

So you want to do a Variant Analysis Project

Matt Settles

Director, Bioinformatics Core

Treating Bioinformatics as a Data Science

Seven stages to data science

1. Define the question of interest
2. Get the data
3. Clean the data
4. Explore the data
5. Fit statistical models
6. Communicate the results
7. Make your analysis reproducible

**Data science done well looks easy and
that's a big problem for data scientists**

**simplystatistics.org
March 3, 2015 by Jeff Leek**

What is Variant Analysis

The Focus on identification of variation in DNA from genomic, exome, targeted sequencing, or reduced representation sequencing data.

Designing Experiments

Beginning with the question of interest (and working backwards)

- The final step of an analysis is the application of a model to each variant in your dataset.

Traditional statistical considerations and basic principals of statistical design of experiments apply.

- **Control** for effects of outside variables, avoid/consider possible biases, avoid confounding variables in sample preparation.
 - **Randomization** of samples, plots, etc.
 - **Replication** is essential (minimum here depends on MAF and how strongly the variant is associated with phenotype)
- You should know your final model and comparison of interest before beginning your experiment.

General rules for preparing samples

- Prepare more samples than you are going to need, i.e. expect some will be of poor quality, or fail
- Preparation stages should occur across all samples at the same time (or as close as possible) and by the same person
- Spend time practicing a new technique to produce the highest quality product you can, reliably
- Quality should be established using Fragment analysis traces (pseudo-gel images, RNA RIN > 7.0)
- DNA/RNA should not be degraded
 - 260/280 ratios for RNA should be approximately 2.0 and 260/230 should be between 2.0 and 2.2. Values over 1.8 are acceptable
- Quantity should be determined with a Fluorometer, such as a Qubit.

Generating libraries

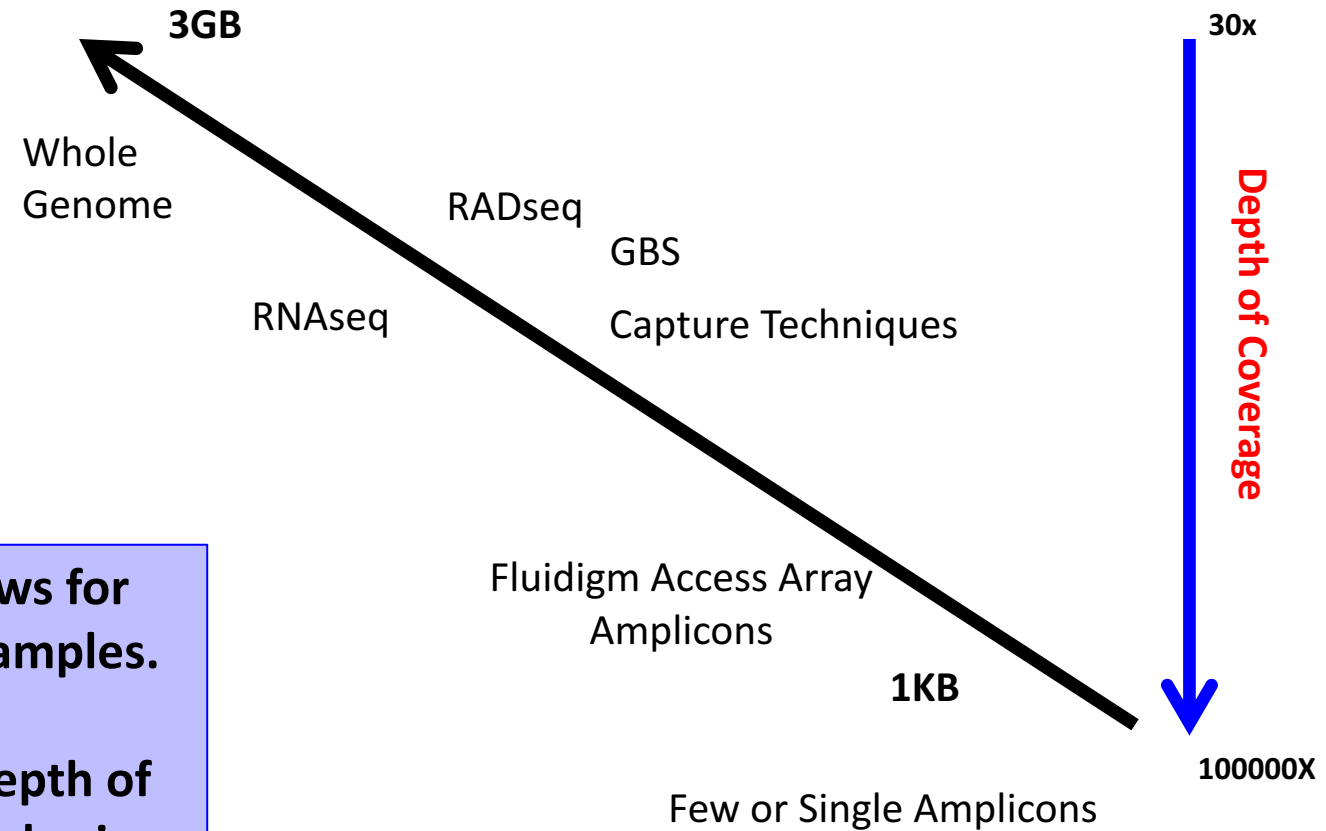
Types of Libraries

- Whole genome shotgun

Reduced Representation Libraries

- Exome capture
- RADseq
- GBS
- Others

Genomic Reduction



Genomic reduction allows for greater multiplexing of samples.

You can fine tune your depth of coverage needs and sample size with the reduction technique

Sequenced Basepairs per samples per lane

The first and most basic question is how many base pairs of sequence will I get

Factors to consider then are:

1. Number of reads being sequenced
2. Read length (if reads are paired, consider them as individuals for this calculations)
3. Number of samples being sequenced
4. Expected percentage of good bases/reads

$$\frac{bp}{sample} = \frac{readLength * (\# reads)}{\# samples} * 0.8$$

The number of reads and read length data are best obtained from the manufacturer's website (search for specifications) and always use the lower end of the estimate.

Genomic Coverage

Once you have the number of base pairs per sample you can then determine expected coverage

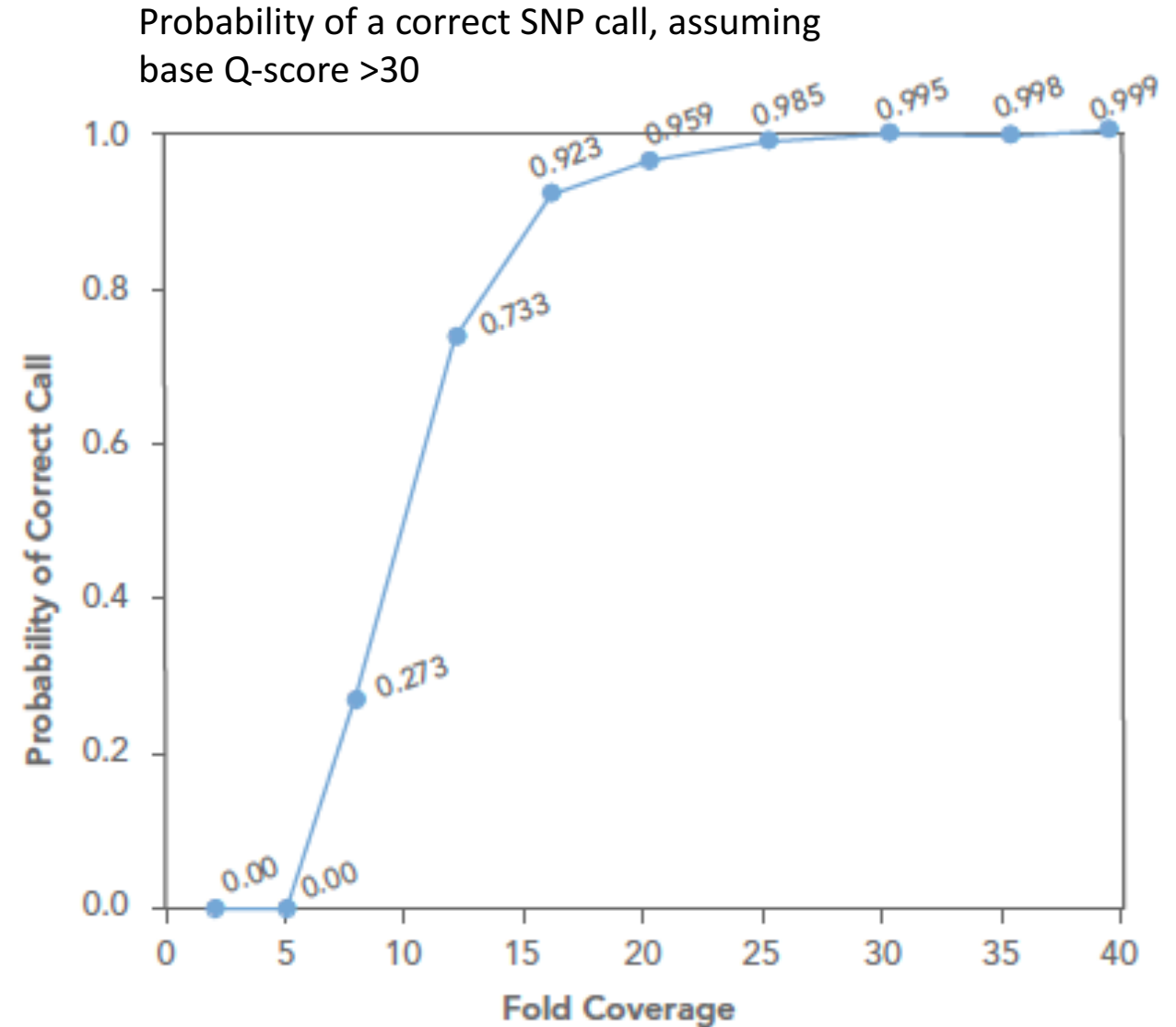
Factors to consider then are:

1. Length of the genome
2. Any extra-genomic sequence (ie mitochondria, virus, plasmids, etc.). For bacteria in particular, these can become a significant percentage

$$\frac{\text{ExpectedCoverage}}{\text{sample}} = \frac{\frac{(\text{readLength} * \text{numReads}) * 0.8}{\text{numSamples}} * \text{num.lanes}}{\text{TotalGenomicContent}}$$

Variant Analysis

- Read length contributes to uniqueness of mapping
- Paired reads are required to identify structure changes
- For a single individual target > 30x coverage.
- In population studies, the greater the number of samples less coverage per samples that is required. (ex. with 1000 samples 2x coverage per sample may be sufficient)

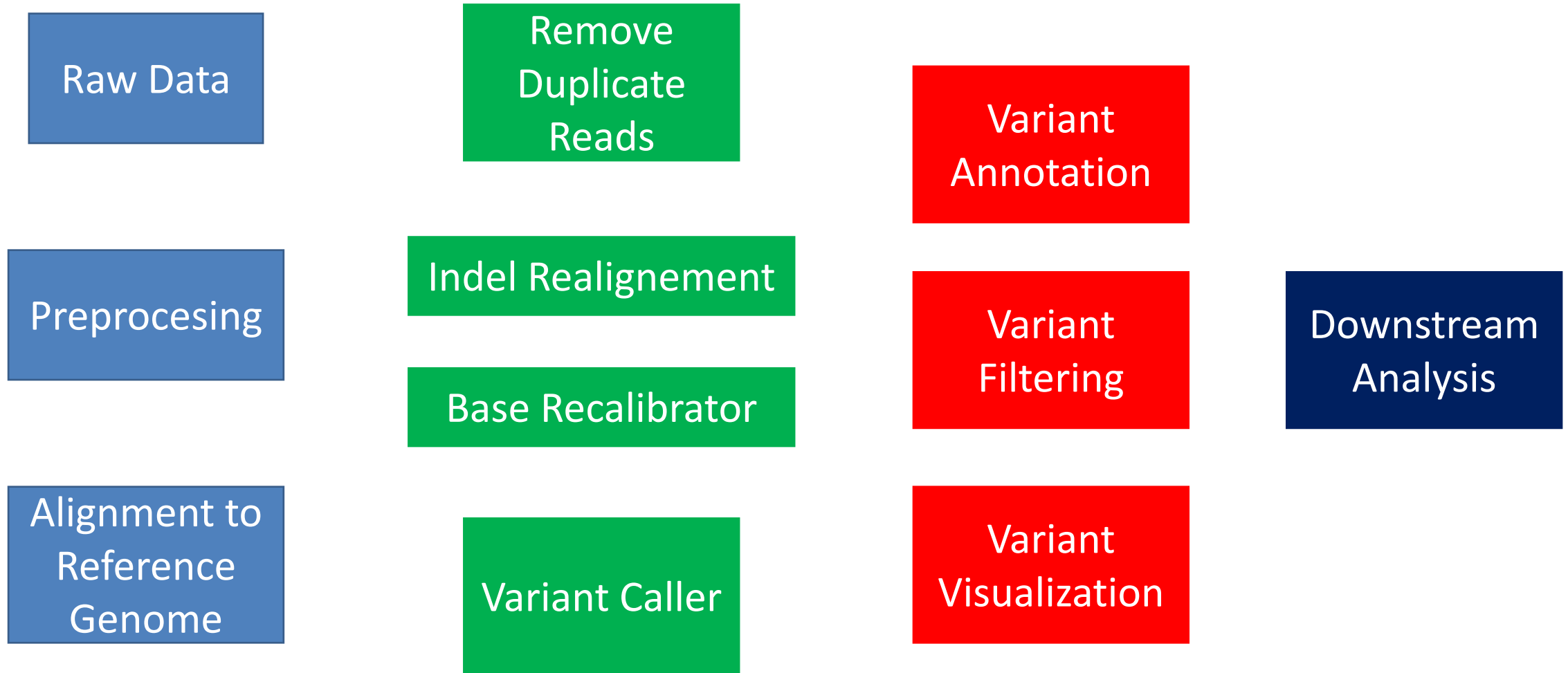


Overview of Variant data analysis

Prerequisites

- Access to a multi-core (24 cpu or greater), 'high' memory 64Gb or greater Linux server.
- Familiarity with the 'command line' and at least one programming language.
- Basic knowledge of how to install software
- Basic knowledge of R (or equivalent) and statistical programming
- Basic knowledge of Statistics and model building

Variant pipeline overview



Sequence Preprocessing: Why?

- We have found that aggressively “cleaning” and processing reads can make a large difference to the **speed** and **quality** of assembly and mapping results. Cleaning your reads means, removing reads/bases that are:
 - other unwanted sequence (polyA tails in RNA-seq data)
 - artificially added onto sequence of primary interest (vectors, adapters, primers)
 - join short overlapping paired-end reads
 - low quality bases
 - originate from PCR duplication
 - not of primary interest (contamination)
- Preprocessing also produces a number of statistics that are technical in nature that should be used to evaluate “experimental consistancy”

Sequence Preprocessing: QA/QC

- Beyond generating better data for downstream analysis, cleaning statistics also give you an idea as to the quality of the sample, library generation, and sequencing quality used to generate the data.
- This can help inform you of what you might do in the future.
- I've found it best to perform QA/QC on both the run as a whole (poor samples can affect other samples) and on the samples themselves as they compare to other samples **(REMEMBER, BE CONSISTANT)**.
 - Reports such as Basespace for Illumina, are great ways to evaluate the runs as a whole.
 - PCA/MDS plots of the preprocessing summary are a great way to look for technical bias across your experiment

Mapping: Consideration

- Placing reads in regions that do not exist in the reference genome (reads extend off the end) [mitochondrial, plasmids, structural variants, etc.].
- Sequencing errors and variations: alignment between read and true source in genome may have more differences than alignment with some other copy of repeat.
- What if the closest fully sequenced genome is too divergent? (3% is a common assumed alignment capability)
- Placing reads in repetitive regions: Some algorithms only return 1 mapping; If multiple: map quality = 0
- Algorithms that use paired-end information => might prefer correct distance over correct alignment.

Mapping: Many Aligners to choose from

https://en.wikipedia.org/wiki/List_of_sequence_alignment_software

Generally BWA works well

- BWA aln for < 70bp reads

- BWA mem for \geq 70bp reads

Mapping: QA/QC

- Mapper produce summary statistics, view the summary report (in a text editor) and compare across samples.
 - Other additional summary statistics can be produced with:
samtools flagstat
samtools idxstats
samtools stats

Indel Realignment, Base Recalibration

- Local realignment of insertions and deletions. Insertion-deletion (indel) mutations, with respect to the reference genome, can contain misalignments across BAM files. Misalignment of indel mutations, which can often be erroneously scored as substitutions, reduces the accuracy of downstream variant calling steps.
- A base quality score recalibration (BQSR). This step adjusts base quality scores based on detectable and systematic errors. This step also increases the accuracy of downstream variant calling algorithms.

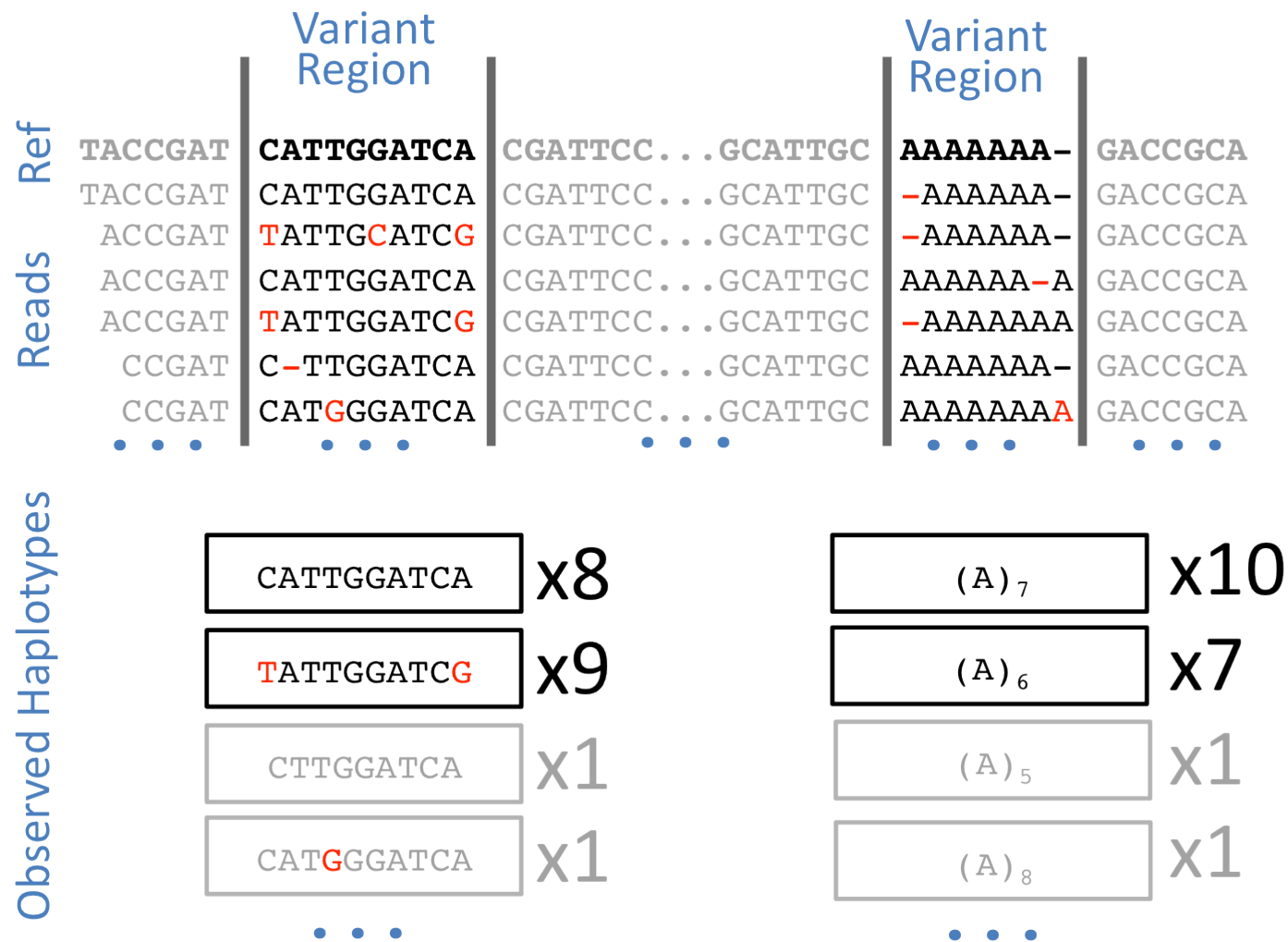
Variant Caller – Haplotype callers

- Older techniques would try and call variants based directly on the alignment, one base at a time. Current 'haplotype' based callers use the reads directly and short sets of sequences to call variants.
- Identify variant bases, genotype likelihood and allele frequency while avoiding instrument noise.
- Variant Analysis produces a VCF (Variant Call Format) file.

Variant Calling – Challenges

- Base Calling Errors:
 - Errors vary by technology/instrument, increase as the numbers of cycles increases and the sequence context (ex. GC)
- Low coverage sequence
 - Not enough sequence coverage to infer the alleles at a site
- Inaccurate mapping
 - Aligned reads should be reported with a mapping score

Variant Calling: Alignments can be incorrect



VCF File format

```
##fileformat=VCFv4.2
##fileDate=20090805
##source=myImputationProgramV3.1
##reference=file:///seq/references/1000GenomesPilot-NCBI36.fasta
##contig=<ID=20,length=62435964,assembly=B36,md5=f126cdf8a6e0c7f379d618ff66beb2da,species="Homo sapiens",taxonomy=x>
##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=A,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=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##FORMAT=<ID=HQ,Number=2,Type=Integer,Description="Haplotype Quality">
```

#CHROM	POS	ID	REF	ALT	QUAL	FILTER	INFO	FORMAT	NA00001	NA00002	M
20	14370	rs6054257	G	A	29	PASS	NS=3;DP=14;AF=0.5;DB;H2	GT:GQ:DP:HQ	0 0:48:1:51,51	1 0:48:8:51,51	1
20	17330	.	T	A	3	q10	NS=3;DP=11;AF=0.017	GT:GQ:DP:HQ	0 0:49:3:58,50	0 1:3:5:65,3	0
20	1110696	rs6040355	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
20	1230237	.	T	.	47	PASS	NS=3;DP=13;AA=T	GT:GQ:DP:HQ	0 0:54:7:56,60	0 0:48:4:51,51	0
20	1234567	microsat1	GTC	G,GTCT	50	PASS	NS=3;DP=9;AA=G	GT:GQ:DP	0/1:35:4	0/2:17:2	1

Variant Annotation – SNPeff

Variant annotation and effect prediction tool

- Using a genome fasta file (same used to map reads to) and a gtf genome annotation file, creates a database specific to an organism.
- Given a VCF file with predicted variants, SNPs, INDEL, etc.
- Produces an annotated VCF (Addition of the ANN field), calculates the effects they produce on known genes (e.g. amino acid changes).
- Variant annotation uses the coordinates and alleles in the VCF file to infer biological context for each variant including the location of each mutation, its biological consequence (frameshift/silent mutation), and any affected genes.

Variant Filtering

Variant analysis usually produces hundreds of thousands (euk) of candidate variants, not all are 'real' or 'interesting'. Filtering is usually experiment based, and takes into account expectations.

- Remove all low quality variants
- In gene/ not in gene
- Synonymous/non-Synonymous SNPs
- Consider only rare-mutations
- Comparison to public data

Visualization - IGV



Downstream Analysis

- Downstream analysis is often to determine genotype-phenotype association from sequencing data (Based on GWAS)
- Application of statistical models to determine whether SNPs or Haplotypes are associated with a phenotype
- Gene Set Enrichment Analysis for variants in Pathways/GO

Software

Preprocessing:

- Python 2.7
 - Modules: argparse, optparse, distutils
- bowtie2 - contaminant screening
 - <http://bowtie-bio.sourceforge.net/bowtie2/index.shtml>
- Super-Deduper – Identify and remove PCR duplicates
 - <https://github.com/dstreett/Super-Deduper>
- Sickle – Trim low quality regions
 - <https://github.com/dstreett/sickle>
- Scythe – Identify and remove adapters in SE reads
 - <https://github.com/ucdavis-bioinformatics/scythe>
- FLASH2 – Join overlapping reads, identify and remove adapter in PE reads
 - <https://github.com/dstreett/FLASH2>

Software

Mapping:

- Bwa mem – map reads to a reference
 - <http://sourceforge.net/projects/bio-bwa/files/>
- samtools – processing of sam/bam file
 - <http://www.htslib.org/>

Mapping adjustments, Variant Analysis:

- GATK – Genome Analysis Tool Kit
 - <https://software.broadinstitute.org/gatk/>
- Freebayes
 - <https://github.com/ekg/freebayes>

Software

Variant Annotation and Filtering

- SNPeff
 - <http://snpeff.sourceforge.net/>
- SnpSift
 - <http://snpeff.sourceforge.net/SnpSift.html>
- Annovar
 - <http://annovar.openbioinformatics.org/en/latest/>

Visualization

- IGV - Integrative Genome Viewer
 - <http://software.broadinstitute.org/software/igv/>

Software

Analysis of Variants:

- Plink-seq
 - <https://atgu.mgh.harvard.edu/plinkseq/>
- EPACKTS
 - <https://genome.sph.umich.edu/wiki/EPACKTS>
- R <http://www.r-project.org/>
 - R Packages: TVTB, VariantTools
 - <http://bioconductor.org/packages/release/bioc/html/TVTB.html>
 - <http://bioconductor.org/packages/release/bioc/html/VariantTools.html>
- RStudio
 - <https://www.rstudio.com/>