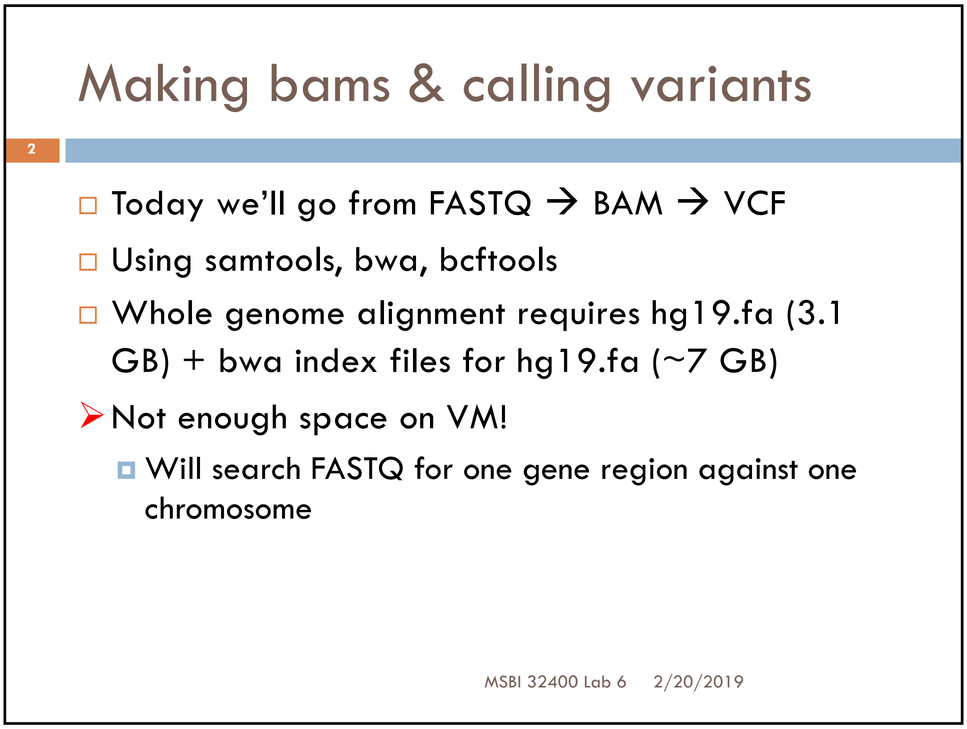
MSBI 32400 – LAB 6
LARRY HELSETH, PHD AND
JASON EDELSTEIN

February 20, 2019



Making bams & calling variants

2

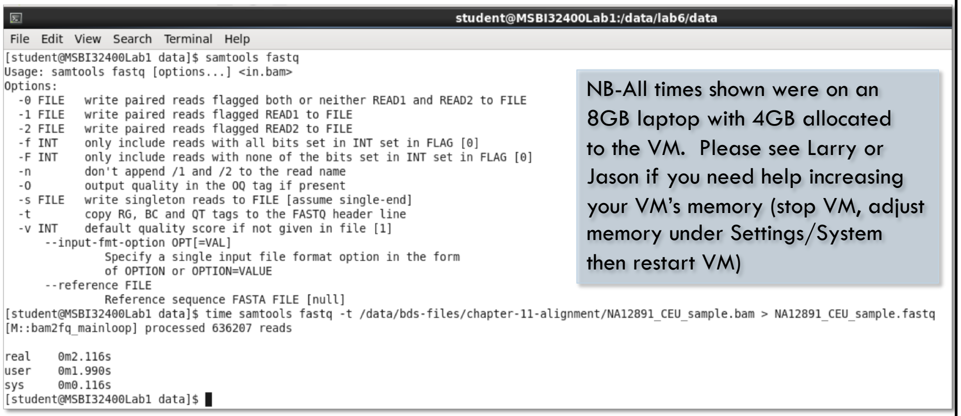
- □ Today we'll go from FASTQ → BAM → VCF
- □ Using samtools, bwa, bcftools
- □ Whole genome alignment requires hg19.fa (3.1 GB) + bwa index files for hg19.fa (~7 GB)
- ➢ Not enough space on VM!
  - ▫ Will search FASTQ for one gene region against one chromosome

MSBI 32400 Lab 6 2/20/2019

Setup Lab6 folders then extract FASTQ

- □ Make /data/lab6/bin, /data/lab6/data, /data/lab6/doc, /data/lab6/results & /data/lab6/src
- □ Go to /data/lab6/data
- □ Run samtools fastq to extract reads from Vince Buffalo's sample BAM in /data/bds-files/chapter-11-alignment/NA12891_CEU_sample.bam

Syntax


student@MSBI32400Lab1:/data/lab6/data
File Edit View Search Terminal Help

```
[student@MSBI32400Lab1 data]$ samtools fastq
Usage: samtools fastq [options...] <in.bam>
Options:
  -0 FILE   write paired reads flagged both or neither READ1 and READ2 to FILE
  -1 FILE   write paired reads flagged READ1 to FILE
  -2 FILE   write paired reads flagged READ2 to FILE
  -f INT    only include reads with all bits set in INT set in FLAG [0]
  -F INT    only include reads with none of the bits set in INT set in FLAG [0]
  -n        don't append /1 and /2 to the read name
  -O        output quality in the OQ tag if present
  -s FILE   write singleton reads to FILE [assume single-end]
  -t        copy RG, BC and QT tags to the FASTQ header line
  -v INT    default quality score if not given in file [1]
      --input-fmt-option OPT[=VAL]
               Specify a single input file format option in the form
               of OPTION or OPTION=VALUE
      --reference FILE
               Reference sequence FASTA FILE [null]
[student@MSBI32400Lab1 data]$ time samtools fastq -t /data/bds-files/chapter-11-alignment/NA12891_CEU_sample.bam > NA12891_CEU_sample.fastq
[M::bam2fq_mainloop] processed 636207 reads

real    0m2.116s
user    0m1.990s
sys     0m0.116s
[student@MSBI32400Lab1 data]$ 
```

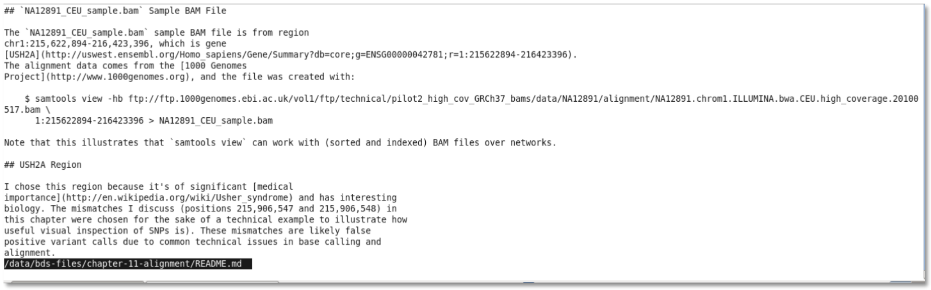
NB-All times shown were on an 8GB laptop with 4GB allocated to the VM. Please see Larry or Jason if you need help increasing your VM's memory (stop VM, adjust memory under Settings/System then restart VM)

samtools fastq -t (path to BAM) > NA12891_CEU_sample.fastq

Download a reference genome

5

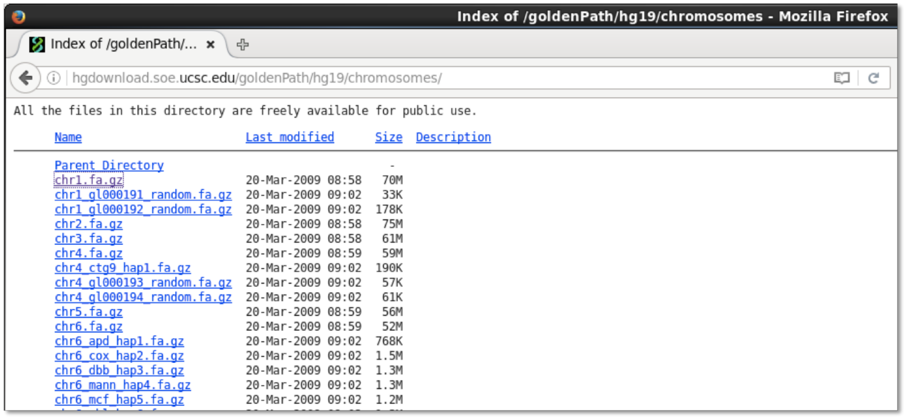
- □ Buffalo’s chapter 11 README.md shows USH2A gene coordinates from chromosome 1:



```
## `NA12891_CEU_sample.bam` Sample BAM File

The `NA12891_CEU_sample.bam` sample BAM file is from region
chr1:215,622,894-216,423,396, which is gene
[USH2A](http://uswest.ensembl.org/Homo_sapiens/Gene/Summary?db=core;g=ENSG00000042781;r=1:215622894-216423396).
The alignment data comes from the [1000 Genomes
Project](http://www.1000genomes.org), and the file was created with:

    $ samtools view -hb ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/pilot2_high_cov_GRCh37_bams/data/NA12891/alignment/NA12891.chrom1.ILLUMINA.bwa.CEU.high_coverage.20100
517.bam \
      1:215622894-216423396 > NA12891_CEU_sample.bam

Note that this illustrates that `samtools view` can work with (sorted and indexed) BAM files over networks.

## USH2A Region

I chose this region because it's of significant [medical
importance](http://en.wikipedia.org/wiki/Usher_syndrome) and has interesting
biology. The mismatches I discuss (positions 215,906,547 and 215,906,548) in
this chapter were chosen for the sake of a technical example to illustrate how
useful visual inspection of SNPs is). These mismatches are likely false
positive variant calls due to common technical issues in base calling and
alignment.
/data/bds-files/chapter-11-alignment/README.md
```

Download chr1 from UCSC

6



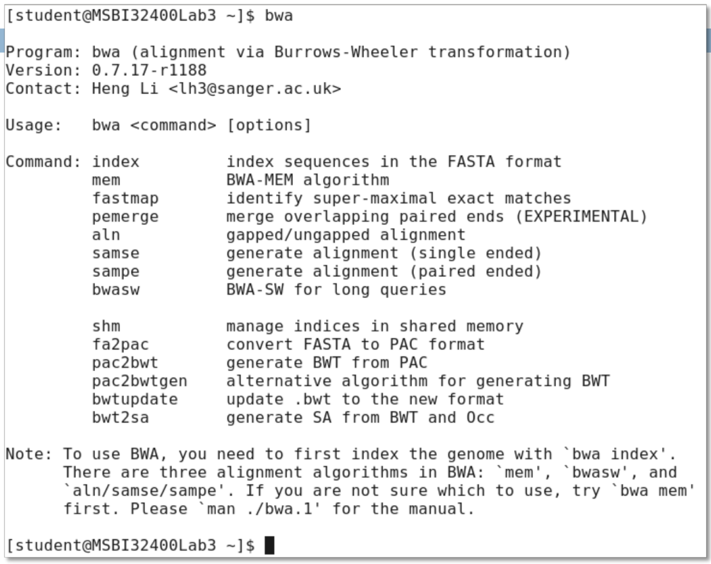
Index of /goldenPath/hg19/chromosomes - Mozilla Firefox
Index of /goldenPath/... ×
hgdownload.soe.ucsc.edu/goldenPath/hg19/chromosomes/

```
All the files in this directory are freely available for public use.

      Name                       Last modified      Size  Description

      Parent Directory                                -
      chr1.fa.gz                 20-Mar-2009 08:58   70M
      chr1_gl000191_random.fa.gz 20-Mar-2009 09:02   33K
      chr1_gl000192_random.fa.gz 20-Mar-2009 09:02  178K
      chr2.fa.gz                 20-Mar-2009 08:58   75M
      chr3.fa.gz                 20-Mar-2009 08:58   61M
      chr4.fa.gz                 20-Mar-2009 08:59   59M
      chr4_ctg9_hap1.fa.gz       20-Mar-2009 09:02  190K
      chr4_gl000193_random.fa.gz 20-Mar-2009 09:02   57K
      chr4_gl000194_random.fa.gz 20-Mar-2009 09:02   61K
      chr5.fa.gz                 20-Mar-2009 08:59   56M
      chr6.fa.gz                 20-Mar-2009 08:59   52M
      chr6_apd_hap1.fa.gz        20-Mar-2009 09:02  768K
      chr6_cox_hap2.fa.gz        20-Mar-2009 09:02  1.5M
      chr6_dbb_hap3.fa.gz        20-Mar-2009 09:02  1.3M
      chr6_mann_hap4.fa.gz       20-Mar-2009 09:02  1.3M
      chr6_mcf_hap5.fa.gz        20-Mar-2009 09:02  1.2M
```

Move chr1.fa.gz from your ~/Downloads to /data/lab6/data and extract using gunzip

BWA commands

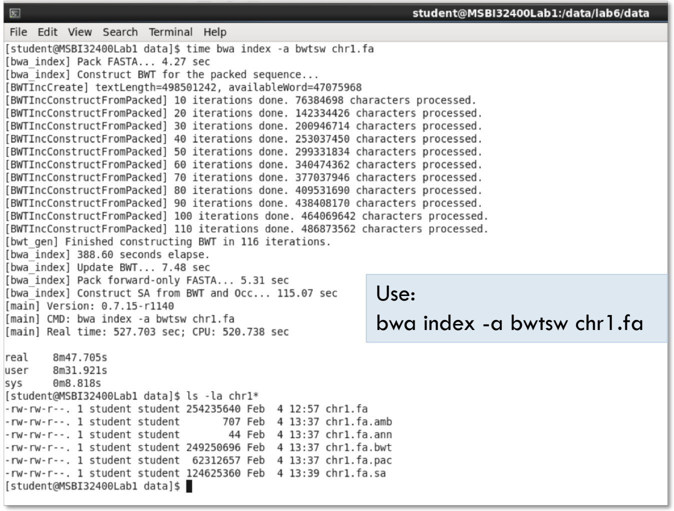
7



```
[student@MSBI32400Lab3 ~]$ bwa

Program: bwa (alignment via Burrows-Wheeler transformation)
Version: 0.7.17-r1188
Contact: Heng Li <lh3@sanger.ac.uk>

Usage:   bwa <command> [options]

Command: index         index sequences in the FASTA format
         mem           BWA-MEM algorithm
         fastmap       identify super-maximal exact matches
         pemerge       merge overlapping paired ends (EXPERIMENTAL)
         aln           gapped/ungapped alignment
         samse         generate alignment (single ended)
         sampe         generate alignment (paired ended)
         bwasw         BWA-SW for long queries

         shm           manage indices in shared memory
         fa2pac        convert FASTA to PAC format
         pac2bwt       generate BWT from PAC
         pac2bwtgen    alternative algorithm for generating BWT
         bwtupdate     update .bwt to the new format
         bwt2sa        generate SA from BWT and Occ

Note: To use BWA, you need to first index the genome with `bwa index'.
      There are three alignment algorithms in BWA: `mem', `bwasw', and
      `aln/samse/sampe'. If you are not sure which to use, try `bwa mem'
      first. Please `man ./bwa.1' for the manual.

[student@MSBI32400Lab3 ~]$ █
```

Need to index for bwa

8

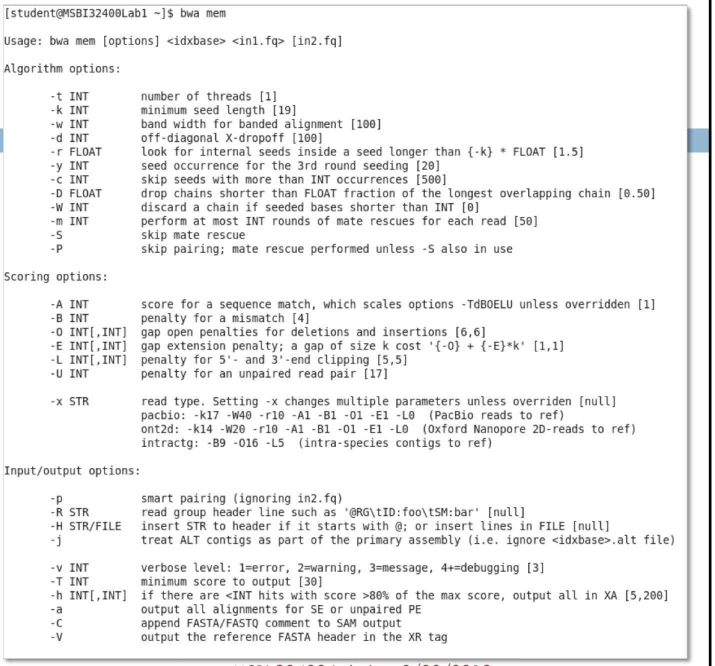

student@MSBI32400Lab1:/data/lab6/data
File Edit View Search Terminal Help

```
[student@MSBI32400Lab1 data]$ time bwa index -a bwtsw chr1.fa
[bwa_index] Pack FASTA... 4.27 sec
[bwa_index] Construct BWT for the packed sequence...
[BWTIncCreate] textLength=498501242, availableWord=47075968
[BWTIncConstructFromPacked] 10 iterations done. 76384698 characters processed.
[BWTIncConstructFromPacked] 20 iterations done. 142334426 characters processed.
[BWTIncConstructFromPacked] 30 iterations done. 200946714 characters processed.
[BWTIncConstructFromPacked] 40 iterations done. 253037450 characters processed.
[BWTIncConstructFromPacked] 50 iterations done. 299331834 characters processed.
[BWTIncConstructFromPacked] 60 iterations done. 340474362 characters processed.
[BWTIncConstructFromPacked] 70 iterations done. 377037946 characters processed.
[BWTIncConstructFromPacked] 80 iterations done. 409531690 characters processed.
[BWTIncConstructFromPacked] 90 iterations done. 438408170 characters processed.
[BWTIncConstructFromPacked] 100 iterations done. 464069642 characters processed.
[BWTIncConstructFromPacked] 110 iterations done. 486873562 characters processed.
[bwt_gen] Finished constructing BWT in 116 iterations.
[bwa_index] 388.60 seconds elapse.
[bwa_index] Update BWT... 7.48 sec
[bwa_index] Pack forward-only FASTA... 5.31 sec
[bwa_index] Construct SA from BWT and Occ... 115.07 sec
[main] Version: 0.7.15-r1140
[main] CMD: bwa index -a bwtsw chr1.fa
[main] Real time: 527.703 sec; CPU: 520.738 sec

real    8m47.705s
user    8m31.921s
sys     0m8.818s
[student@MSBI32400Lab1 data]$ ls -la chr1*
-rw-rw-r--. 1 student student 254235640 Feb  4 12:57 chr1.fa
-rw-rw-r--. 1 student student       707 Feb  4 13:37 chr1.fa.amb
-rw-rw-r--. 1 student student        44 Feb  4 13:37 chr1.fa.ann
-rw-rw-r--. 1 student student 249250696 Feb  4 13:37 chr1.fa.bwt
-rw-rw-r--. 1 student student  62312657 Feb  4 13:37 chr1.fa.pac
-rw-rw-r--. 1 student student 124625360 Feb  4 13:39 chr1.fa.sa
[student@MSBI32400Lab1 data]$ █
```

Use:

bwa index -a bwtsw chr1.fa

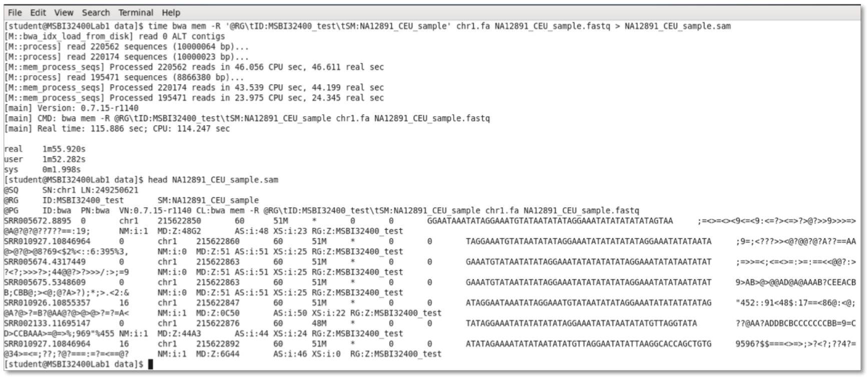
bwa mem syntax:



```
[student@MSBI32400Lab1 ~]$ bwa mem

Usage: bwa mem [options] <idxbase> <in1.fq> [in2.fq]

Algorithm options:

       -t INT        number of threads [1]
       -k INT        minimum seed length [19]
       -w INT        band width for banded alignment [100]
       -d INT        off-diagonal X-dropoff [100]
       -r FLOAT      look for internal seeds inside a seed longer than {-k} * FLOAT [1.5]
       -y INT        seed occurrence for the 3rd round seeding [20]
       -c INT        skip seeds with more than INT occurrences [500]
       -D FLOAT      drop chains shorter than FLOAT fraction of the longest overlapping chain [0.50]
       -W INT        discard a chain if seeded bases shorter than INT [0]
       -m INT        perform at most INT rounds of mate rescues for each read [50]
       -S            skip mate rescue
       -P            skip pairing; mate rescue performed unless -S also in use

Scoring options:

       -A INT        score for a sequence match, which scales options -TdBOELU unless overridden [1]
       -B INT        penalty for a mismatch [4]
       -O INT[,INT]  gap open penalties for deletions and insertions [6,6]
       -E INT[,INT]  gap extension penalty; a gap of size k cost '{-O} + {-E}*k' [1,1]
       -L INT[,INT]  penalty for 5'- and 3'-end clipping [5,5]
       -U INT        penalty for an unpaired read pair [17]

       -x STR        read type. Setting -x changes multiple parameters unless overriden [null]
                     pacbio: -k17 -W40 -r10 -A1 -B1 -O1 -E1 -L0  (PacBio reads to ref)
                     ont2d: -k14 -W20 -r10 -A1 -B1 -O1 -E1 -L0  (Oxford Nanopore 2D-reads to ref)
                     intractg: -B9 -O16 -L5  (intra-species contigs to ref)

Input/output options:

       -p            smart pairing (ignoring in2.fq)
       -R STR        read group header line such as '@RG\tID:foo\tSM:bar' [null]
       -H STR/FILE   insert STR to header if it starts with @; or insert lines in FILE [null]
       -j            treat ALT contigs as part of the primary assembly (i.e. ignore <idxbase>.alt file)

       -v INT        verbose level: 1=error, 2=warning, 3=message, 4+=debugging [3]
       -T INT        minimum score to output [30]
       -h INT[,INT]  if there are <INT hits with score >80% of the max score, output all in XA [5,200]
       -a            output all alignments for SE or unpaired PE
       -C            append FASTA/FASTQ comment to SAM output
       -V            output the reference FASTA header in the XR tag
```

Align and generate SAM file

- bwa mem -R '@RG\tID:**MSBI32400_test**\tSM:NA12891_CEU_sample' chr1.fa NA12891_CEU_sample.fastq > NA12891_CEU_sample.sam



```
[student@MSBI32400Lab1 data]$ time bwa mem -R '@RG\tID:MSBI32400_test\tSM:NA12891_CEU_sample' chr1.fa NA12891_CEU_sample.fastq > NA12891_CEU_sample.sam
[M::bwa_idx_load_from_disk] read 0 ALT contigs
[M::process] read 220562 sequences (10000064 bp)...
[M::process] read 220174 sequences (10000023 bp)...
[M::mem_process_seqs] Processed 220562 reads in 46.056 CPU sec, 46.611 real sec
[M::process] read 195471 sequences (8866380 bp)...
[M::mem_process_seqs] Processed 220174 reads in 43.539 CPU sec, 44.199 real sec
[M::mem_process_seqs] Processed 195471 reads in 23.975 CPU sec, 24.345 real sec
[main] Version: 0.7.15-r1140
[main] CMD: bwa mem -R @RG\tID:MSBI32400_test\tSM:NA12891_CEU_sample chr1.fa NA12891_CEU_sample.fastq
[main] Real time: 115.886 sec; CPU: 114.247 sec

real    1m55.920s
user    1m52.282s
sys     0m1.998s
[student@MSBI32400Lab1 data]$ head NA12891_CEU_sample.sam
@SQ     SN:chr1 LN:249250621
@RG     ID:MSBI32400_test       SM:NA12891_CEU_sample
@PG     ID:bwa  PN:bwa  VN:0.7.15-r1140 CL:bwa mem -R @RG\tID:MSBI32400_test\tSM:NA12891_CEU_sample chr1.fa NA12891_CEU_sample.fastq
SRR005672.8895  0       chr1    215622850       60      51M     *       0       0       GGAATAAATATAGGAAATGTATAATATATAGGAAATATATATATATAGTAA     ;=<>=<><9<=<9:<=?><=>?>@?>>9>>>=>
@A@?@?@??7??==:19;      NM:i:1  MD:Z:48G2       AS:i:48 XS:i:23 RG:Z:MSBI32400_test
SRR010927.10846964      0       chr1    215622860       60      51M     *       0       0       TAGGAAATGTATAATATATAGGAAATATATATATATAGGAAATATATAATA     ;9=;<???>><@?@@?@?A??==AA
@>@?@>@8?69<$2%<::6:395%3,      NM:i:0  MD:Z:51 AS:i:51 XS:i:25 RG:Z:MSBI32400_test
SRR005674.4317449       0       chr1    215622863       60      51M     *       0       0       GAAATGTATAATATATAGGAAATATATATATATAGGAAATATATAATATAT     ;=>>=<;<=<>=:>=:==<<@@?:>
?<?;>>>?>;44@@?>?>>>/:>;=9      NM:i:0  MD:Z:51 AS:i:51 XS:i:25 RG:Z:MSBI32400_test
SRR005675.5348609       0       chr1    215622863       60      51M     *       0       0       GAAATGTATAATATATAGGAAATATATATATATAGGAAATATATAATATAT     9>AB>@>@@AD@A@AAAB?CEEACB
B;CBB@;><@;@?A>?);*;>.<2:&      NM:i:0  MD:Z:51 AS:i:51 XS:i:25 RG:Z:MSBI32400_test
SRR010926.10855357      16      chr1    215622847       60      51M     *       0       0       ATAGGAATAAATATAGGAAATGTATAATATATAGGAAATATATATATATAG     "452::91<48$:17==<86@:<@;
@A?@>?=B?@AA@?@>@>@>?=?=A<      NM:i:1  MD:Z:0C50       AS:i:50 XS:i:22 RG:Z:MSBI32400_test
SRR002133.11695147      0       chr1    215622876       60      48M     *       0       0       TATAGGAAATATATATATATAGGAAATATATAATATATGTTAGGTATA        ??@AA?ADDBCBCCCCCCBB=9=C
D>CCBAAA>=@=>%;969"%455 NM:i:1  MD:Z:44A3       AS:i:44 XS:i:24 RG:Z:MSBI32400_test
SRR010927.10846964      16      chr1    215622892       60      51M     *       0       0       ATATAGAAAATATATAATATATGTTAGGAATATATTAAGGCACCAGCTGTG     9596?$$===<>=>;>?<?;??4?=
@34>=<=;??;?@?===:=?=<==@?      NM:i:1  MD:Z:6G44       AS:i:46 XS:i:0  RG:Z:MSBI32400_test
[student@MSBI32400Lab1 data]$
```

Samtools

11

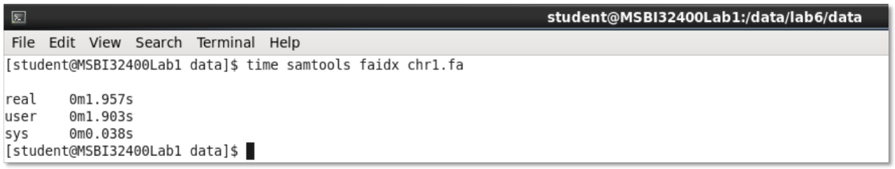
From man page:

- Samtools is a set of utilities that manipulate alignments in the BAM format. It imports from and exports to the SAM (Sequence Alignment/Map) format, does sorting, merging and indexing, and allows to retrieve reads in any regions swiftly.
- Samtools is designed to work on a stream. It regards an input file '-' as the standard input (stdin) and an output file '-' as the standard output (std-out). Several commands can thus be combined with Unix pipes. Samtools always output warning and error messages to the standard error output (stderr).
- Samtools is also able to open a BAM (not SAM) file on a remote FTP or HTTP server if the BAM file name starts with 'ftp://' or 'http://'. Samtools checks the current working directory for the index file and will download the index upon absence. Samtools does not retrieve the entire alignment file unless it is asked to do so.

Need samtools index of chr1.fa

12

- samtools faidx builds a .fai file


student@MSBI32400Lab1:/data/lab6/data
File Edit View Search Terminal Help

```
[student@MSBI32400Lab1 data]$ time samtools faidx chr1.fa

real    0m1.957s
user    0m1.903s
sys     0m0.038s
[student@MSBI32400Lab1 data]$ 
```

Convert SAM to BAM

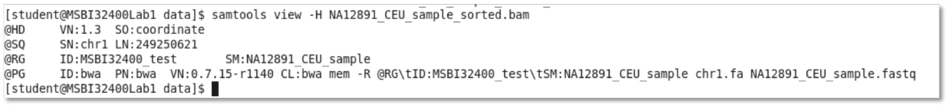
13

- □ samtools view -bt chr1.fa.fai NA12891_CEU_sample.sam > NA12891_CEU_sample.bam
- □ samtools sort -o NA12891_CEU_sample_sorted.bam NA12891_CEU_sample.bam
- □ samtools index NA12891_CEU_sample_sorted.bam

View header of new sorted BAM

14

- □ The -R '@RG' syntax put our new ID and sample name in header along with the @PG (program) info for how we generated the alignment

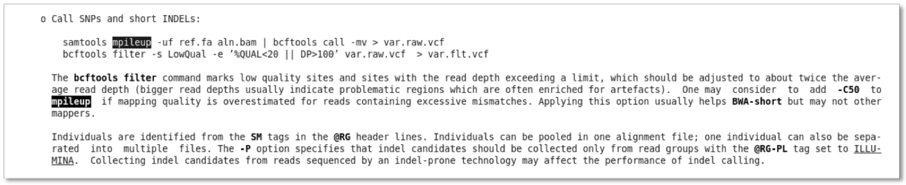


```
[student@MSBI32400Lab1 data]$ samtools view -H NA12891_CEU_sample_sorted.bam
@HD     VN:1.3  SO:coordinate
@SQ     SN:chr1 LN:249250621
@RG     ID:MSBI32400_test       SM:NA12891_CEU_sample
@PG     ID:bwa  PN:bwa  VN:0.7.15-r1140 CL:bwa mem -R @RG\tID:MSBI32400_test\tSM:NA12891_CEU_sample chr1.fa NA12891_CEU_sample.fastq
[student@MSBI32400Lab1 data]$ █
```

Check samtools man page

15

- □ man samtools then search for mpileup (use '/mpileup')



```
 o Call SNPs and short INDELs:

     samtools mpileup -uf ref.fa aln.bam | bcftools call -mv > var.raw.vcf
     bcftools filter -s LowQual -e '%QUAL<20 || DP>100' var.raw.vcf  > var.flt.vcf

   The bcftools filter command marks low quality sites and sites with the read depth exceeding a limit, which should be adjusted to about twice the aver-
   age read depth (bigger read depths usually indicate problematic regions which are often enriched for artefacts).  One may  consider  to  add  -C50  to
   mpileup  if mapping quality is overestimated for reads containing excessive mismatches. Applying this option usually helps BWA-short but may not other
   mappers.

   Individuals are identified from the SM tags in the @RG header lines. Individuals can be pooled in one alignment file; one individual can also be sepa-
   rated  into  multiple  files. The -P option specifies that indel candidates should be collected only from read groups with the @RG-PL tag set to ILLU-
   MINA.  Collecting indel candidates from reads sequenced by an indel-prone technology may affect the performance of indel calling.
```

- □ See also: http://proquestcombo.safaribooksonline.com.proxy.uchicago.edu/book/bioinformatics/9781449367480/visualizing-alignments-with-samtools-tview-and-the-integrated-genomics-viewer/idp33784528_html

MSBI 32400 Lab 6 2/20/2019

bcftools

16

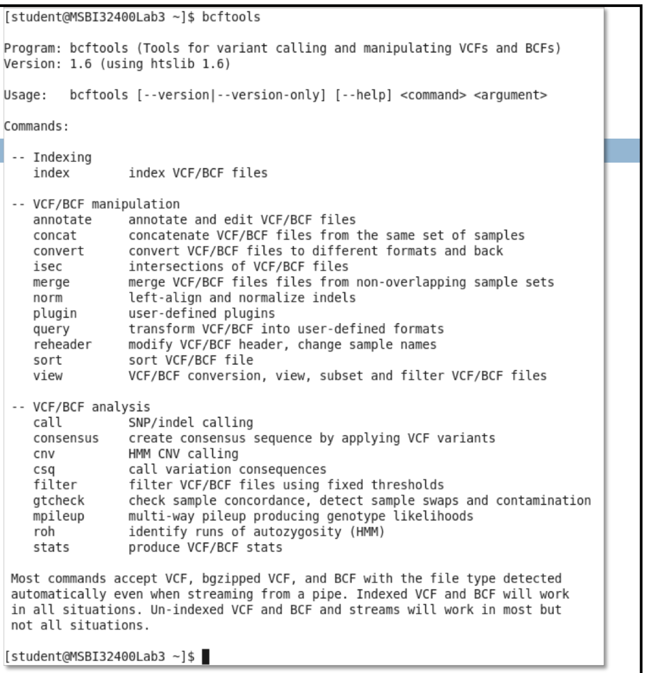
(***bcftools mentioned last week as a better way to convert 23andMe to VCF***)

From man page:

- □ BCFtools is a set of utilities that manipulate variant calls in the Variant Call Format (VCF) and its binary counterpart BCF. All commands work transparently with both VCFs and BCFs, both uncompressed and BGZF-compressed.
- □ Most commands accept VCF, bgzipped VCF and BCF with filetype detected automatically even when streaming from a pipe. Indexed VCF and BCF will work in all situations. Un-indexed VCF and BCF and streams will work in most, but not all situations. In general, whenever multiple VCFs are read simultaneously, they must be indexed and therefore also compressed.
- □ BCFtools is designed to work on a stream. It regards an input file "-" as the standard input (stdin) and outputs to the standard output (stdout). Several commands can thus be combined with Unix pipes.

MSBI 32400 Lab 6 2/20/2019

bcftools syntax:

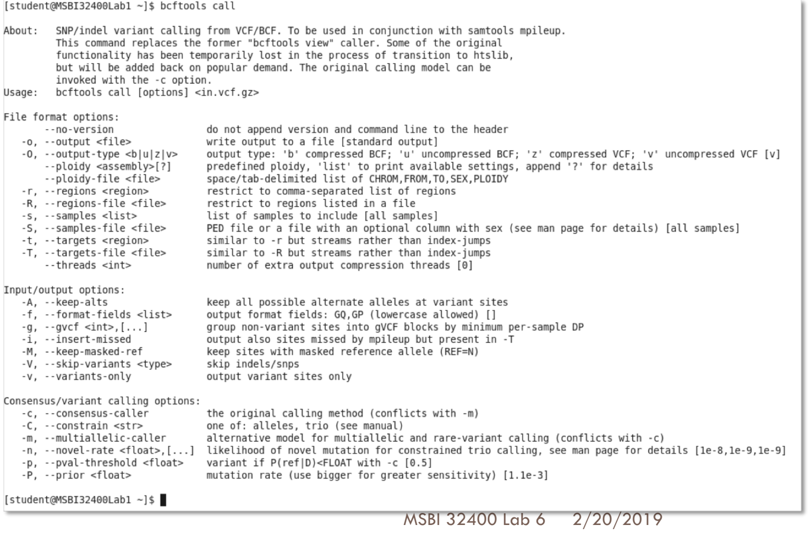


```
[student@MSBI32400Lab3 ~]$ bcftools

Program: bcftools (Tools for variant calling and manipulating VCFs and BCFs)
Version: 1.6 (using htslib 1.6)

Usage:   bcftools [--version|--version-only] [--help] <command> <argument>

Commands:

 -- Indexing
    index        index VCF/BCF files

 -- VCF/BCF manipulation
    annotate     annotate and edit VCF/BCF files
    concat       concatenate VCF/BCF files from the same set of samples
    convert      convert VCF/BCF files to different formats and back
    isec         intersections of VCF/BCF files
    merge        merge VCF/BCF files files from non-overlapping sample sets
    norm         left-align and normalize indels
    plugin       user-defined plugins
    query        transform VCF/BCF into user-defined formats
    reheader     modify VCF/BCF header, change sample names
    sort         sort VCF/BCF file
    view         VCF/BCF conversion, view, subset and filter VCF/BCF files

 -- VCF/BCF analysis
    call         SNP/indel calling
    consensus    create consensus sequence by applying VCF variants
    cnv          HMM CNV calling
    csq          call variation consequences
    filter       filter VCF/BCF files using fixed thresholds
    gtcheck      check sample concordance, detect sample swaps and contamination
    mpileup      multi-way pileup producing genotype likelihoods
    roh          identify runs of autozygosity (HMM)
    stats        produce VCF/BCF stats

 Most commands accept VCF, bgzipped VCF, and BCF with the file type detected
 automatically even when streaming from a pipe. Indexed VCF and BCF will work
 in all situations. Un-indexed VCF and BCF and streams will work in most but
 not all situations.

[student@MSBI32400Lab3 ~]$ 
```

Generate mpileup & run bcftools

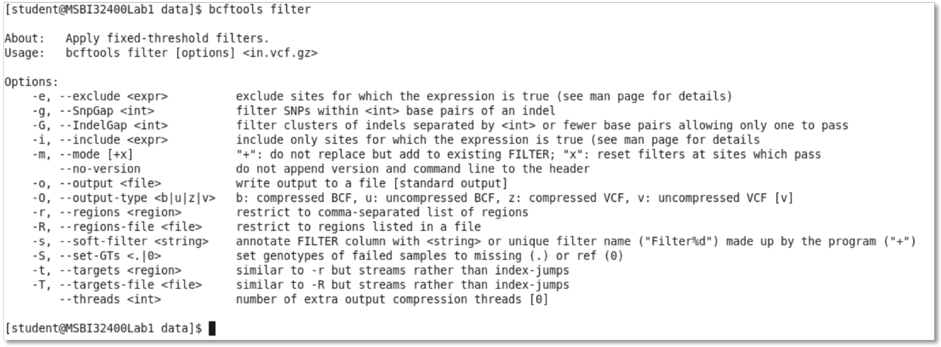
- samtools mpileup -uf chr1.fa NA12891_CEU_sample_sorted.bam | bcftools **call** -mv > NA12891_CEU_sample_sorted_var.raw.vcf
- bcftools **filter** -s LowQual -e '%QUAL<20' NA12891_CEU_sample_sorted_var.raw.vcf > NA12891_CEU_sample_sorted_var.flt.vcf
- How many variants are called in the final VCF? How many variants are called with "PASS"?
  - Include in your README for Jason

bcftools call syntax



```
[student@MSBI32400Lab1 ~]$ bcftools call

About:   SNP/indel variant calling from VCF/BCF. To be used in conjunction with samtools mpileup.
         This command replaces the former "bcftools view" caller. Some of the original
         functionality has been temporarily lost in the process of transition to htslib,
         but will be added back on popular demand. The original calling model can be
         invoked with the -c option.
Usage:   bcftools call [options] <in.vcf.gz>

File format options:
       --no-version                do not append version and command line to the header
   -o, --output <file>             write output to a file [standard output]
   -O, --output-type <b|u|z|v>     output type: 'b' compressed BCF; 'u' uncompressed BCF; 'z' compressed VCF; 'v' uncompressed VCF [v]
       --ploidy <assembly>[?]      predefined ploidy, 'list' to print available settings, append '?' for details
       --ploidy-file <file>        space/tab-delimited list of CHROM,FROM,TO,SEX,PLOIDY
   -r, --regions <region>          restrict to comma-separated list of regions
   -R, --regions-file <file>       restrict to regions listed in a file
   -s, --samples <list>            list of samples to include [all samples]
   -S, --samples-file <file>       PED file or a file with an optional column with sex (see man page for details) [all samples]
   -t, --targets <region>          similar to -r but streams rather than index-jumps
   -T, --targets-file <file>       similar to -R but streams rather than index-jumps
       --threads <int>             number of extra output compression threads [0]

Input/output options:
   -A, --keep-alts                 keep all possible alternate alleles at variant sites
   -f, --format-fields <list>      output format fields: GQ,GP (lowercase allowed) []
   -g, --gvcf <int>,[...]          group non-variant sites into gVCF blocks by minimum per-sample DP
   -i, --insert-missed             output also sites missed by mpileup but present in -T
   -M, --keep-masked-ref           keep sites with masked reference allele (REF=N)
   -V, --skip-variants <type>      skip indels/snps
   -v, --variants-only             output variant sites only

Consensus/variant calling options:
   -c, --consensus-caller          the original calling method (conflicts with -m)
   -C, --constrain <str>           one of: alleles, trio (see manual)
   -m, --multiallelic-caller       alternative model for multiallelic and rare-variant calling (conflicts with -c)
   -n, --novel-rate <float>,[...]  likelihood of novel mutation for constrained trio calling, see man page for details [1e-8,1e-9,1e-9]
   -p, --pval-threshold <float>    variant if P(ref|D)<FLOAT with -c [0.5]
   -P, --prior <float>             mutation rate (use bigger for greater sensitivity) [1.1e-3]

[student@MSBI32400Lab1 ~]$ █
```

bcftools filter syntax



```
[student@MSBI32400Lab1 data]$ bcftools filter

About:   Apply fixed-threshold filters.
Usage:   bcftools filter [options] <in.vcf.gz>

Options:
    -e, --exclude <expr>          exclude sites for which the expression is true (see man page for details)
    -g, --SnpGap <int>            filter SNPs within <int> base pairs of an indel
    -G, --IndelGap <int>          filter clusters of indels separated by <int> or fewer base pairs allowing only one to pass
    -i, --include <expr>          include only sites for which the expression is true (see man page for details
    -m, --mode [+x]               "+": do not replace but add to existing FILTER; "x": reset filters at sites which pass
        --no-version              do not append version and command line to the header
    -o, --output <file>           write output to a file [standard output]
    -O, --output-type <b|u|z|v>   b: compressed BCF, u: uncompressed BCF, z: compressed VCF, v: uncompressed VCF [v]
    -r, --regions <region>        restrict to comma-separated list of regions
    -R, --regions-file <file>     restrict to regions listed in a file
    -s, --soft-filter <string>    annotate FILTER column with <string> or unique filter name ("Filter%d") made up by the program ("+")
    -S, --set-GTs <.|0>           set genotypes of failed samples to missing (.) or ref (0)
    -t, --targets <region>        similar to -r but streams rather than index-jumps
    -T, --targets-file <file>     similar to -R but streams rather than index-jumps
        --threads <int>           number of extra output compression threads [0]

[student@MSBI32400Lab1 data]$ █
```

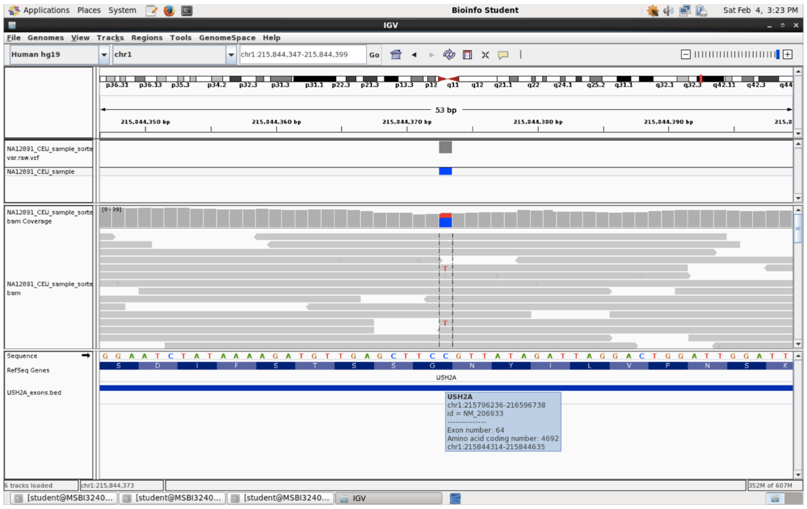
Open BAM & VCF in IGV

21

- View exon 64 and look for SNPs called in VCF
  - Most SNPs in introns, but a few in exons
  - Record the coordinates and Amino Acid # to send to Jason

IGV view (BAM + bai + VCF + BED)

22


Applications Places System
Bioinfo Student
Sat Feb 4, 3:23 PM
IGV
File Genomes View Tracks Regions Tools GenomeSpace Help
Human hg19
chr1
chr1:215,844,347-215,844,399
Go
p36.31 p36.13 p35.3 p34.2 p32.3 p31.3 p31.1 p22.3 p21.3 p13.3 p12 q11 q12 q21.1 q22 q24.1 q25.2 q31.1 q32.1 q32.3 q42.11 q42.3 q44
53 bp
215,844,350 bp
215,844,360 bp
215,844,370 bp
215,844,380 bp
215,844,390 bp
215,8
NA12891_CEU_sample_sorte var.raw.vcf
NA12891_CEU_sample
NA12891_CEU_sample_sorte bam Coverage
[0 - 39]
NA12891_CEU_sample_sorte bam
T
T
Sequence
G G A A T C T A T A A A A G A T G T T G A G C T T C C G T T A T A G A T T A G G A C T G G A T T G G A T T
RefSeq Genes
S D I F S T S S G N Y I L V P N S K
USH2A
USH2A_exons.bed
6 tracks loaded
chr1:215,844,373
352M of 607M
[student@MSBI3240...
[student@MSBI3240...
[student@MSBI3240...
IGV

USH2A
chr1:215796236-216596738
id = NM_206933

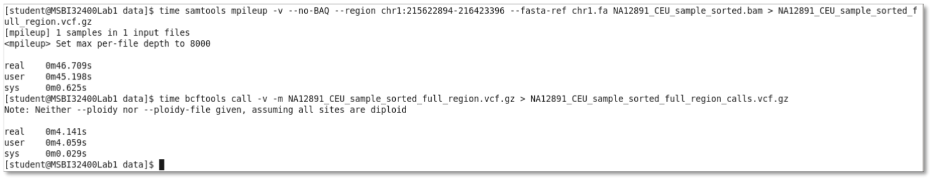
Exon number: 64
Amino acid coding number: 4692
chr1:215844314-215844635

Another way

Vince Buffalo shows:

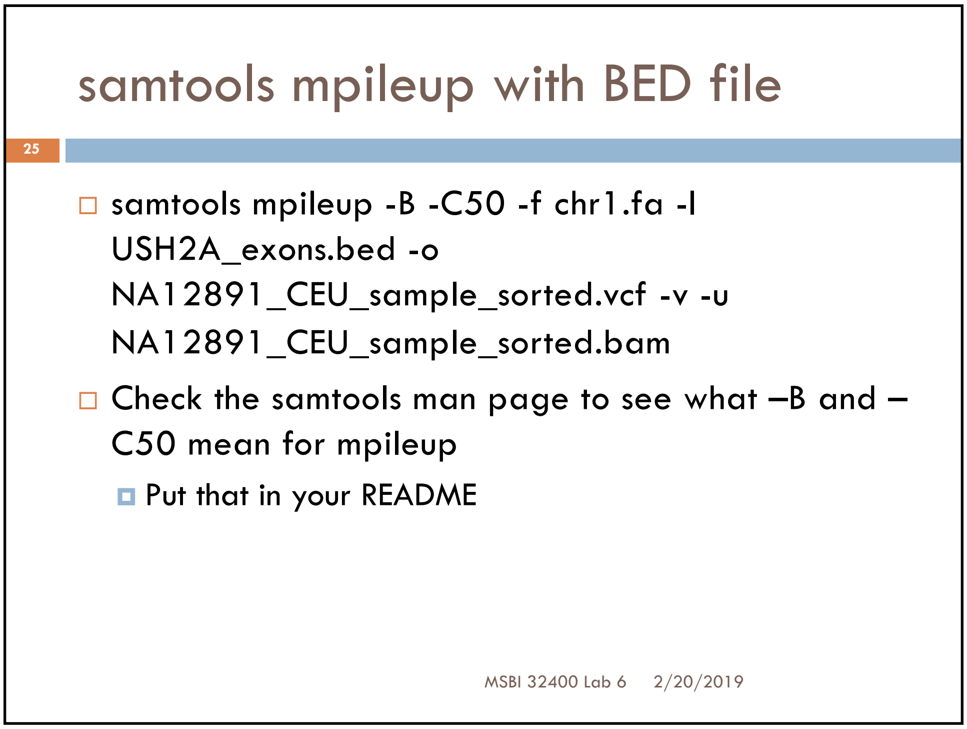
- □ samtools mpileup –v --no-BAQ --region 1:215906528-215906567 –fasta-ref…
  - His coordinates won't work for our BAM since it uses chr1
  - Also, his coordinates are only 39 bp!
  - If you try his notes, use the full sequence from his Chapter 11 README file (chr1:215622894-216423396)

Vince's way



```
[student@MSBI32400Lab1 data]$ time samtools mpileup -v --no-BAQ --region chr1:215622894-216423396 --fasta-ref chr1.fa NA12891_CEU_sample_sorted.bam > NA12891_CEU_sample_sorted_f
ull_region.vcf.gz
[mpileup] 1 samples in 1 input files
<mpileup> Set max per-file depth to 8000

real    0m46.709s
user    0m45.198s
sys     0m0.625s
[student@MSBI32400Lab1 data]$ time bcftools call -v -m NA12891_CEU_sample_sorted_full_region.vcf.gz > NA12891_CEU_sample_sorted_full_region_calls.vcf.gz
Note: Neither --ploidy nor --ploidy-file given, assuming all sites are diploid

real    0m4.141s
user    0m4.059s
sys     0m0.029s
[student@MSBI32400Lab1 data]$ █
```

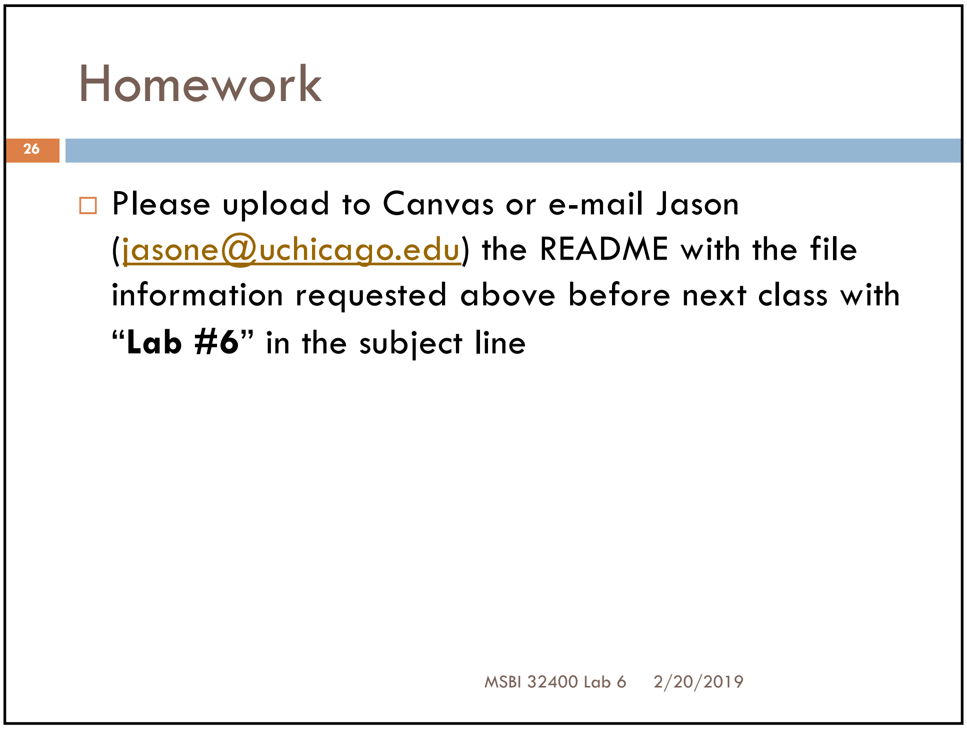
- □ His VCF is the very similar to the one generated before, though he outputs a vcf.gz which is not recognized by IGV or gunzip
- □ Solution: bgzip -d NA12891_CEU_sample_sorted_full_region.vcf.gz then open in IGV or text editor for viewing



samtools mpileup with BED file

25

- samtools mpileup -B -C50 -f chr1.fa -l USH2A_exons.bed -o NA12891_CEU_sample_sorted.vcf -v -u NA12891_CEU_sample_sorted.bam
- Check the samtools man page to see what –B and –C50 mean for mpileup
  - Put that in your README



Homework

26

- Please upload to Canvas or e-mail Jason (jasone@uchicago.edu) the README with the file information requested above before next class with “**Lab #6**” in the subject line