Introduction to NGS Data Analysis on the HPRC Clusters
Your Login Password

- 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 passwords
- Change passwords frequently

There will be a 10 minute break halfway through today's short course
For More Help...

Website:  hprc.tamu.edu
Email:    help@hprc.tamu.edu
Telephone: (979) 845-0219
Visit us in person: Henderson Hall, Room 114A

Help us, help you -- we need more info
• Which Cluster
• UserID/NetID
• Job id(s) if any
• Location of your jobfile, input/output files
• Application used if any
• Module(s) loaded if any
• Error messages
• Steps you have taken, so we can reproduce the problem
Using SSH - MobaXterm (on Windows)
Next Generation Sequencing (NGS)
## Illumina Sequencing Technology

<table>
<thead>
<tr>
<th></th>
<th>MiniSeq System</th>
<th>MiSeq Series</th>
<th>NextSeq Series</th>
<th>HiSeq Series</th>
<th>HiSeq X Series*</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Key Methods</strong></td>
<td>Amplicon, targeted RNA, small RNA, and targeted gene panel sequencing.</td>
<td>Small genome, amplicon, and targeted gene panel sequencing.</td>
<td>Everyday exome, transcriptome, and targeted resequencing.</td>
<td>Production-scale genome, exome, transcriptome sequencing, and more.</td>
<td>Population-scale whole-genome sequencing.</td>
</tr>
<tr>
<td><strong>Maximum Output</strong></td>
<td>7.5 Gb</td>
<td>15 Gb</td>
<td>120 Gb</td>
<td>1500 Gb</td>
<td>1800 Gb</td>
</tr>
<tr>
<td><strong>Maximum Reads per Run</strong></td>
<td>25 million</td>
<td>25 million</td>
<td>400 million</td>
<td>5 billion</td>
<td>6 billion</td>
</tr>
<tr>
<td><strong>Maximum Read Length</strong></td>
<td>2 x 150 bp</td>
<td>2 x 300 bp</td>
<td>2 x 150 bp</td>
<td>2 x 150 bp</td>
<td>2 x 150 bp</td>
</tr>
<tr>
<td><strong>Run Time</strong></td>
<td>4–24 hours</td>
<td>4–55 hours</td>
<td>12–30 hours</td>
<td>&lt;1–3.5 days (HiSeq 3000/HiSeq 4000) 7 hours–6 days (HiSeq 2500)</td>
<td>&lt;3 days</td>
</tr>
<tr>
<td><strong>Benchtop Sequencer</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>


---

*NovaSeq 5000

- Same as HiSeq
- 2000 Gb
- 6.6 billion
- 2 x 150 bp
- 19 - 40 hrs
- no
Illumina Sequencing Technology

- **iSeq™**: 1.2 Gb
- **MiniSeq™**: 7.5 Gb
- **MiSeq®**: 15 Gb
- **NextSeq®**: 120 Gb
- **HiSeq®**: 1.5 Tb
- **HiSeq® X**: 1.8 Tb
- **NovaSeq®**: 1 Tb - 6 Tb

**Output per run**

- **Small whole genomes, targeted sequencing, (non-metagenomic)**
Illumina Sequencing Libraries

**single end**

- Genomic DNA
- Fragment (200-500bp)
- Ligate Adaptors
- ~400 bp sequenced region (read length)

**paired ends**

- Genomic DNA
- Fragment (200-500bp)
- Ligate Adaptors
- ~400 bp sequenced regions (read length)

**mate pairs**

- Genomic DNA is fragmented with a Mate Pair Fragment Enzyme, which attaches a biotinylated junction adapter (green) to both ends of the fragmented molecule.
- ~2,000 bp “insert size”
- ~400 bp “fragment size”
- ~2,000 bp sequenced regions (read length)
**Paired End Reads**

- **Read 1 of a pair**
  - 100 base read length

- **dsDNA**
  - ~200 bp fragment size
  - ~400 bp fragment size

- **Read 2 of a pair**
  - 100 base read length

**FASTQ format**

**Read 1 pair fastq file**

@M00861:1:000000000-A36BE:1:1101:14650:1529 1:N:0:8
TTCTTAAAAATACCATAAAAGCTTAAACTTGCCATTTACGGATTAATTCCATCTCTCTTCTTTTGCCTATCTCTTCTTTTAAAGTTAAATGACTCATAACGG
+
FFFHBFHFHHIIIHIFIHFHCHGFGHGHIIHHHHD/ ?DGHHH@DB, 5EHHGHHIHIHIF?FGGGCHCBDFGHFHDGHGFFFGDFHH?DFHDFHFHFHFHFFFHHH

**Read 2 pair fastq file**

@M00861:1:000000000-A36BE:1:1101:14650:1529 2:N:0:8
ACTAAAAATCAATTCTATCTATATTCTACACTTTATCTATCTATTTAAGGTGATGGCCTTAAAATAATAATTGCATATTTTGCATTTTTGCAATAGCGG
+
BFFHIHHHHFHDFHIHIHHHHHHHHHHFDFHHHIHIIHIIH=AAHFHHFCHGFHHHHHGGHHHIHFHFFEGGHHDGGHHH/CGHIFHHH
Maintain Read Pair Order

DNA Fragment lengths will be different but sequence reads may all be the same length.

Left Read 1 paired end fastq file

Right Read 2 paired end fastq file
MiSeq Can Perform Initial QC Trimming

DNA Fragment lengths will be different but sequence reads can have different lengths

Left Read 1 paired end fastq file

Right Read 2 paired end fastq file
PacBio Long Read Sequencing

Sequel Sequencer

Read lengths >20 kb
Data per SMRT Cell: 5–8 Gb

Half of data in reads: >20 kb

Top 5% of reads: >35 kb

Maximum read length: >60 kb

pacb.com
PacBio Long Read Sequencing

SMRTbell Template

Polymerase Read

Subreads

Circular Consensus Sequence
(Read of Insert)

+ Strand yellow
- Strand purple

Shorter DNA fragment equals more subreads

pacb.com

m54001_160302_121501.subreads.bam
12345

[1] “m” = movie
[2] Instrument Serial Number
PacBio Sequencing Tools

- **Sequence Alignments**
  - Minimap2, pbalign, blasr (pbbioconda)

- **Correct PacBio reads with Illumina reads (computationally intensive)**
  - Proovread,
  - LSC

- **Genome Assembly**
  - Canu: PacBio long read assembler
    - run in grid mode on Curie with no SUs charged
  - Unicycler

- **Improve draft assemblies**
  - ArrowGrid_HPRC (Terra)
  - Purge_Haplotigs (Terra)
  - Circlator

https://hprc.tamu.edu/wiki/Bioinformatics:PacBio_tools
NGS Tools on Ada
Where to Find NGS Tools

- TAMU HPRC Documentation
  - https://hprc.tamu.edu/wiki/index.php/Ada:Bioinformatics
- Type the following UNIX commands to see which tools are already installed on Ada
  - module avail
  - module spider toolname
  - module key assembly
    (not case sensitive, but read the entire output)
    (some modules may be missed because this searches tool descriptions)
- If you find a tool that you want installed on Ada, send an email with the URL link to: help@hprc.tamu.edu
  - SeqAnswers http://seqanswers.com/wiki/Software/list
  - Omictools.com
  - slideshare.net – find shared NGS presentations
Ada Software Toolchains

- Use the same toolchains in your job scripts
  
  Use the same toolchains in your job scripts with the following example:
  
  ```bash
  module load Bowtie2/2.2.6-intel-2015B
  module load TopHat/2.1.0-intel-2015B
  module load Cufflinks/2.2.1-intel-2015B
  ```

- Avoid loading mixed toolchains:
  
  ```bash
  module load Bowtie2/2.2.2-ictce-6.3.5
  module load TopHat/2.0.14-goolf-1.7.20
  module load Cufflinks/2.2.1-intel-2015B
  ```

- Avoid loading defaults which may have different toolchains
  
  ```bash
  module load Bowtie2 TopHat Cufflinks
  ```
The GCCcore Toolchain

● To minimize the number of software builds, the GCCcore-6.3.0 toolchain modules can be loaded alone or with any one of the following 2017A toolchains
  ○ intel/2017A
  ○ iomkl/2017A
  ○ foss/2017A

● Example of loading a GCCcore module with a 2017A module

```bash
module load Bowtie2/2.3.3.1-GCCcore-6.3.0
module load TopHat/2.1.1-intel-2017A-Python-2.7.12
```
Python-version-bare modules

- You need to load a non ‘-bare’ Python version along with the -bare module
  - If you do not, then the older default OS Python version will be used
- Used in conjunction with GCCcore builds in order to reduce the number of software modules built.

intell/2017A iomkl/2017A foss/2017A

Three Examples of loading GCCcore Python -bare and a Python module with a 2017A toolchain

1. module load Cython/0.25.2-GCCcore-6.3.0-Python-2.7.12-bare
   module load Python/2.7.12-foss-2017A

2. module load Cython/0.25.2-GCCcore-6.3.0-Python-2.7.12-bare
   module load Python/2.7.12-iomkl-2017A

3. module load Cython/0.25.2-GCCcore-6.3.0-Python-2.7.12-bare
   module load HISAT2/2.1.0-intel-2017A-Python-2.7.12

Loads Python indirectly
Use $\text{TMPDIR}$ whenever possible

- Use the $\text{TMPDIR}$ if the application you are running can utilize a temporary directory for writing temporary files which are deleted when the job ends.
- A temp directory ($\text{TMPDIR}$) is automatically assigned for each job which uses the disk(s) on the compute node not the $\text{SCRATCH}$ shared file system:
  - Especially useful when a computational tool writes tens of thousands of temporary files which are deleted when the job is finished and are not needed for the final results.
  - This is useful since files on $\text{TMPDIR}$ will not count against your file quota.
  - Don't use $\text{TMPDIR}$ if your software uses temporary files for restarting where it left off if it should stop before completion.
  - Will significantly speed up an mpiBLAST job.

```bash
java -Xmx53g -jar $EBROOTPICARD/FastqToSam.jar TMP_DIR=$\text{TMPDIR} \ 
FASTQ=$\text{pe1}_1$ FASTQ2=$\text{pe1}_2$ OUTPUT=$\text{outfile}$ SAMPLE_NAME=$\text{sample_name}$ \ 
SORT_ORDER=$\text{sort_order}$ MAX_RECORDS_IN_RAM='null'
```
Template Job Scripts
Access GCATemplate Scripts for Ada from the HPRC wiki

https://hprc.tamu.edu/wiki/Bioinformatics:Sequence_QC#FastQC

Genomic Computational Analysis Templates

Click to see template script on github
```bash
Mesub -l /lx/bash  # uses the bash login shell to initialize the job's execution environment.
Mesub -J 'FastQC'  # job name
Mesub -N 1   # assigns 1 core for execution
Mesub -R 'span[64:65]' # assigns 2 cores per node
Mesub -R 'rusage[mem=256M0]' # reserves 256MB memory per core
Mesub -W 256M # sets to 256MB process enforceable memory limit. (M * n)
Mesub -W 1:00 # sets to 1 hour the job's runtime wall-clock limit.
Mesub -o stdout.js  # directs the job's standard output to stdout.js
Mesub -e stderr.js  # directs the job's standard error to stderr.js

module load FastQC/0.11.6-Java-1.8.0

# README
- FastQC homepage: http://www.babraham.cam.ac.uk/projects/fastqc/

# T2/2016: these variables as needed:
threads=2;  # make sure this is = your sum - n value
seqfile=.../data/R1/R1_001.fastq.gz
seqfile=.../data/R2/R2_001.fastq.gz
```
Finding NGS job template scripts using GCATemplates on Ada

- Select #4 then find the template that contains fastqc
- Final step will save a template job script file to your current working directory
- After you save the template file:

  ```
  module purge
  ```

For practice, we will copy a template file

- `mkdir $SCRATCH/ngs_class`
- `cd $SCRATCH/ngs_class`
- `module load GCATemplates`
- `gcatemplates`
Sample GCATemplate Job Script (Ada)

```ada
#BSUB -L /bin/bash
#BSUB -J blastx
#BSUB -n 1
#BSUB -R "span[ptile=1]"
#BSUB -R "rusage[mem=2500]"
#BSUB -M 2500
#BSUB -W 2:00
#BSUB -o stdout.%J
#BSUB -e stderr.%J

module load BLAST+/2.2.31-intel-2015B-Python-3.4.3

<<README
README

# blastx: search protein databases using a translated nucleotide query

blastx -query mrna_seq_nt.fasta -db /scratch/datasets/blast/nr \
   -outfmt 10 -out mrna_seq_nt_blastout.csv
```
#BSUB -L /bin/bash
#BSUB -J blastx
#BSUB -n 1
#BSUB -R "span[ptile=1]"
#BSUB -R "rusage[mem=2500]"
#BSUB -M 2500
#BSUB -W 2:00
#BSUB -o stdout.%J
#BSUB -e stderr.%J

module load BLAST+/2.2.31-intel-2015B-Python-3.4.3

<<README
README

This is a section of comments

This is a single line comment and not run as part of the script

# blastx: search protein databases using a translated nucleotide query

blastx -query mrna_segs_nt.fasta -db /scratch/datasets/blast/nr \ -outfmt 10 -out mrna_segs_nt_blastout.csv

This is the command to run the application

These parameters are read by the job scheduler

Load the required module(s) first

This means the command is continued on the next line;
The space before the \ is required
Do not put a space after the \
Quality Control (QC)
QC Evaluation

● Use FastQC to visualize quality scores
  − Displays quality score distribution of a subset of ~200,000 reads
    ● Input is a fastq file or files
    ● Can disable grouping (binning) of sequence regions
  − Will alert you of poor read characteristics
  − Can be run as a GUI or a command line interface

module load FastQC/0.11.6-Java-1.8.0

● FastQC will process using one CPU core per file
  − If there are 10 fastq files to analyze and 4 cores used
    ● 4 files will start processing and 6 will wait in a queue
  − If there is only one fastq file to process then using 10 cores does not speed up the process
FastQC Exercise

- Use the GCATemplate for FastQC to submit a job evaluating the two sequence files
  - `gedit run_fastqc_0.11.6_ada.sh &`
  - `bsub < run_fastqc_0.11.6_ada.sh`

- After your fastqc job is complete, unzip the results file and you can view the results files with `lynx` and `eog` (eog requires X11 login)
  - `unzip DR34_R1_fastqc.zip`
FastQC Report using lynx

 lynx DR34_R1_fastqc.html
FastQC Output Image Quality Distribution

eog DR34_R1_fastqc/Images/per_base_quality.png

Prior to QC trimming

click for the next image in the same directory, or use the left/right arrow keys
FastQC Output Image Quality Distribution

@ERR504787.2.1 M00368:15:00000000-0-A0HKH:1:5:21261:10968-1 length=100
GATCGGAAGACGACGCTCTGACGTACGATCGACGTGTAGCTGTAGCTGAAAACAAAACACCAAACATAATGCCGTAAAA
+ERR504787.2.1 M00368:15:00000000-0-A0HKH:1:5:21261:10968-1 length=100
@ERR504787.3.1 M00368:15:000000000-0-A0HKH:1:3:12724:25677-1 length=100
GATTTGTTTGATTGGACAGTAGTTGTACGTTGCTTTACTTCTTGCTTTTCTTATTTAACACAAACCTCCTGCAAAGTATTATGGGAGATGTAGGTAGTT
+ERR504787.3.1 M00368:15:00000000-0-A0HKH:1:3:12724:25677-1 length=100
BCCFDEFFHHHHJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
@ERR504787.5.1 M00368:15:00000000-0-A0HKH:1:2:16161:12630-1 length=100
TATTTAAGTTGAACATGCTACCGAAATATGTAACGTAGTGCTTACGCTACGCTACGCTACGCTACGCTACGCTACGCTACGCTACGCTACGCTACGCT
+ERR504787.5.1 M00368:15:000000000-0-A0HKH:1:2:16161:12630-1 length=100
CCCFFFFFHHHHJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ

![Graph of FASTQ format](chart.png)
FastQC Output Image Quality Distribution

- @ERR504787.2.1 M00368:15:000000000-A0HKH:1:5:21261:10968-1 length=100
- @ERR504787.3.1 M00368:15:000000000-A0HKH:1:3:12724:25677-1 length=100
- @ERR504787.5.1 M00368:15:000000000-A0HKH:1:2:16161:12630-1 length=100

FASTQ format

Average quality score distribution at position one

Upper extreme
Upper quartile
Median
Lower quartile
Lower extreme
FastQC Output Image Quality Distribution

Positions are ‘binned’ after the first few positions
Illumina Transposon Insertion Site

Sequence content across all bases

Position in read (bp)
Illumina Transposon Insertion Site

![kmer_profiles.png](image-url)
FastQC Flowcell Quality Image

MiSeq flowcell

per_tile_quality.png

Flowcell quality mapping
Good per_tile quality

Position in read (bp)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

Quality per tile

good quality poor quality
FastQC Flowcell Quality Image

MiSeq flowcell

bottom of flowcell

top of flowcell

good quality  poor quality

per_tile_quality.png
Failed QC Examples
Example 1. Expired MiSeq mate-pair kit (9 months expired)
Example 2. Sequence prep adapters still on ends of DNA library fragments
FastQC Output Image

Flowcell: not good per_tile quality

Example 3. Faulty flowcell

MiSeq flowcell

good quality  poor quality
QC Quality Trimming

- Sequence quality trimming tools
  - module spider Trimmomatic
  - recommended tool
- Trimmomatic will maintain paired end read pairing after trimming
- Trim reads based on quality scores
  - Trim the same number of bases from each read or
  - Use a sliding window to calculate average quality at ends of sequences
- Decide if you want to discard reads with Ns
  - some assemblers replace Ns with As or a random base G, C, A or T
- Trim adapter sequences
  - Trimmomatic has a file of Illumina adapter sequences

  module load Trimmomatic/0.38-Java-1.8.0

  ls $EBROOTTRIMMOMATIC/adapters/
Paired End Short Reads

Read 1 of a pair
100 base read length

dsDNA

~200 bp

Read 2 of a pair
100 base read length

~400 bp fragment size

FASTQ format

Read 1 pair fastq file

@M00861:1:000000000-A36BE:1:1101:14650:1529 1:N:0:8
TTCTTAAAAATACCATAAAAGGCTTAAACTTGCCATTTACGGATTAATTCCAACTCTTTTCGGCTATCTTCATCTCTCTTTTAAAGTTAATGACTCAACCGG
+
FFFHBFHHIIIIIIIFHHHCGEFGHHIHHIIHDD/?DGGHH@DEB,5EGHGHHIIHIF?FGGHHCNBDFGHDGFDGGFDFH?DFHDFHDFHFFDFHH

Read 2 pair fastq file

@M00861:1:000000000-A36BE:1:1101:14650:1529 2:N:0:8
ACTAAAAATCAATTTTATCAATTTCAAGCTCTACTATTATATTACTCAATTATTTATGATGATGCCACTTTAAATAAAATATTGATGATACATATTTGGCAATAGCGG
+
BFFHIHHHHFHHGDHHIIHHHHGGHHHHFHDFHIFIHIIHIHDFHHIIHHHIIH=AAHHIIIIHGFHPHHHHHHGGHHIIHIIHGFHDFGGHHHDGHHH/CGHFFHHHHH
Trimming PE Short Sequence Reads

File 1 from sequencer

100 bases

 QC trim

100 bases

File 2 from sequencer

100 bases

 QC trim

50 bases

minimum read length = 40

Resulting FASTQ Files with trimmed reads

Paired end 1 trimmed file

Paired end 2 trimmed file
Trimming PE Short Sequence Reads

Filter 1 from sequencer
Filter 2 from sequencer

100 bases
100 bases

QC trim
QC trim

20 bases

minimum read length = 40

Resulting FASTQ Files with trimmed reads

Paired end 1 trimmed file
Paired end 2 trimmed file
Single end reads
Merge Overlapping Paired End Short Read Reads

**fragment 1**
- dsDNA

**fragment 2**
- dsDNA
Merge Overlapping Paired End Short Read Reads

**fragment 1**
- dsDNA

**fragment 2**
- dsDNA

Paired end read 1 (left)
- Green arrow

Paired end read 2 (right)
- Blue arrow
Merge Overlapping Paired End Short Read

fragment 1
- dsDNA

fragment 2
- dsDNA

Paired end read 1 (left)  Paired end read 2 (right)  Unpaired ‘merged’ read

Tools for merging overlapping reads:
- module spider FLASH
- module spider Coperead
- module spider PEAR
Trimmomatic Exercise using GCATemplates on Ada

Genomic Computational Analysis Templates

- Select #4 then find the template that contains trimmomatic
- Save the template script to your pwd
- Review the template script contents
- submit the template script to the scheduler
- Review the output files

For practice, we will copy a template file

gcatemplates
Mapping Reads to a Reference Assembly
Mapping Reads to a Reference Assembly

- Align reads using bwa
  - `module spider BWA`
    - bwa index files for UCSC genomes found here
      - `/scratch/datasets/genome_indexes/ucsc/mm10/bwa_0.7.12_index/`

- Align reads using bowtie or bowtie2
  - `module spider Bowtie`
    - Bowtie index files for UCSC genomes found here:
      - `/scratch/datasets/genome_indexes/ucsc/mm10/bowtie_index/`
  - `module spider Bowtie2`
    - Bowtie2 index files for UCSC found here:
      - `/scratch/datasets/genome_indexes/ucsc/mm10/bowtie2_index/`
Visualize bam File Alignments
Sample bam and reference files

```
cd $SCRATCH/ngs_class
```

For this samtools demo, add symbolic links* to the example files in your working directory

```
ln -s /scratch/training/intro_to_ngs/alignments/dr34.sam
```

Add a symbolic link to the example reference genome fasta file

```
ln -s /scratch/training/intro_to_ngs/genomes/c_dubliniensis.fa
```

Use the tab key when typing these long paths

* The symbolic links are used to make the commands shorter for demonstration purposes only. You do not need to make symbolic links in order to use `samtools tview`
Sorting Alignment sam/bam Files

- Sequence Alignment/Map format (sam)
  - view sam files using the UNIX command: `more dr34.sam`

- Binary Alignment/Map format (bam)
  - Compressed (binary) sam files need samtools to view
    - `module load SAMtools/1.8-GCCcore-6.3.0`
  - Recommended: sort sam/bam file based on coordinate into bam format
    - `samtools sort -@ 1 -m 2G -o dr34.bam dr34.sam`
  - Create an index of the bam file using samtools
    - A samtools index is needed prior to viewing bam files in browsers
      - `samtools index dr34.bam`
      - `dr34.bam.bai`
Viewing sam/bam Files

Viewing bam files using samtools

```
samtools view dr34.bam | more
```
view only alignments

```
samtools view -H dr34.bam
```
view only header

```
samtools view -h dr34.bam | more
```
view header + alignments
Alignment Statistics

```
samtools flagstat dr34.bam
```

150000 + 0 in total (QC-passed reads + QC-failed reads)
0 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
140150 + 0 mapped (93.43% : N/A)
150000 + 0 paired in sequencing
75002 + 0 read1
74998 + 0 read2
85639 + 0 properly paired (57.09% : N/A)
136854 + 0 with itself and mate mapped
3296 + 0 singletons (2.20% : N/A)
909 + 0 with mate mapped to a different chr
56 + 0 with mate mapped to a different chr (mapQ>=5)

Both reads in the pair are mapped on the same chromosome and in FR or RF orientation
Sam Flags and Bits

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

**Decoding SAM flags**

This utility makes it easy to identify what are the properties of a read based on its SAM flag value, or conversely, to find what the SAM Flag value would be for a given combination of properties.

To decode a given SAM flag value, just enter the number in the field below. The encoded properties will be listed under Summary below, to the right.

**SAM Flag** 99

**Switch to mate** Toggle first in pair/second in pair

**Find SAM flag by property:**

- 1 (read paired)
- 2 (read mapped in proper pair)
- 4 (read unmapped)
- 8 (mate unmapped)
- 16 (read reverse strand)
- 32 (mate reverse strand)
- 64 (first in pair)
- 128 (second in pair)
- 256 (not primary alignment)
- 512 (read fails platform/vendor quality checks)
- 1024 (read is PCR or optical duplicate)
- 2048 (supplementary alignment)

**Summary:**

- read paired
- read mapped in proper pair
- mate reverse strand
- first in pair

**SAM Flag is the sum of Bits**

99 = 64 + 32 + 2 + 1
Sam Flags and Bits

- Flags describe alignments (the flag value is the sum of bits)

![Diagram showing sam format with read id, flag, chromosome, genome coordinate and samtools view command examples]

- Filter bam alignments based on bit in flag (-f and/or -F)
  - Keep only reads that are 'mapped in proper pair'
    ```bash
    samtools view -h -b -f 2 dr34.bam > dr34_paired_reads.bam
    ```
  - Keep all except reads that are 'PCR or optical duplicate'
    ```bash
    samtools view -h -b -F 1024 dr34.bam > dr34_dedup_reads.bam
    ```
SAMtools without a Reference Genome

Reference genome represented on top as NNNNNNNNNN

```
    samtools tview dr34.bam
```
SAMtools with a Reference Genome

Reference genome sequence displayed on top

```
samtools tview dr34.bam c_dubliniensis.fa
```
SAMtools with a Reference Genome

Type ? for help menu

```
samtools tview dr34.bam c_dubliniensis.fa
```

---

`?` - Help menu

- Arrows: Small scroll movement
- h,j,k,l: Small scroll movement
- H,J,K,L: Large scroll movement
- ctrl-H: Scroll 1k left
- ctrl-L: Scroll 1k right
- space: Scroll one screen
- backspace: Scroll back one screen
- g: Go to specific location
- m: Color for mapping qual
- n: Color for nucleotide
- b: Color for base quality
- c: Color for cs color
- z: Color for cs qual
- .: Toggle on/off dot view
- s: Toggle on/off ref skip
- r: Toggle on/off rd name
- N: Turn on nt view
- C: Turn on cs view
- i: Toggle on/off ins
- v: Inverse video
- g: Exit

Underline: Secondary or orphan
Blue: 0-9  Green: 10-19
Yellow: 20-29  White: >=30
View at a Specific Coordinate

```bash
samtools tview dr34.bam c_dubliniensis.fa -p 1:315398
```
Sequence Error Correction
In Short Reads
Sequencing Errors in Short Reads

Tool for correcting sequencing errors: module spider Lighter
Digital Normalization
Digital Normalization

Reduce memory requirements by reducing the number of redundant sequence reads if you have a very high sequencing coverage (> 200x)

module spider BBMap

Use the bbnorm.sh script in the BBMap module

A Reference-Free Algorithm for Computational Normalization of Shotgun Sequencing Data
C. Titus Brown1,2,*, Adina Howe2, Qingpeng Zhang1, Alexis B. Pyrkosz3, Timothy H. Brom1
1 Computer Science and Engineering, Michigan State University, East Lansing, MI, USA
2 Microbiology and Molecular Genetics, Michigan State University, East Lansing, MI, USA
3 USDA Avian Disease and Oncology Laboratory, East Lansing, MI, USA
* E-mail: ctb@msu.edu
Sequence Variant Calling
Sequence Variant Calling

- Start with aligning reads to a reference
  - GATK does not require QC trimming
  - Mark PCR duplicates with Picard
- Differentiate between sequencing errors and SNPs
  - Calling SNPs may require a min read depth of 10x (higher for indels)
  - Calling variants may require 1/3 of reads to contain SNP
  - Strand bias may result as a consequence of the sequencing chemistry's response to certain DNA sequence motifs but it can be detected computationally
- BLAST reads with SNPs to identify variant calls due to misalignments especially with duplicated genes
- Variant Call Format (vcf) – standard format of variant calls
- Identify multiple-nucleotide polymorphism (MNP)
  - Two SNPs within a single codon
  - When might MNPs not be accurate?

<table>
<thead>
<tr>
<th>codon</th>
<th>translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference: TTT</td>
<td>Phe</td>
</tr>
<tr>
<td>SNP 1: TTA</td>
<td>Leu</td>
</tr>
<tr>
<td>SNP 2: TAT</td>
<td>Tyr</td>
</tr>
<tr>
<td>SNP 1 + 2: TAA</td>
<td>STOP</td>
</tr>
</tbody>
</table>
Marking PCR Duplicates

- PCR duplicates are artifacts resulting from a PCR amplification step during NGS library preparations.
- PCR duplicates should be removed/marked as to not bias the frequency of variants or gene expression levels
  - Use picard tools to mark duplicates
  - Freebayes will ignore marked duplicates during variant calling

```
module spider picard
```
Variant Calling Tools

Use bam file of sequence reads aligned to a reference as input for the following four work flows

1. **GATK**
   - module spider GATK picard SAMtools
   - No need to QC trim reads, the GATK best practices pipeline will perform the necessary steps including marking PCR duplicates
   - You need a set of known variants for your species (dbSNP) or you can bootstrap your population to get variant frequency
   - Used in conjunction with other tools
     - samtools
     - picard

2. **SAMtools and BCFtools**
   - module spider SAMtools BCFtools

3. **VarScan**
   - module spider VarScan

4. **FreeBayes**
   - module spider FreeBayes
##fileformat=VCFv4.0
##fileDate=20110705
##reference=1000GenomesPilot-NCBI37
##phasing=partial
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of Samples With Data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number=.,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership, build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FILTER=<ID=q10,Description="Quality below 10">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">

<table>
<thead>
<tr>
<th>CHROM</th>
<th>POS</th>
<th>ID</th>
<th>REF</th>
<th>ALT</th>
<th>QUAL</th>
<th>FILTER</th>
<th>INFO</th>
<th>FORMAT</th>
<th>Sample1</th>
<th>Sample2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4370</td>
<td>rs6057</td>
<td>G</td>
<td>A</td>
<td>29</td>
<td>.</td>
<td>NS=2;DP=13;AF=0.5;DB;H2</td>
<td>GT:GQ:DP:HQ</td>
<td>0/0:48:1:52,51</td>
<td>1/0:48:8:51,51</td>
</tr>
<tr>
<td>2</td>
<td>7330</td>
<td>.</td>
<td>T</td>
<td>A</td>
<td>q10</td>
<td>.</td>
<td>NS=5;DP=12;AF=0.017</td>
<td>GT:GQ:DP:HQ</td>
<td>0/0:46:3:58,50</td>
<td>0/1:3:5:65,3</td>
</tr>
<tr>
<td>2</td>
<td>110696</td>
<td>rs6055</td>
<td>A</td>
<td>G,T</td>
<td>67</td>
<td>PASS</td>
<td>NS=2;DP=10;AF=0.333,0.667;AA=T;DB</td>
<td>GT:GQ:DP:HQ</td>
<td>1/2:21:6:23,27</td>
<td>2/1:2:0:18,2</td>
</tr>
<tr>
<td>2</td>
<td>130237</td>
<td>.</td>
<td>T</td>
<td>.</td>
<td>47</td>
<td>.</td>
<td>NS=2;DP=16;AA=T</td>
<td>GT:GQ:DP:HQ</td>
<td>0/0:54:7:56,60</td>
<td>0/0:48:4:56,51</td>
</tr>
<tr>
<td>2</td>
<td>134567</td>
<td>microsat1</td>
<td>GTCT</td>
<td>G,GTACT</td>
<td>50</td>
<td>PASS</td>
<td>NS=2;DP=9;AA=G</td>
<td>GT:GQ:DP</td>
<td>0/1:35:4</td>
<td>0/2:17:2</td>
</tr>
</tbody>
</table>

3 more columns not shown due to width of rows
vcf File Column Descriptions

<table>
<thead>
<tr>
<th>##fileformat=VCFv4.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>##fileDate=20110705</td>
</tr>
<tr>
<td>##reference=1000GenomesPilot-NCBI37</td>
</tr>
<tr>
<td>##phasing=partial</td>
</tr>
<tr>
<td>##INFO=&lt;ID=NS,Number=1,Type=Integer,Description=&quot;Number of Samples With Data&quot;&gt;</td>
</tr>
<tr>
<td>##INFO=&lt;ID=DP,Number=1,Type=Integer,Description=&quot;Total Depth&quot;&gt;</td>
</tr>
<tr>
<td>##INFO=&lt;ID=AF,Number=.,Type=Float,Description=&quot;Allele Frequency&quot;&gt;</td>
</tr>
<tr>
<td>##INFO=&lt;ID=AA,Number=1,Type=String,Description=&quot;Ancestral Allele&quot;&gt;</td>
</tr>
<tr>
<td>##INFO=&lt;ID=DB,Number=0,Type=Flag,Description=&quot;dbSNP membership, build 129&quot;&gt;</td>
</tr>
<tr>
<td>##INFO=&lt;ID=H2,Number=0,Type=Flag,Description=&quot;HapMap2 membership&quot;&gt;</td>
</tr>
<tr>
<td>##FILTER=&lt;ID=q10,Description=&quot;Quality below 10&quot;&gt;</td>
</tr>
<tr>
<td>##FILTER=&lt;ID=s50,Description=&quot;Less than 50% of samples have data&quot;&gt;</td>
</tr>
<tr>
<td>##FORMAT=&lt;ID=GQ,Number=1,Type=Integer,Description=&quot;Genotype Quality&quot;&gt;</td>
</tr>
<tr>
<td>##FORMAT=&lt;ID=GT,Number=1,Type=String,Description=&quot;Genotype&quot;&gt;</td>
</tr>
<tr>
<td>##FORMAT=&lt;ID=DP,Number=1,Type=Integer,Description=&quot;Read Depth&quot;&gt;</td>
</tr>
<tr>
<td>##FORMAT=&lt;ID=HQ,Number=2,Type=Integer,Description=&quot;Haplotype Quality&quot;&gt;</td>
</tr>
</tbody>
</table>

#CHROM POS    ID        REF  ALT     QUAL FILTER INFO                              FORMAT       Sample1        Sample2
2      4370   rs6057    G    A       29   . NS=2;DP=13;AF=0.5;DB;H2 GT:GQ:DP:HQ  0|0:48:1:52,51 1|0:48:8:51,51
2      7330   .         T    A       3    q10 NS=5;DP=12;AF=0.017 GT:GQ:DP:HQ  0|0:46:3:58,50 0|1:3:5:65,3
2      110696 rs6055    A    G,T     67   PASS NS=2;DP=10;AF=0.333,0.667;AA=T;DB GT:GQ:DP:HQ  1|2:21:6:23,27 2|1:2:0:18,2
2      130237 .         T    .       47   . NS=2;DP=16;AA=T GT:GQ:DP:HQ  0|0:54:7:56,60 0|0:48:4:56,51
2      134567 microsat1 GTCT G,GTACT 50   PASS NS=2;DP=9;AA=G GT:GQ:DP  0/1:48:4 0/2:17:2

variants that are phased are inherited together (paternal)
| indicates phased variants
/ indicates non-phased variants

Sample1 haplotypes: GTGT and GTTT
Sample2 haplotypes: ATTT and GAGT

https://www.broadinstitute.org/gatk/guide/tagged?tag=phasing
Summarizing Variant Calls from Different Tools

The mean percentage with standard deviation of confidence variant calls with equal to or higher than the quality score threshold of 20 are represented for (A) Illumina data sets

Huang et al 2015 doi:10.1038/srep17875
Consequence of Amino Acid Change

- Assess consequence of amino acid change based on sequence conservation across multiple species using the PROVEAN tool.
- Variants with a score equal to or below -2.5 are considered “deleterious”.

Verifying that enough supporting sequences were found.

```
## PROVEAN v1.1 output ##
# Query sequence file:  CTRG_00013.fa
# Variation file:   CTRG_00013.var
# Protein database: /scratch/datasets/blast/nr
[16:01:13] searching related sequences...
[16:16:36] clustering subject sequences...
# Number of clusters:   30
# Number of supporting sequences used:  245
[16:18:39] computing delta alignment scores...
## PROVEAN scores ##
# VARIATION SCORE
A431S   -0.455
E411K   -3.051
E226Q   -1.564
```

"deleterious"
Annotate Variants

- A file of variant calls in vcf format is needed
- A reference sequence with gene annotations is needed
- snpEff annotates a vcf file
  - There are > 2,500 pre-built databases available and you can build your own if needed
  - Annotates MNP (multiple nucleotide polymorphism)
- Codon change due to two SNPs: ACA → GGA

```bash
module spider snpEff
```

```
5  325795 .  AC  GG  23.8901 .
AB=0.428571;ABP=3.32051;AC=1;AF=0.5;AN=2;AO=3;CIGAR=2X;DP=7;DPB=7;DPRA=0;EPP=3.73412;
EPPR=3.0103;GTI=0;LEN=2;MEANALT=1;MQM=33;MQMR=48.5;NS=1;NUMALT=1;ODDS=5.49681;PAIRED=0;
PAIREDR=0.5;PAO=0;PQA=0;PQR=0;PRO=0;QA=114;QR=150;RO=4;RPL=3;RPP=9.52472;RPPR=3.0103;
SAFE=2;SAP=3.73412;SAR=1;SRF=2;SRP=3.0103;SRR=2;TYPE=mnp;technology.ILLUMINA=1;
ANN=GG|missense_variant|MODERATE|CD36_51230|CD36_51230|transcript|CAX41505.1|
protein_coding|1/1|c.1657_1658delACinsGG|p.Thr553Gly|1657/1851|1657/1851|553/616|
```
Viewing SNPs in a Diploid Organism

Strand:
Blue = +
Red = -

heterozygous SNP
homozygous SNP
Example of Sequencing Strand Bias

Can identify/estimate strand bias using values in vcf file

Strand bias counts:
SRF, SRR, SAF, SAR

Bias estimate:
SAP

IonTorrent Proton

Strand:
Blue = +
Red = -
RNA-seq Overview
RNA-seq Applications

● Differential Expression (DE) and transcript abundance
  ○ HISAT2, Bowtie, TopHat, Cufflinks, Cuffmerge, Cuffdiff
  ○ DESeq and DESeq2 (R package)
  ○ EdgeR (R package)

● Transcriptome assembly (find isoforms and rare transcripts)
  ○ de novo (Trinity, Oases, SOAPdenovo-Trans)
  ○ reference based (Trinity, StringTie)

● Genome Annotation
  ○ Align to assembly for validation of gene models

● Variant Calling
  ○ STAR/Picard/GATK (Haplotype Caller (HC) in RNA-seq mode)

● de novo genome assembly scaffolding
  ○ L_RNA_scaffolder

● Identify fusion transcripts
  ○ tophat-fusion
Sequence Depth for RNA-seq Differential Expression

- Using more biological replicates instead of increasing sequencing depth resulted in improved accuracy of expression estimation.
- Use more biological replicates at lower sequencing depth is more beneficial than fewer samples at a higher sequencing depth.
- Increasing sequence depth is beneficial for exon or transcript-specific expression studies.

RNA-seq differential expression studies: more sequence or more replication?
doi: [10.1093/bioinformatics/btt688](https://doi.org/10.1093/bioinformatics/btt688) PMCID: PMC3904521
RNA-seq Transcriptome Assembly

- Assembly with a reference genome
  
  - module spider Trinity
  
  - module spider HISAT2 Cufflinks
  
  - module spider Scripture
  
  - module spider StringTie

- de novo assembly without a reference genome
  
  - module spider Trinity
  
  - module spider Oases
Digital Normalization for Transcriptome Assembly

- Reduce memory requirements by reducing the number of redundant sequence reads if you have a very high sequencing coverage (> 200x)
- Trinity 2.4.0+ automatically normalizes reads to a depth of 50
- The `bbnorm.sh` script in BBMap can normalize reads

```
module spider BBMap
```
Trinity – How it works:

RNA-Seq reads → Linear contigs → de-Bruijn graphs → Transcripts + Isoforms

ideally one graph per gene/transcript

Thousands of disjoint graphs

Running Trinity on Ada

- Trinity uses 100,000s of intermediate files
  - Contact help@hprc.tamu.edu and request a file quota increase before running Trinity or use the $TMPDIR in your job script
  - Run one Trinity job at a time and check resource usage
    - `showquota`
    - It is recommended not to run multiple Trinity jobs unless you are using $TMPDIR
  - Trinity creates checkpoints and can be restarted if it stops due to file/disk quota met, out of memory or runtime
    - Checkpoints are not available when running Trinity in Galaxy
    - Checkpoints are not available if you use $TMPDIR with Trinity
      - need to rsync results from $TMPDIR at end of job script
      - checkpoints are stored in $TMPDIR which is deleted after job ends
- See GCATemplates for sample Trinity scripts
ChIP-seq
Chromatin immunoprecipitation (ChIP) is a technique for identifying and characterizing elements in protein-DNA interactions involved in gene regulation or chromatin organization.

Chromatin immunoprecipitation sequencing (ChIP-Seq) on the SOLiD™ system
Chromatin immunoprecipitation sequencing (ChIP-Seq) on the SOLiD™ system. Nature Methods 6, (2009)
The goal is to find a consensus DNA sequence among the sequences at each peak which will give us the DNA sequence motif that a protein recognizes and binds.

A sequence logo can be used to represent the DNA sequence motif where the protein binds.

Generate a sequence logo with the R package seqLogo.

```
module load R_tamu/3.3.1-intel-2015B-default-mt
```
ChIP-seq Tools

- Protein-DNA interactions
  - module spider MACS
  - module spider MACS2

- Subdivision of ChIP-seq regions into discrete signal peaks
  - module spider PeakSplitter

- Peak caller
  - module spider PeakRanger
  - module spider BroadPeak

- Identify enriched domains from histone modification ChIP-seq data
  - module spider SICER
HPRC Resources

- Free Help
  - Send an email to help@hprc.tamu.edu if you have any questions regarding Bioinformatics tools usage on HPRC clusters
  - First spend some time investigating the error
    - read log files, stdout file, stderr file, tool manual
    - Google search
    - Google user groups: many are tool specific
- Include details about your issue
  - Which cluster or which Galaxy you are using
  - Which tool you are using
  - Which modules you have loaded
  - Commands you used in your job script
  - Error messages you are seeing
- HPRC NGS data analysis tools Documentation
  - https://hprc.tamu.edu/wiki/Bioinformatics
Thank you

Any questions?