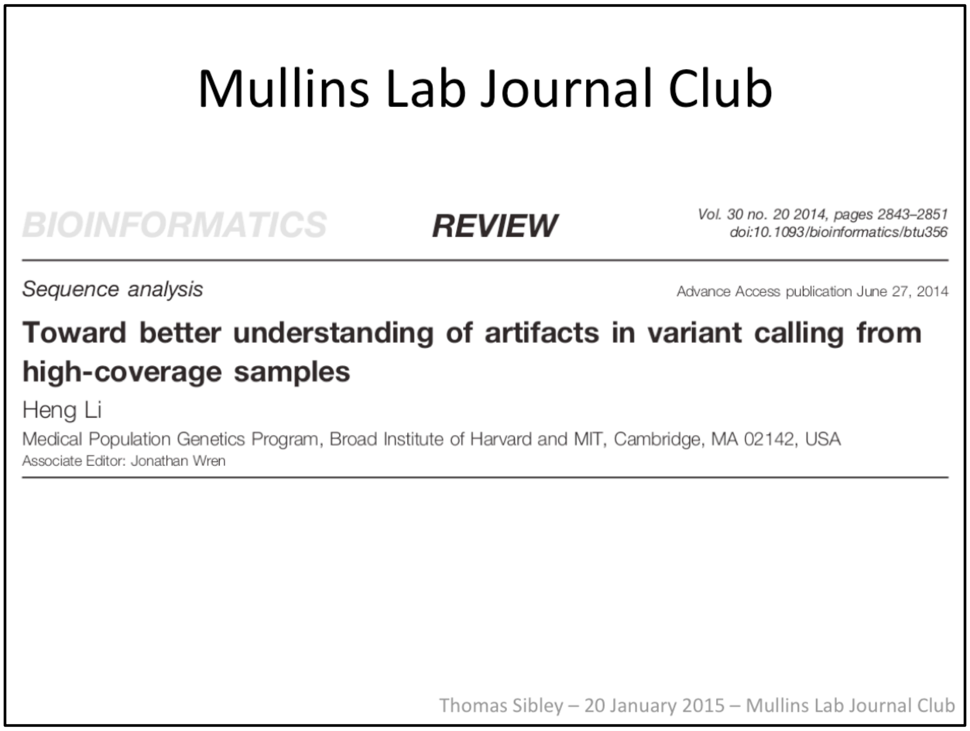


# Mullins Lab Journal Club

BIOINFORMATICS **REVIEW** *Vol. 30 no. 20 2014, pages 2843–2851*
*doi:10.1093/bioinformatics/btu356*

*Sequence analysis* Advance Access publication June 27, 2014

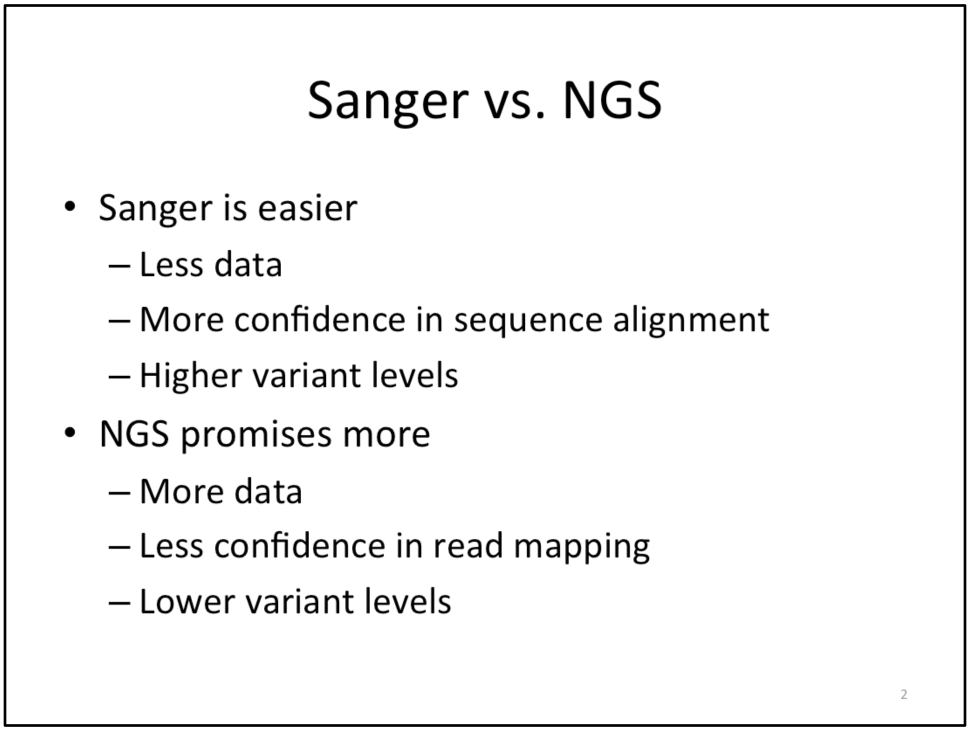
**Toward better understanding of artifacts in variant calling from high-coverage samples**

Heng Li
Medical Population Genetics Program, Broad Institute of Harvard and MIT, Cambridge, MA 02142, USA

Associate Editor: Jonathan Wren

Thomas Sibley – 20 January 2015 – Mullins Lab Journal Club

Heng Li is the creator of many bioinformatics tools for next-gen sequencing, including samtools, BWA, and MAQ. He's widely regarded as an expert on the analysis of sequences. I came across this paper while working on a data analysis pipeline for the drug-resistance project Ross is working on in Lisa's lab.

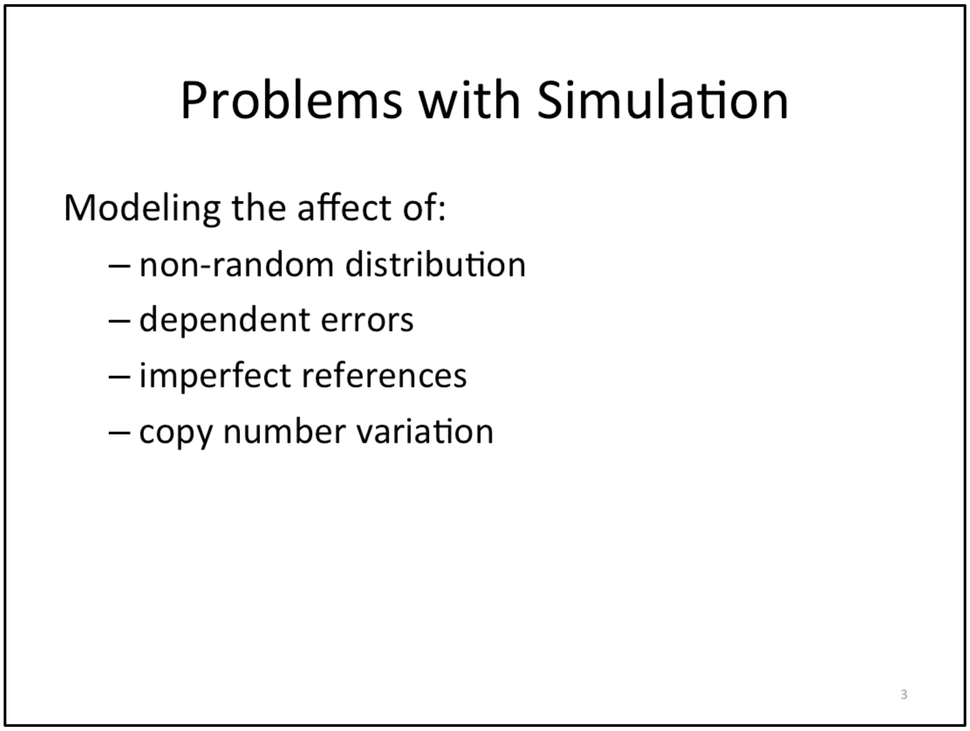


## Sanger vs. NGS

- Sanger is easier
  - Less data
  - More confidence in sequence alignment
  - Higher variant levels
- NGS promises more
  - More data
  - Less confidence in read mapping
  - Lower variant levels

For NGS data, the total error rate of SNP/SNV call is pretty unclear. Additionally, there are many possible sources of error, and it's unclear which are the leading causes of error in calls. There's no good, standardized, general-purpose "truth set" for calibration of SNP calls. The lack of both of these make it hard to determine "good" from "bad" SNP calls, or assign a probability of correctness.

That said, being able to detect low-level variation reliably is useful in innumerable contexts, so we try to do this anyway with all sorts of modeling, filtering, and thresholding.



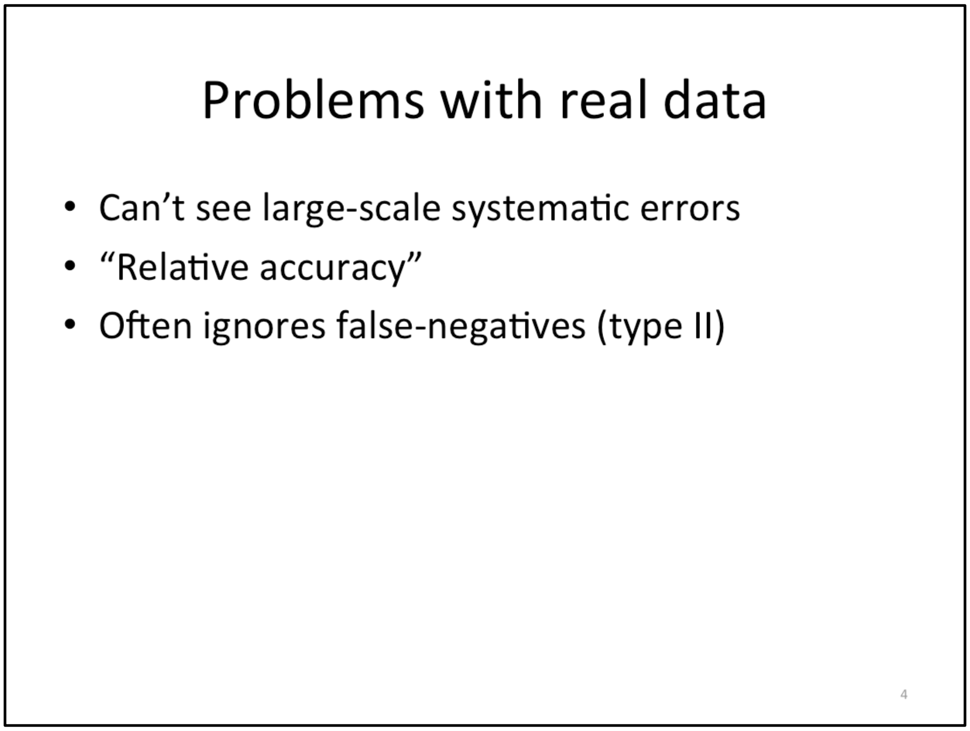
# Problems with Simulation

Modeling the affect of:

- non-random distribution
- dependent errors
- imperfect references
- copy number variation

Simulation of NGS data is one potential means of calibrating SNP callers, but it suffers from a number of intractable problems causing diminishing returns that make it less appealing.

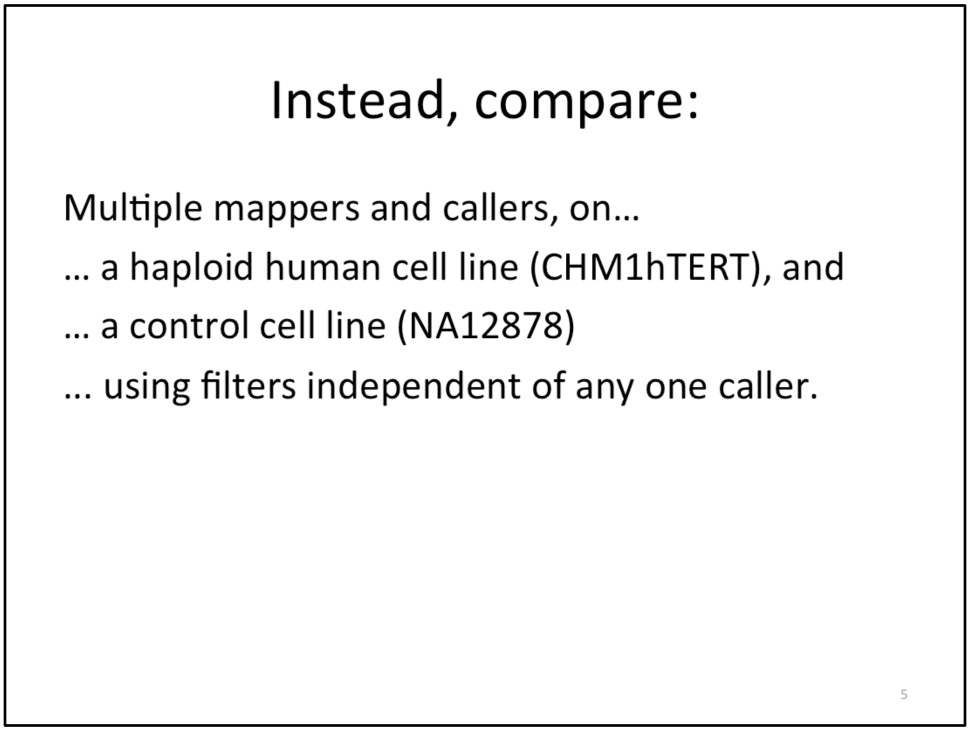
All of these factors affect or have the potential to affect variant calls, but they are also hard to simulate realistically. If we try, do the results say more about the simulated data itself than the variant calling methods? If we ignore them, we underestimate error rates.



# Problems with real data

- Can't see large-scale systematic errors
- "Relative accuracy"
- Often ignores false-negatives (type II)

Real data is an improvement in terms of a calibration truth set, but it too has issues which are hard to resolve. For example, using real variant data hides systematic errors which all callers make. Comparison methods operating on real data, either via experimental genotyping of select sites or comparing against other tools/pipelines, only produce a relative accuracy without any grounding point for overall accuracy. False-negatives are usually hard to ascertain since they require some level of ground-truth, and comparison methods which only look at existing data don't necessarily tell the full story on these type II errors. Li notes that experimental validation using microarrays tends towards easier regions, and so doesn't necessarily provide an unbiased look across all regions.



## Instead, compare:

Multiple mappers and callers, on…

… a haploid human cell line (CHM1hTERT), and

… a control cell line (NA12878)

... using filters independent of any one caller.

Li’s solution to these problems with simulation and real variant data is to instead use a haploid human cell line to call variants using multiple mappers (aligners) and callers. The hypothesis is that “vast majority” of heterozygous calls should be errors of some sort, and the affect of extent of the true heterozygous sites in the cell line (such as those via somatic mutations) should only put a lower bound on the error rate estimation. Theoretically the fewer heterozygous variants called, the better the caller. The haploid cell line is derived from a molar pregnancy where the egg was missing a nucleus.

A positive control cell line is used to avoid overrating callers which simply aren’t as sensitive. Using filters independent of any particular caller provides another level of control on the variant calls, and also hopefully suggests good overall filters regardless of the specific pipeline used.

CHM1 is from a PCR library, the control is not.

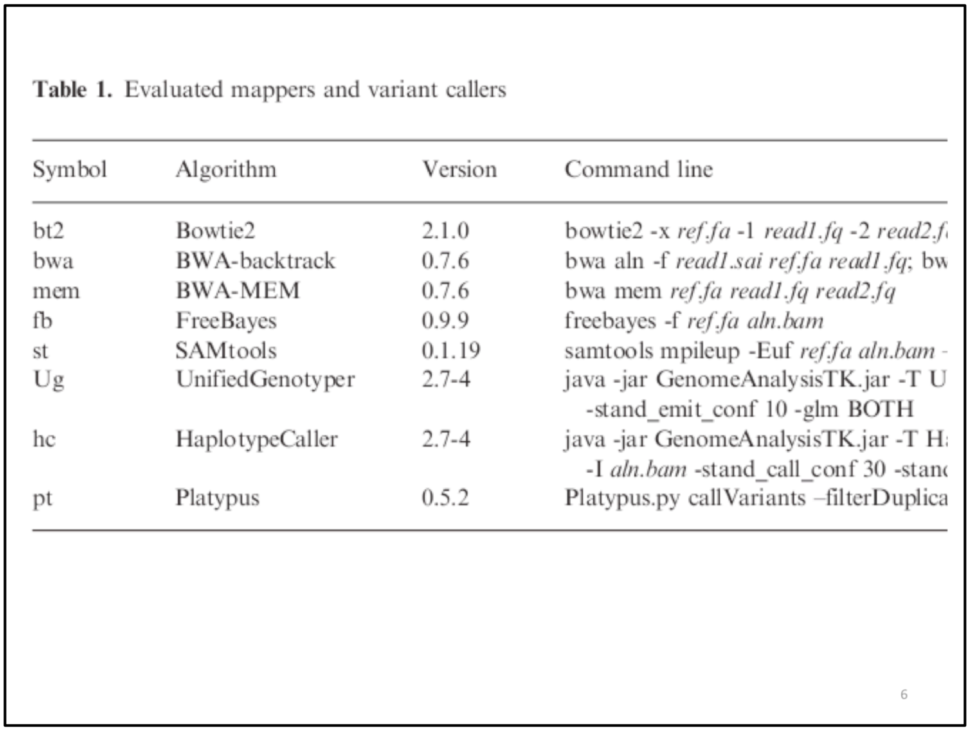
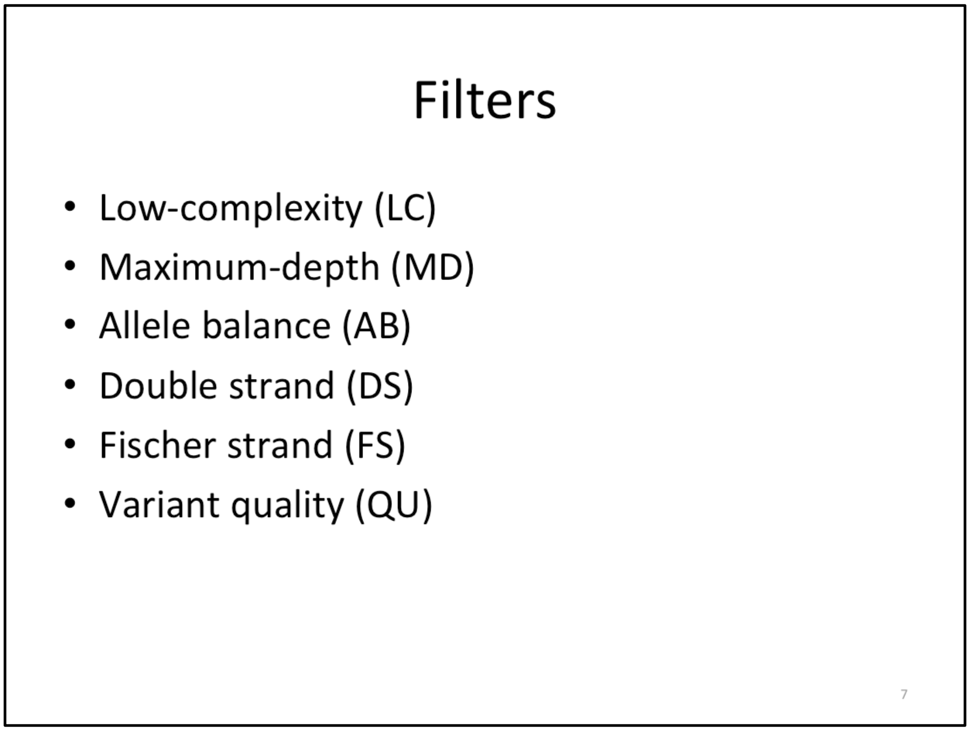


**Table 1.** Evaluated mappers and variant callers

| Symbol | Algorithm | Version | Command line |
|---|---|---|---|
| bt2 | Bowtie2 | 2.1.0 | bowtie2 -x *ref.fa* -1 *read1.fq* -2 *read2.f* |
| bwa | BWA-backtrack | 0.7.6 | bwa aln -f *read1.sai ref.fa read1.fq*; bw |
| mem | BWA-MEM | 0.7.6 | bwa mem *ref.fa read1.fq read2.fq* |
| fb | FreeBayes | 0.9.9 | freebayes -f *ref.fa aln.bam* |
| st | SAMtools | 0.1.19 | samtools mpileup -Euf *ref.fa aln.bam* - |
| Ug | UnifiedGenotyper | 2.7-4 | java -jar GenomeAnalysisTK.jar -T U<br>-stand_emit_conf 10 -glm BOTH |
| hc | HaplotypeCaller | 2.7-4 | java -jar GenomeAnalysisTK.jar -T H<br>-I *aln.bam* -stand_call_conf 30 -stan |
| pt | Platypus | 0.5.2 | Platypus.py callVariants –filterDuplica |

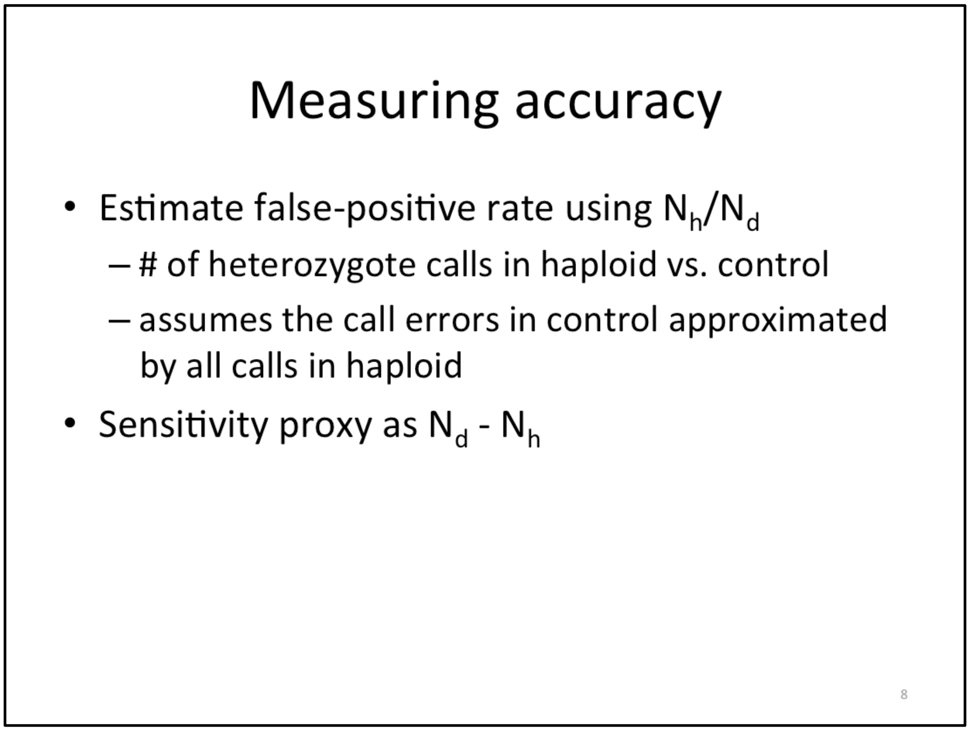
The two primary mappers used were Bowtie2 and BWA-MEM (maximal exact matches). The callers are Freebayes, from the Marth lab at Boston College, samtools, two different ones from the Genome Analysis ToolKit (GATK) by the Broad Institute, and Platypus.



## Filters

- Low-complexity (LC)
- Maximum-depth (MD)
- Allele balance (AB)
- Double strand (DS)
- Fischer strand (FS)
- Variant quality (QU)

Panel of filters, which are applied in order regardless of pipeline, so they're cumulative.



## Measuring accuracy

- Estimate false-positive rate using $N_h/N_d$
  - # of heterozygote calls in haploid vs. control
  - assumes the call errors in control approximated by all calls in haploid
- Sensitivity proxy as $N_d$ - $N_h$

The paper asserts that the two sequence datasets are very similar.  For example, both are from 100bp Illumina reads, both have about the same coverage (after PCR duplicate removal from CHM1 data), and both have roughly the same number of called variants per haplotype. Haplotypes in this case are alleles which are close to each other and statistically associated in the data, or more simply put, alleles which occur together.

All right, so let’s look at the first figure.

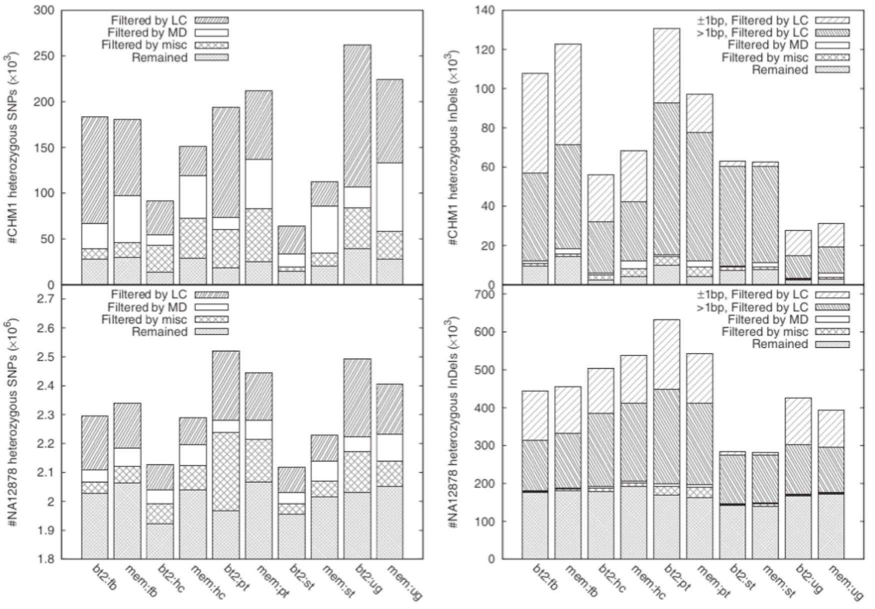


**Fig. 1.** Effect of filters. LC filter: not overlapping LCRs identified by the DUST algorithm. MD filter: read depth below $d+3\sqrt{d}$, where $d$ is the average read depth. Miscellaneous filter (misc) includes three filters: allele balance above 30%, variants supported by non-reference reads on both strands and Fisher strand $P$-value is >0.01. Filters are applied in the order of LC, MD and misc, with MD applied to variants passing LC, and misc applied to variants passing both LC and MD. For each call set, the total height of the bar gives the number of raw variant calls with the reported quality in VCF no <30. Note that the Y-axes are scaled differently

Calls after filtering. Columns alternate by mapper (bt2 then mem), grouped by caller. Note that y axes are in thousands except the lower left, which is in millions, and the scales are different. Note lots of 1bp indels filtered by LC, in both data sets, which implies it's not just PCR error.

80-90% of heterozygous indel calls in the haploid cell line fall within LCRs, while up to 60% of heterozygous SNP calls do. False-positive rate for indels is calculated as 10-40% in LCRs, but only 1-8% outside. If you assume that all calls in CHM1 are errors, then after filtering the error rate looks like about 15-30k per human genome (~3 billion bp), or 1 error per 100-200kbp.

The question still remains: Is the haploid cell line actually haploid? What fraction of heterozygous calls are true positives rather than false positives is important, and if it's too high than it's not a good negative control. The paper tries to address that with the next two figures.

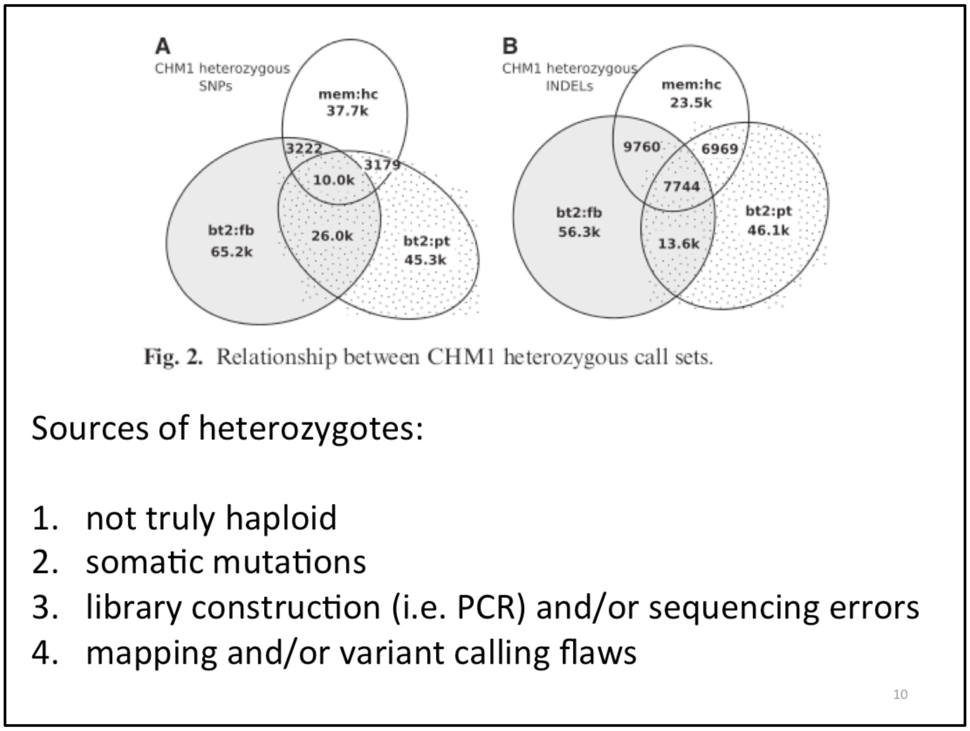


**Fig. 2.** Relationship between CHM1 heterozygous call sets.

Sources of heterozygotes:

1. not truly haploid
2. somatic mutations
3. library construction (i.e. PCR) and/or sequencing errors
4. mapping and/or variant calling flaws

The paper gives the following options for possible sources of heterozygotes. Obviously the paper thinks, or hopes, 1 and 2 are minimal and that 3 and 4 are the vast majority of sources. In particular, the premise is that the mappers and callers should take 3 into account, and so error rate estimations lump the last two sources together.

The author seems to be convinced that CHM1 is effectively haploid due to the a) small overlap of heterozygous callsets in figure 2 (suggesting sequencing error and/or algorithm artifacts as causes) and b) that there is an order of magnitude difference between the raw calls against the positive control vs. the haploid line (the implication being that there should be 10x more if it wasn't haploid). However, Li speculates after a perusal of other datasets that "there may be 5-20k heterozygotes in CHM1 with strong alignment support from multiple Illumina libraries," but that "it is hard to get a more accurate estimate … with the data we are using." … which isn't exactly confidence enhancing.

However, the argument goes, even if the 3-way intersections of figure 2 are treated as real heterozygotes, then it only places a lower bound of error rate estimation at 17.7k per genome, or about 1 per 170k. The rest of the heterozygous calls made in CHM1 by callers can be used to calculate an error rate at least above that lower bound of accuracy. Assuming all that, the best caller post-filtering seems to be mem:hc, which has an error rate of 2-3 per 170k.

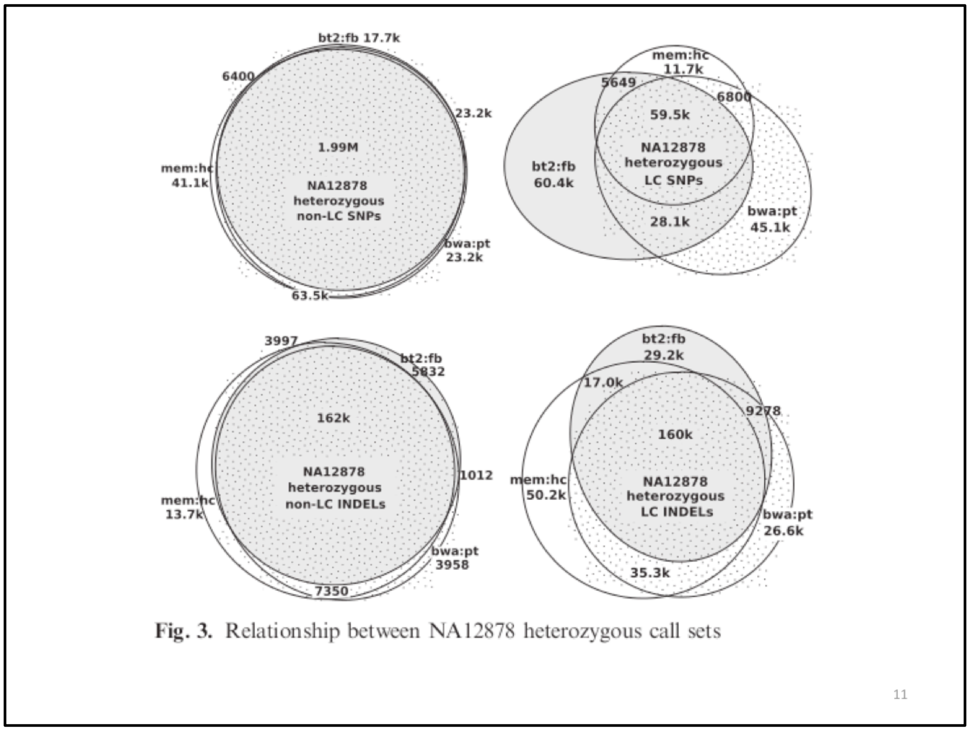
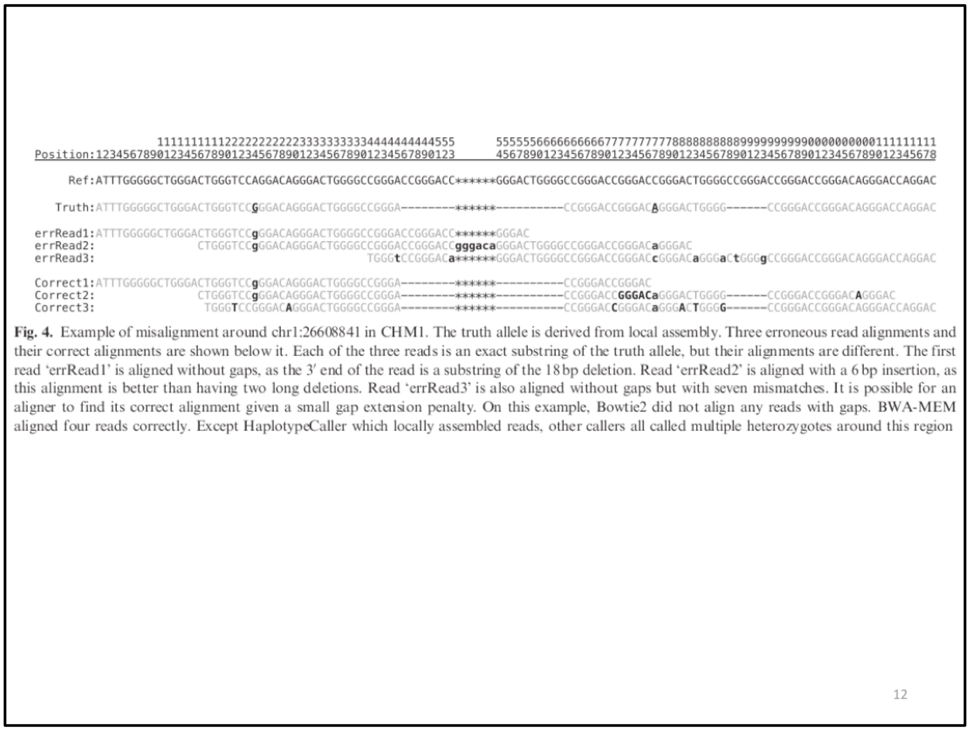


**Fig. 3.** Relationship between NA12878 heterozygous call sets

Now this it the positive control data, split by SNP vs. indel and LC vs. non-LC.

Post-filtering, the callers all had very high agreement on the positive control dataset, and indel false-positive rates are reported in LCRs as 10-40% depending on the callset. Outside LCRs, indel false positive rates range from 1-8%.

The noticeably smaller 3-way overlap in LCRs reconfirms that there's a lot to be said for good local alignments around indels. PCR error is ruled out as the primary cause due to the similar rates of LC filtering of 1bp indels seen in figure 1 between both data sets, even though the positive control was not prepped with PCR. This leaves the mapping and alignment as a source. For example…

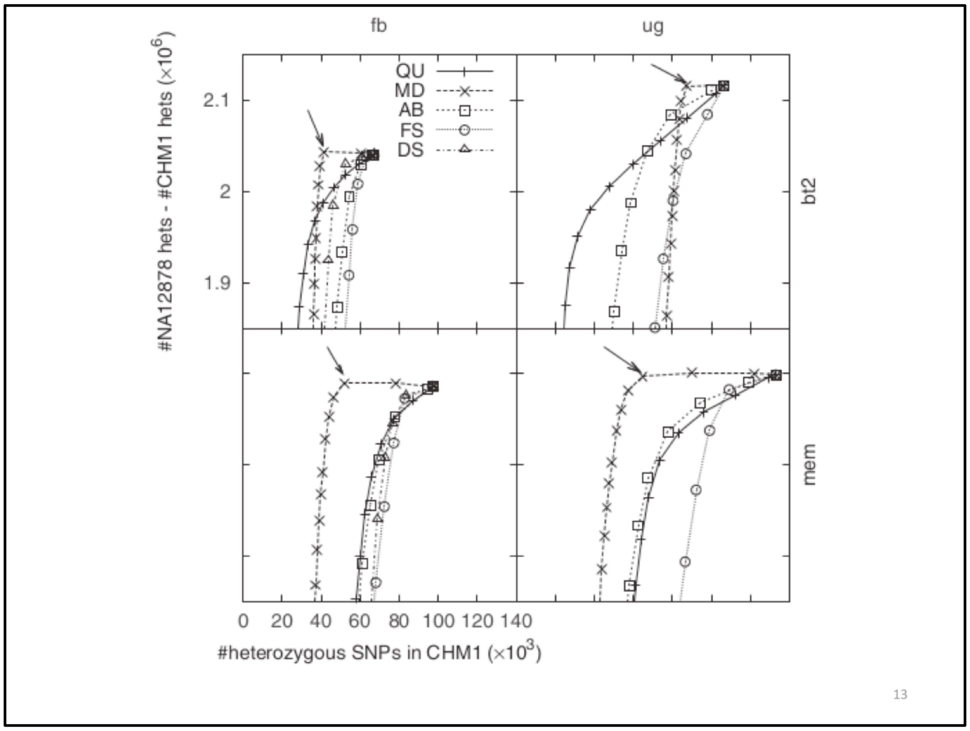


```
                  1111111111222222222233333333334444444444555      555555666666666677777777778888888888999999999900000000001111111111
         Position:12345678901234567890123456789012345678901234567890123      45678901234567890123456789012345678901234567890123456789012345678

     Ref:ATTTGGGGGCTGGGACTGGGTCCAGGACAGGGACTGGGGCCGGGACCGGGACC******GGGACTGGGGCCGGGACCGGGACCGGGACTGGGGCCGGGACCGGGACCGGGACAGGGACCAGGAC

   Truth:ATTTGGGGGCTGGGACTGGGTCCGGGACAGGGACTGGGGCCGGGA--------******----------CCGGGACCGGGACAGGGACTGGGG------CCGGGACCGGGACAGGGACCAGGAC

errRead1:ATTTGGGGGCTGGGACTGGGTCCgGGACAGGGACTGGGGCCGGGACCGGGACC******GGGAC
errRead2:               CTGGGTCCgGGACAGGGACTGGGGCCGGGACCGGGACCgggacaGGGACTGGGGCCGGGACCGGGACaGGGAC
errRead3:                                        TGGGtCCGGGACa******GGGACTGGGGCCGGGACCGGGACcGGGACaGGGaCtGGGgCCGGGACCGGGACAGGGACCAGGAC

Correct1:ATTTGGGGGCTGGGACTGGGTCCgGGACAGGGACTGGGGCCGGGA--------******----------CCGGGACCGGGAC
Correct2:               CTGGGTCCgGGACAGGGACTGGGGCCGGGA--------******----------CCGGGACCGGGACaGGGACTGGGG------CCGGGACCGGGACAGGGAC
Correct3:                TGGGTCCGGGACAGGGACTGGGGCCGGGA--------******----------CCGGGACCGGGACaGGGACTGGGG------CCGGGACCGGGACAGGGACCAGGAC
```

**Fig. 4.** Example of misalignment around chr1:26608841 in CHM1. The truth allele is derived from local assembly. Three erroneous read alignments and their correct alignments are shown below it. Each of the three reads is an exact substring of the truth allele, but their alignments are different. The first read 'errRead1' is aligned without gaps, as the 3′ end of the read is a substring of the 18 bp deletion. Read 'errRead2' is aligned with a 6 bp insertion, as this alignment is better than having two long deletions. Read 'errRead3' is also aligned without gaps but with seven mismatches. It is possible for an aligner to find its correct alignment given a small gap extension penalty. On this example, Bowtie2 did not align any reads with gaps. BWA-MEM aligned four reads correctly. Except HaplotypeCaller which locally assembled reads, other callers all called multiple heterozygotes around this region

12

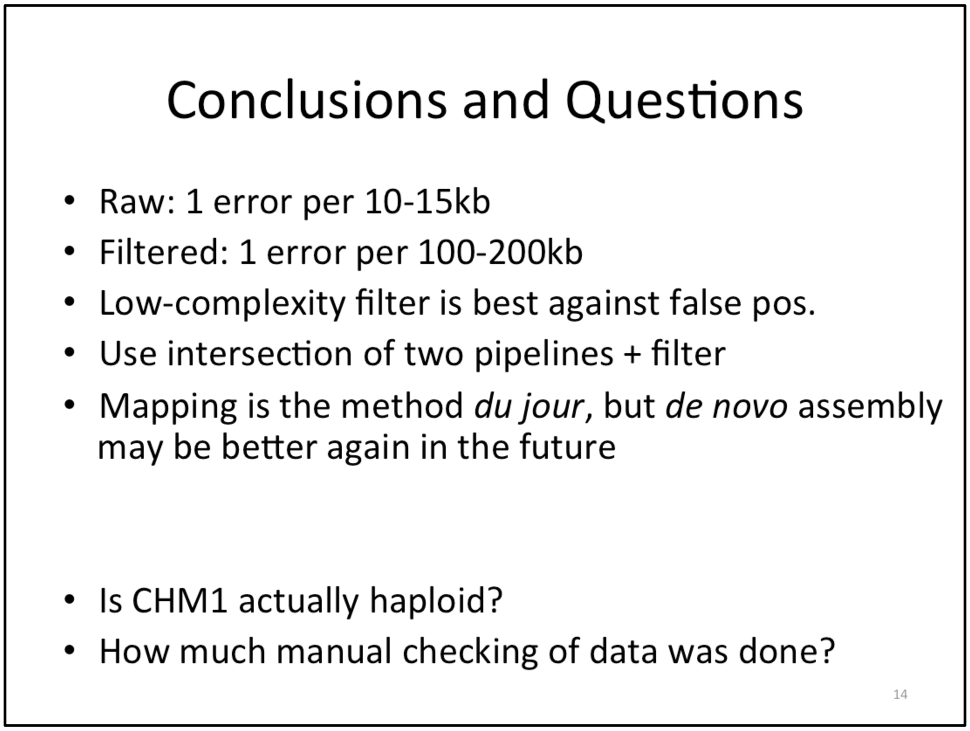
Most of you have probably stared at troublesome regions like this, although perhaps not with reads as short. Local alignment cleared it up, but it demonstrates the large effect that bad mapping can have because it's done independent of other reads.



These are similar to receiver operating characteristic (ROC) plots, which show the various filters (after LC filter) performing at various thresholds (each point) on the false positive vs. true positive axes.

Steeper = better and lefter = better, so curves in the upper left of the plots are good thresholds to use. In this case, the maximum depth filter is most effective, which they interpret filtering out erroneous calls from copy number variations. Their data had an average depth of 50 and a threshold between d+3sqrt(d) and d+4sqrt(d). bwa mem seems to perform the best. It’s hard to know how this would fare on data without mostly uniform read depth.

The other filters stand out less and don’t offer any obvious optimal thresholds.



## Conclusions and Questions

- Raw: 1 error per 10-15kb
- Filtered: 1 error per 100-200kb
- Low-complexity filter is best against false pos.
- Use intersection of two pipelines + filter
- Mapping is the method *du jour*, but *de novo* assembly may be better again in the future

- Is CHM1 actually haploid?
- How much manual checking of data was done?

The raw call sets suggests a calling error every 10-15kb. This is a “total error” estimate, which includes errors caused by PCR, sequencing, alignment, etc. After filtering, it’s an order of magnitude better. Their one concrete suggestion is to use the intersection of two raw callsets and then apply filtering to that shared subset.

The methods Li uses in this paper are intriguing, and I was hoping for results that would help inform our look at variants in HIV. Unfortunately the results mostly support what we already know or suspect, and they don’t contain much in the way of concretely usable findings, at least for us. Part of this is that some techniques are based on the well-studied human genome datasets used and aren’t applicable to all viral sequencing.

Thanks!

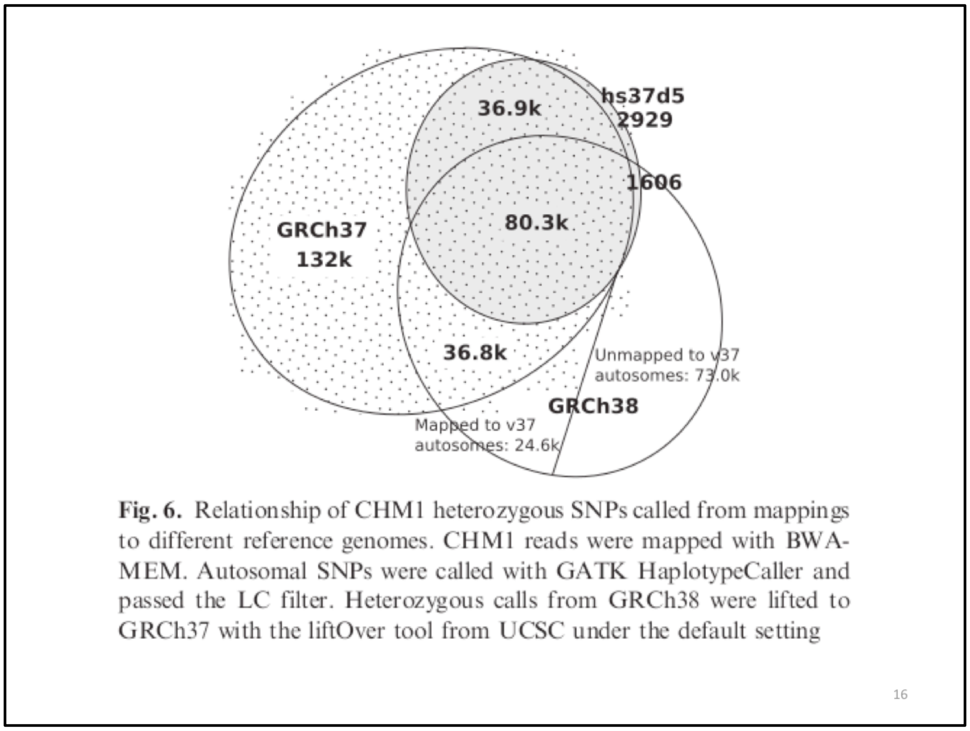


**Fig. 6.** Relationship of CHM1 heterozygous SNPs called from mappings to different reference genomes. CHM1 reads were mapped with BWA-MEM. Autosomal SNPs were called with GATK HaplotypeCaller and passed the LC filter. Heterozygous calls from GRCh38 were lifted to GRCh37 with the liftOver tool from UCSC under the default setting

16

Finally, the paper also considered the effects of the reference genome, in this case the