --
output_dir $SCRATCH/NGS_assembly_Oct17/Outputs/Trinity_Output_GSNO_ph8_100K/
RSEM_output_GSNO

left='left_ph8.100k.fastq'
right='right_ph8.100k.fastq’

$TRINITY_HOME/util/align_and_estimate_abundance.pl  --transcripts $fasta_file --seqType fq --left
$SCRATCH/NGS_assembly_Oct17/Data/$left --right $SCRATCH/NGS_assembly_Oct17/Data/$right --
est_method RSEM --aln_method bowtie2 --trinity_mode --prep_reference --output_prefix ph8_100K --
output_dir $SCRATCH/NGS_assembly_Oct17/Outputs/Trinity_Output_GSNO_ph8_100K/RSEM_output_ph8
```

Mapping reads and ExN50 Script - 3

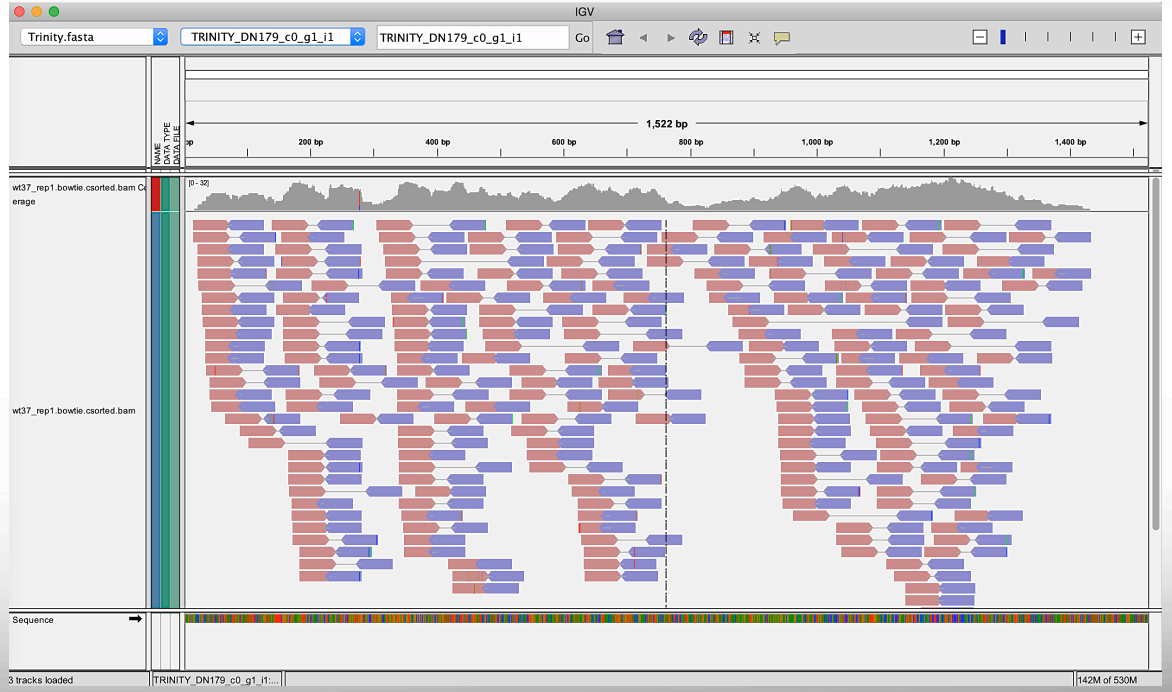
#Creating the count table
$TRINITY_HOME/util/**abundance_estimates_to_matrix.pl** --est_method RSEM --out_prefix Trinity_trans $SCRATCH/NGS_assembly_Oct17/Outputs/Trinity_Output_GSNO_ph8_100K/RSEM_output_GSNO/GSNO_100K.isoforms.results $SCRATCH/NGS_assembly_Oct17/Outputs/Trinity_Output_GSNO_ph8_100K/RSEM_output_ph8/ph8_100K.isoforms.results

$TRINITY_HOME/util/misc/**contig_ExN50_statistic.pl** Trinity_trans.TMM.EXPR.matrix $fasta_file > ExN50.stats

$TRINITY_HOME/util/misc/**plot_ExN50_statistic.Rscript** ExN50.stats

Additional QC Metrics

- Visualizing the mapping
- IGV, covered in last week course
- Needs the Trinity.fasta as the reference, and the reads that are used for the assembly



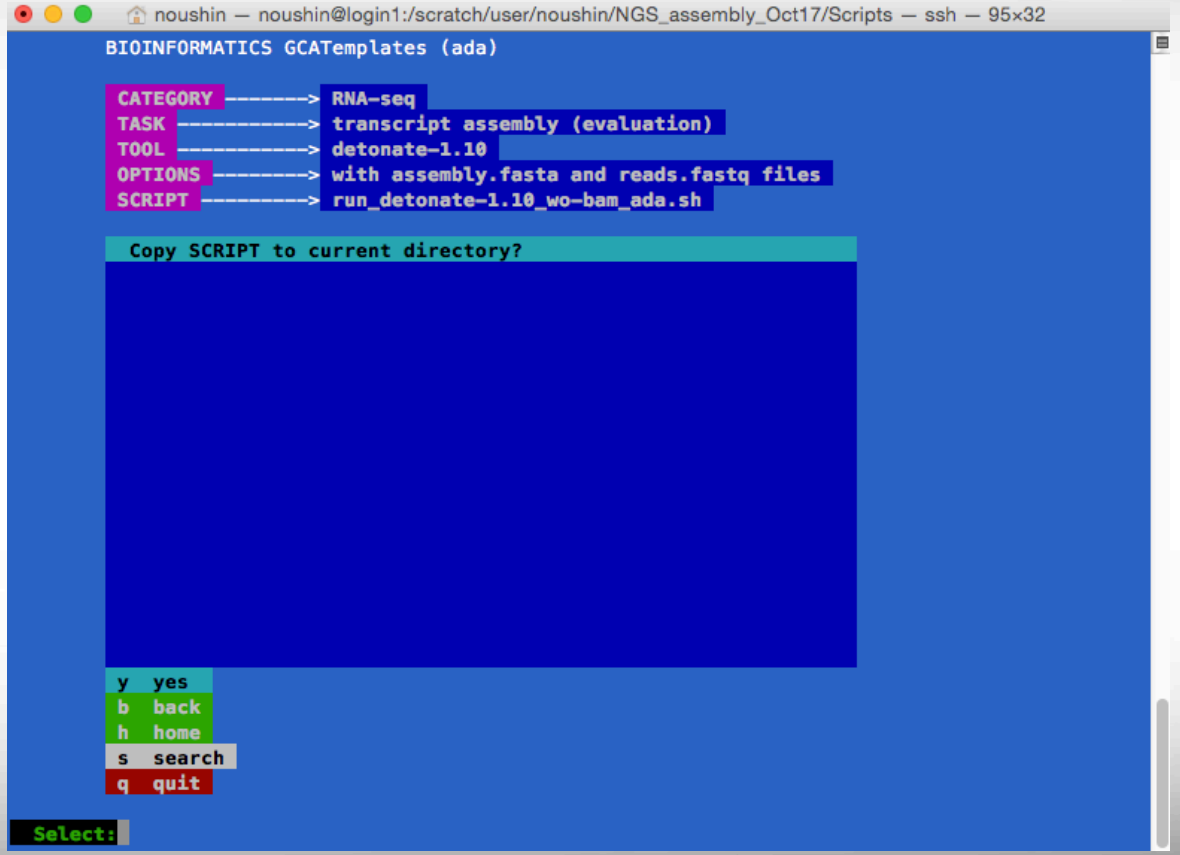
Additional QC Metrics

- DETONATE

DE novo TranscriptOme rNaseq Assembly with or without the Truth Evaluation

  - RSEM-EVAL
  - REF-EVAL

- To study the k-mer proportions present in the transcriptome compared to that in the reference



DETONATE Sample Code - 1

```
#BSUB -L /bin/bash              # uses the bash login shell to initialize the job's execution environment.
#BSUB -J detonate               # job name
#BSUB -n 4                      # assigns 4 cores for execution
#BSUB -R "span[ptile=4]"        # assigns 4 cores per node
#BSUB -R "rusage[mem=500]"      # reserves 500MB memory per core
#BSUB -M 500                    # sets to 500MB per process enforceable memory limit. (M * n)
#BSUB -W 1:00                   # sets to 1 hour the job's runtime wall-clock limit.
#BSUB -o stdout.%J              # directs the job's standard output to stdout.jobid
#BSUB -e stderr.%J              # directs the job's standard error to stderr.jobid

module load DETONATE/1.10-intel-2015B-jkp
module load Bowtie/1.1.2-intel-2015B

<<README
   - DETONATE manual:
      rsem-eval-calculate-score [options] upstream_read_file(s) assembly_fasta_file sample_name L
      rsem-eval-calculate-score [options] --paired-end upstream_read_file(s) downstream_read_file(s)
assembly_fasta_file sample_name L
      rsem-eval-calculate-score [options] --sam/--bam [--paired-end] input assembly_fasta_file sample_name L
README
```

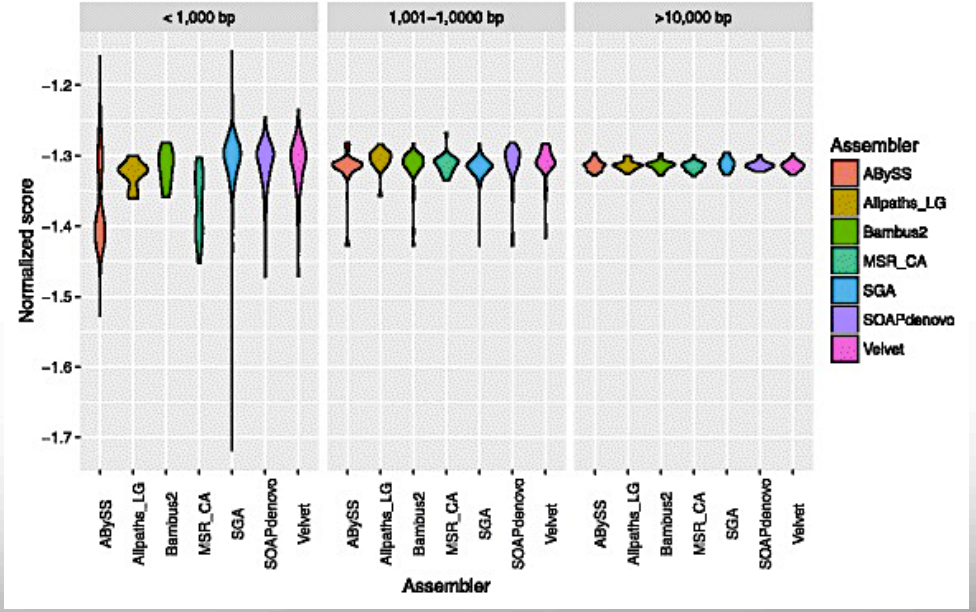
DETONATE Sample Code - 2

```
################################################################################
# TODO Edit these variables as needed:
assembled_transcripts='/scratch/datasets/GCATemplates/data/sra/e_coli/ecoli_rna-
seq_assembly_SRR575493_Trinity.fasta'
reads_fastq='/scratch/datasets/GCATemplates/data/sra/e_coli/ecoli_rna-seq_reads_SRR575493.fastq'

################################################################################
#
rsem-eval-calculate-score $reads_fastq $assembled_transcripts output 50

<<CITATION
    - Acknowledge TAMU HPRC: http://hprc.tamu.edu/research/citation.php

    - DETONATE:
        Bo Li, Nathanael Fillmore, Yongsheng Bai, Mike Collins, James A. Thomson, Ron Stewart,
        and Colin N. Dewey. Evaluation of de novo transcriptome assemblies from RNA-Seq data. Genome
Biology 2014, 15:553.
CITATION
```

Comparing Assemblies or QC

HiMMe: using genetic patterns as a proxy for genome assembly reliability assessment

- HMM-based tool
- Relies on genetic patterns to score genome assemblies
- Using Markov chain, the model is able to detect characteristic genetic patterns, while, by introducing emission probabilities, the noise involved in the process is taken into account.
- Prior knowledge can be used by training the model to fit a given organism or sequencing technology, e,g. SNP database for the species



Any question?

nghaffari@tamu.edu