Variant Calling and Filtering for INDELs

Erik Garrison SeqShop @ University of Michi

Overview

 What are the causes of insertion/deletion (INDEL) mutations?

2. How do we detect INDELs using resequencing methods?

3. Bayesian variant calling. Haplotype-based calling.

An INDEL

A mutation that results from the gain or loss of sequence.

AATTAGCCATTA

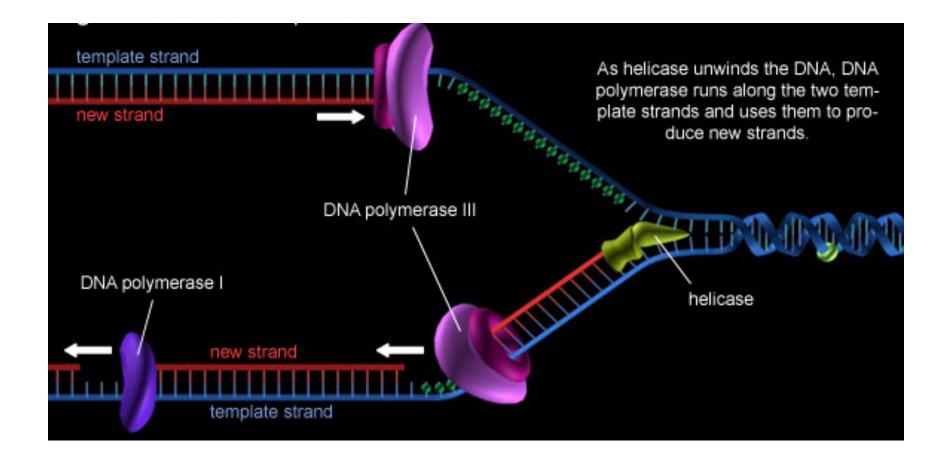
AATTA--CATTA

INDEL genesis

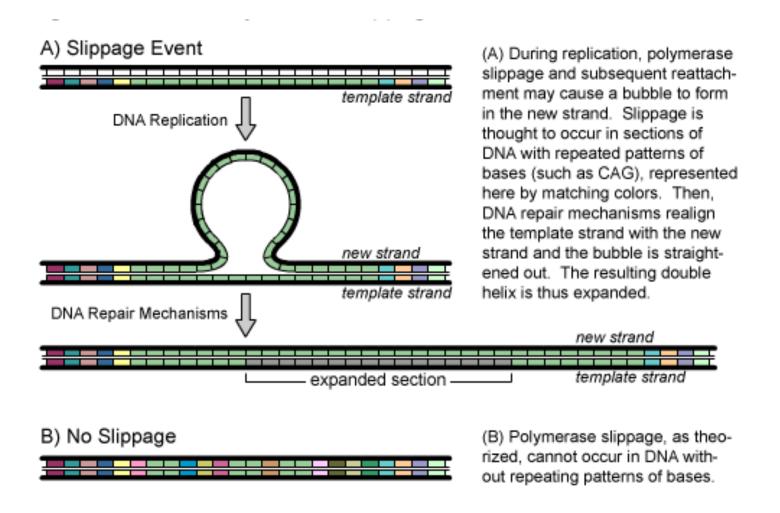
A number of processes are known to generate insertions and deletions in the process of DNA replication:

- Replication slippage
- Double-stranded break repair
- Structural variation (e.g. mobile element insertions, CNVs)

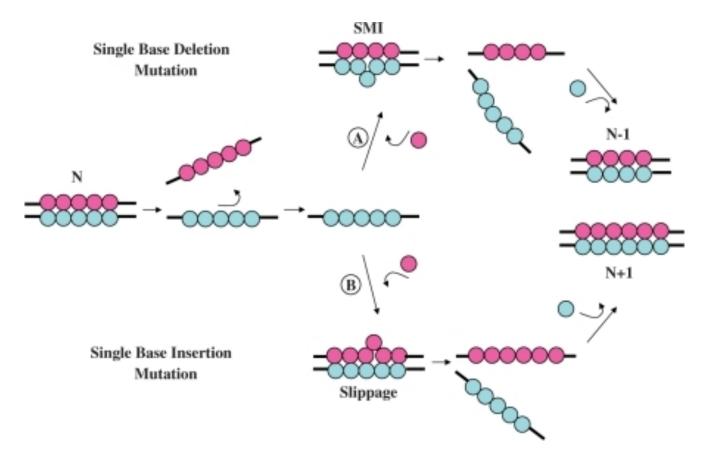
DNA replication



Polymerase slippage

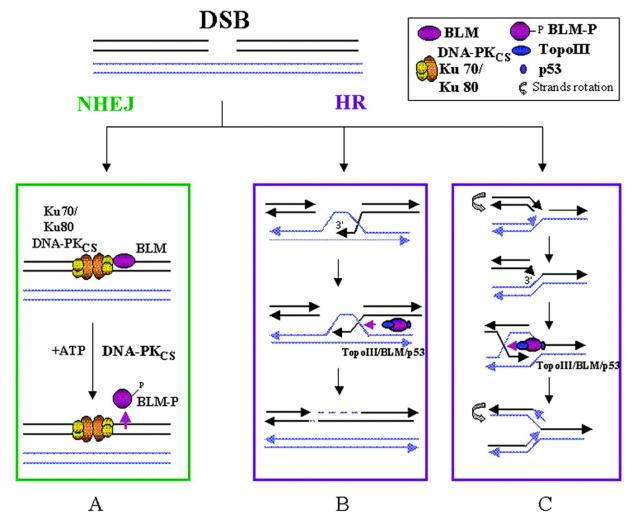


Insertions and deletions via slippage



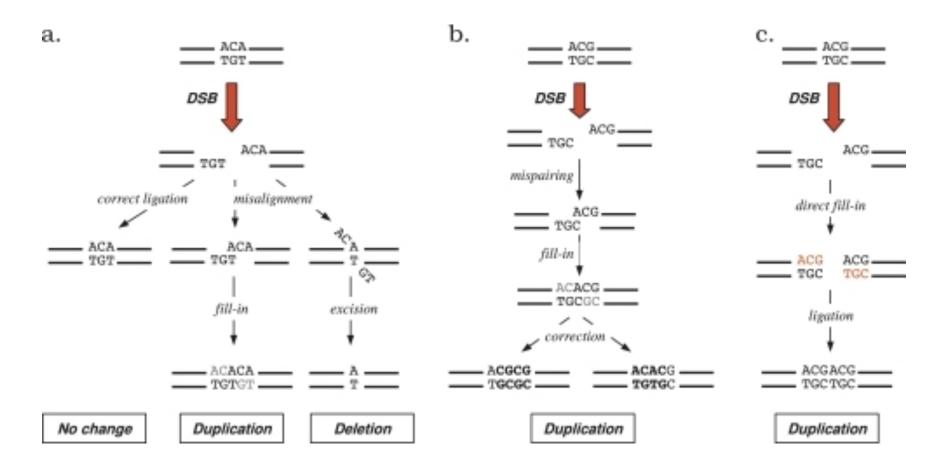
Energetic signatures of single base bulges: thermodynamic consequences and biological implications. Minetti CA, Remeta DP, Dickstein R, Breslauer KJ - Nucleic Acids Res. (2009)

Double-stranded break repair



Possible anti-recombinogenic role of Bloom's syndrome helicase in double-strand break processing. doi: 10.1093/nar/gkg834

NHEJ-derived indels



Calling INDEL variation

Design a process to detect indels from alignment data.

Take ~5 minutes with a partner (or two) and write down the steps your algorithm would execute to detect indels.

Finding variation

CAAATAGACTTCCCCATAACACAAAGCCATCCTGAAAAGTTTTGTTCATTTTAGAAGAAAAAATTTTAAAACCTGAGCAC AAAAGTTCCATAATGAGTAATCAAATTTTATTTCCAATTAAATATGTTATCACCCAGTAGTATGCCATACCAGTTTCTAA TTCATACTTCAATTATCTTCTAATTTAAATTAATGACTATAATTGCTGTTATAAAACAACAGCTCTATAGCCTGCTATTC AGACCAGTAAATAAGAGTTTAAGGGCTTGTGATAGCAAATGAAGTTTCTTATTGGATTTTAAGAAAAATTTTTATAAAAA TATGTGAGGTTATTCAATAGAATCACATTTAATTTGCCAAGCATTTTGCAGAATGCCTAGGACTATGTAAGAAGTATTAA ATTTGCAAGCCCTTTGAATAGTTGTAATTTAAAGATAAAAATTGGTTTAATACCAGACAAAGATAGAAGCACAAGTTAGG TTATTAGAGAATTTAGCCAGTGTATCAGTTTGTATCGTAAGTCATTGGCAAGAACAACGTGTACTTTTCTGTCACCTCCC AACTAGCTATGTTTTGAGCAGTAGGAATATTTAATACCCCTTCCTCCCATTTTTCCTTTGTGTTGTCCAAATTCTGACAA CTCTACTGCCAGATAGCTCAGGGCAAAAATGATAAAGTTCAAGTTAAGAAGGCTCTGCAGTGTTCTCAGTTCTCCTCTGG TGAAAGAGAGAAAGGTTGTGTTTAATTATGAATCTGGGATTTCCAAAACTTTACCCATGCCCTGCCCTGTCCCCTCATTA GCATGAAGCTGTTATTTAAATAGTTCAGCAATAACGACTTTAGTAGCCTCCCTAGGTTAAAAAGATTGAAATTAAATGTG TTTATCTATTGTTCTACTATTCAGTTACCTGATTATAAAATCAAAGATTATTTCATGAAACTCAGTACCCCTTCAGGGAA AAATGCTTACCCAACTTCTATTCAAAATATTTGCGCCAGTAGTTCTGATATGACCCAAGCAGAGTTCACACATTATTAAT CTACTCCTTTCAGTCTTCTAGATGTGTTTCCTCCAAAATCTACCAGATTCTCAAATAATTTCAGGAACTTTCTCCAGAAC AGAAACAAGGTTGTTACTGATACCAACTTTGTCTCCAAACATGGGGAAGATTATCATTGGAAAGATCTATTGATGACCTA TAATACATAGTTGGAACTGTTTATCCACAGAAGTATTCCCCCAAGAATCAACCACAGAGCCAAGATGGAGCTTATGTCATT GTTATGCATACTTCTTTTACGGCTTGTGAGGGCAGTTCATACTATTCTGATTTTACAACTGAGACCCAAGGAACCTGAGT GACTTCTAGGCTCCATTATGTCAAAAAAAACTCAAATGTGAGGCTTTGCCTACACTGAGAAACAGTAGTTCAAGAAACGG TGCCCTGGTTCTGTTAAAATAATCTGAGAGTTATGTGGTAAGTAGTTGAGAGTGAATAGGGTAGCTTTGAGAGGTGACAG CGTGCTGGCAGTCCTCACAGCCCTCGCTGGCTCCAGGCGCCTCCTCTGCCTGGGCTCCCACTTTGGCGGCACTTGAGGAG CCCTTCAGCCCACCACTGCACTGCGGAGCCCCCTTTCTGGGCTGGCCAAGGCCGGAGCCGGCTCCCTCAGCTTGCAGGGA GCTTGGCGGGCCCCGCACTCGGAGCAGCCGGCCAGCCCTTCCAGCCCCAGGCAATGAGAGGCTTAGCACCCGGGCCAGCA GCTGCGGAGGGTGTACTCCGTCCCCCAGCAGTGCCAGCTCACAGGCGCTCCAATTTCTCACCGGGCCTTAGCTGCC TTCGCGCGGGGGGTGCTCGGGACCTGCAGCCCGCCATGCCTGAGCTCCCACCCCTCCATGGGCTCCCGTGCGCCCGAGC CTCCCCGATGAGCACCACCCCCTGCTCCACGGCGCCCAGTCCCATCGACCACCCAAGAGCTGAGGAGTGCGGGCGCACGG CGCGGGACTGGCAGCTCCACCTGCAGCTCTCGTGCGGGATTCACTGGGGGAAGCCAGCTGGGCTCCTGAGTCTGGT GGGGACGTGGAGAACCTTTATGTCTAGCTCAGGGATTGTAAATACACCAATCGGCACTCCGTATCTAGCTCAAGGTTTGT AAACACACCAATCAGCACCCTGTGTCTAGCTTAGTGTTTGTGAACGCACCAAGCCACACTCTGTATCTAGCTACTCTGGT GGGGCTTTGGAGAACCTTTGTGTCCACACTCTGTAGCCAGCTAATCTGGTGGGGACATGGAGAACCTTTGGGTGTAGCTC AGATAAGAGCATAAAAGCAGGCTGCCTGAGCCAGCAGTGGCAACCCGCTTGGGTTCCCTTCCACACTGTGGAAGGTTTGT ACTCCGAACACATCCGAACATCAGAAGGAACAAACTCCAGATGCGCCACATTAAGAGCTGTAACACTCACCGCGAGGGTC CCTGGCTTCATTCTTGAAGTCAGTGAGACCAAGAACCCACCAATTTTGGACACAGTTTGACAATAAATTTACACTCAAAT ATCTCTAAGGAATCAAACTTACAGATTAATAATTAGTAATCAGGTCACGTAAAGTAAATTATAAAAGAGCATTGATACCA AGATTGGCAGAAAGTTTTTTGTGTGACAAAACCAAGTTTTGGCTAAGATACACACTGCTGATGGGAGTCTAAATTGCTGT TCTAGTGAAACTCTTGAACGTGTGCGTCCAAAGACATTTATAAACATGATCTTAGTAGTATTGCTTTTAGTAGCAAATTC TGGAAACATCCCAAATGTCTATCAATAGTGGAATTGATTTGAAAGGGGTGTGGAATGGTAATATAATGGAATAGCCTACA GCTGTTTAGATAAAGGAACTCCAATTAAACATACCAACAAAGATACATTTCAAAAACAAGACGTTGAAAGGAAAAAAGTC ATCAAAACAATACACAACATTCTACCACATTTTTATAAATTCTCAAAATATGCAATATTAAACATGCATTATTTAGGGAG GCATTCAATGTAGCAATGCATTTTTAAGAGGCTGGGATGATAAATGTAAAATTCAGAACAGGTATTATCTCTGGGAACAG GAAGAGGAGGATGCAGTGTTGGGAAGAAATACATATAAGTACAGCAGTAGAGGCAGACTTTTTTTCCTTTTTCC CTTTATTTTTCCTAGCTTTCTTTTAGCTATGGTATTTCTTTAGCTATGGTATGGTATGGTATGGTATGTACTTTCCATAT GGTACATGTGCAGATTTGTTATATAGGTAAACTTGTGTCATGTGGTTTTGTTGCACAAATTATTTTCTCACCCAGGTATT TTTCTGTTTCTGCGTTAGCTTGCCAAGGATAATTGCCTCCAGCTCCATATTCCTGCAAAAGACATAATTTTGTTCCTTTA TATGGTTGCATAGTATTCCATGGTGTATATGTACCGCATTATCTTTAGCCAGTCTATCATTGATGAGCACTTAGGTTGAT TCCATGTCTTCGCTCTTAACATTTTTAAACAGTCTCTGAGTAGAATAGGGTAGGCTGGTGTAAGGAATTACTGTTTTTAA ATTTCTGGGAAGATTTGCAAGAATCTGTGGCAGTTGAGAGTAGGTTCACTTTCGCTTTATTGTGTAATTTATTGTATTTT TCATTTAATTGTACTTTGTAAACTAAATATTTATTGTATATTTTACTTCATTTTTTAATTGCCATATGCAGCTTTAATTT GAGGGGCGGAGGCGGAGGCGGGGGCGGAGGCGGAGGCGGAGGCGGAGGCGGAGGCGGAGGCGGAGGCGGAGGCGGAGGCG GAGGCGAGCACCGCCCCACCGCGCACGCTGCGGTTGCCCCGGCAGCCGCGCCCTGCGTGGCGGAAGCTCACAATCAGCCC GGTCCCTCCGGCTTCCACCCCGCCCCTGCGCTCACCTGCCCGCGCGCTCGCCTTCCGGGGACCCGGGGCCCATGGACAC ATACACCCAGCCCTGCTGTCCCGCGCGCCAGCTCACCAGCCCTACCCAAGGGACATCATTCACGCCTGGGCGCCTCCGCC GGGCTCCGGGAGCCCAAGGTCGCGGCTGGGCCAGCGCTGAGCGTCAGAGGACGAGAGCAGGGGCCTCCCCGGTCGCCCCA

@sample006.000000000

HHHIHIIHHHIHIGHGIIIIGHHHIGEGIFIIGFHIFGIIHGIDGIDIGGHGFIIIIIICIHEIIIIIIHDIIBIIGH@IIBGIBIEIHIIBIIIIGII @sample006.000000001

 CATGTCCTTGTGGAAAGCAGGTAAACTTTAGAAAATTATAAGGAAGCAGAGGAAAGTAACTACAATCCCACCACCACTCAAAAACTTTTAAAAAGTTAACATT

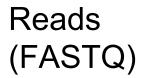
HHIHHIIIIIIGHHIIIGHHIFIIIIFHGHHIIGIHIHIIHIIIIFIIIDAIDIIFHIDIEACCIIIFHD=IHICIIEHFIIIIECIIIICDBBIIIFD9

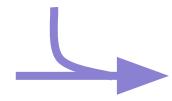
ÄAAGAATATGATGTTCTCTTGTAATTAGCAAGGGCTGATAGGGTTCAAGAGGTGACACTGATATGGATCTTGAGTAACTGTTGACAAATACTTCCCTTTA

##seed=1373972756

##reference=chr0.fa

##source=mutatrix population genome simulator





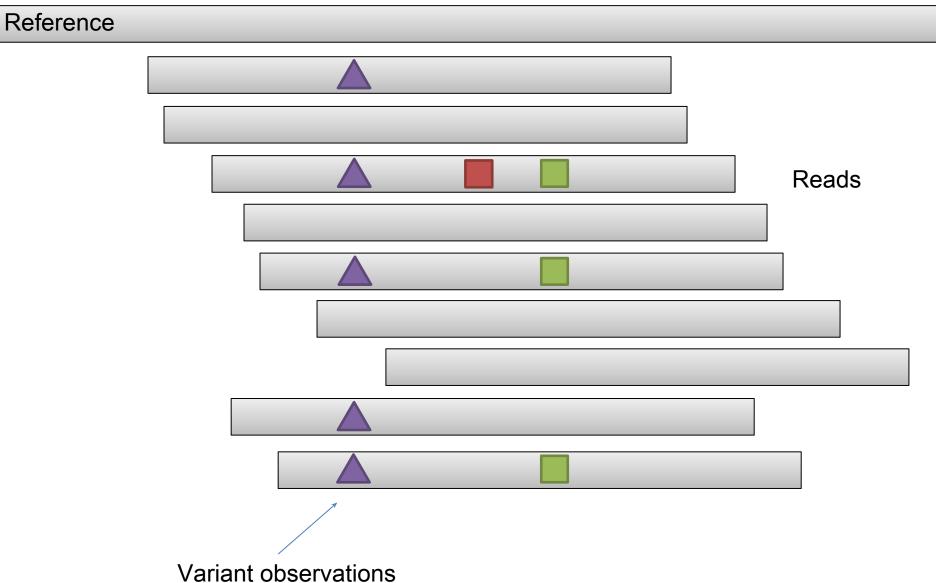
alignment and variant calling

##phasi	na=true												
	#commandline=mutatrix -S sample -p 2 -n 100 chrQ.fa												
##filter="AC > 0"													
	INFO= <id=type.number=a,type=string,description="type (snp,="" allele="" complex)"="" del,="" each="" ins,="" mnp,="" of=""></id=type.number=a,type=string,description="type>												
	NFO= <id=na.number=1.type=integer.description="number alleles"="" alternate="" of=""></id=na.number=1.type=integer.description="number>												
	NFO= <id=len,number=a,type=integer,description="length allele"="" alternate="" each="" of=""></id=len,number=a,type=integer,description="length>												
									repeat loci">				
		Number=						sequence i	repeat toci >				
								1441	lleles in called				
											>		
									the range (0,1]'	>			
							of samples						
##INFU= #CHROM		umber=1, ID	REE=Int	ALT	OUAL OUT				called genotypes				
						FILTER		RMAT sampl			CT	010	110
chrQ	1252		C	A	99				N=1; NA=1; NS=2; TYF		GT	010	1 0
chrQ	3646		Ţ	TC	99				N=1; NA=1; NS=2; TYF		GT	010	0 1
chrQ	6283		C	Ţ	99				N=1; NA=1; NS=2; TYF		GT	010	0 1
chrQ	7412		C	Ţ	99				N=1; NA=1; NS=2; TYF		GT	010	1 0
chrQ	7935		Ţ	C	99				N=1;NA=1;NS=2;TYF		GT	010	1 0
chrQ	8131		T	C	99				=1;NA=1;NS=2;TYPE		GT	0 1	0 1
chrQ	8682		AA	TG	99				N=2;NA=1;NS=2;TYF		GT	010	0 1
chrQ	10926		T	C	99				N=1;NA=1;NS=2;TYF		GT	0 0	0 1
chrQ	11921		G	GTT	99				N=2;NA=1;NS=2;TYF		GT	0 1	0 0
chrQ	12955		T	G	99				N=1;NA=1;NS=2;TYF		GT	010	1 0
chrQ	13808		Ţ	TG	99				V=1;NA=1;NS=2;TYF		GT	1 0	0 0
chrQ	15271		A	G	99				N=1; NA=1; NS=2; TYF		GT	0 1	0 0
chrQ	15407		A	C	99				N=1;NA=1;NS=2;TYF		GT	1 0	9 9
chrQ	16486		C	G	99				N=1; NA=1; NS=2; TYF		GT	1 0	9 9
chrQ	16563		T	A	99				N=1;NA=1;NS=2;TYF		GT	0 1	9 9
chrQ	16748		GTT	G	99				N=2; NA=2; NS=2; TYF		GT	0/0	0/1
chrQ	17697		G	C	99				N=1; NA=1; NS=2; TYF		GT	010	0 1
chrQ	19568		A	G	99				N=1; NA=1; NS=2; TYF		GT	0 1	0 0
chrQ	20750		G	A	99				N=1;NA=1;NS=2;TYF		GT	010	1 0
chrQ	21532		Ţ	C	99				N=1; NA=1; NS=2; TYF		GT	1 0	9 9
chrQ	22291		C	Ţ	99				N=1; NA=1; NS=2; TYF		GT	0 1	9 9
chrQ	23193		G	A	99				N=1;NA=1;NS=2;TYF		GT	010	0 1
chrQ	23954		CTAA	TTAA	99				1=4; NA=2; NS=2; TYF		GT	0/0	0/1
chrQ	24467		C	Ţ	99				N=1; NA=1; NS=2; TYF		GT	0 0	0 1
chrQ	26100		G	A	99				N=1;NA=1;NS=2;TYF		GT	0 1	9 9
chrQ	29654		T	A	99				N=1;NA=1;NS=2;TYF		GT	1 0	0 0
chrQ	30062		Ţ	C	99				N=1; NA=1; NS=2; TYF		GT	0 1	0 0
chrQ	31790		A	G	99				N=1;NA=1;NS=2;TYF		GT	0 1	9 9
chrQ	32792		T	C	99				N=1;NA=1;NS=2;TYF		GT	1 1	0 1
chrQ	33376		CC	C	99				=1;NA=1;NS=2;TYPE		GT	1 0	0 1
chrQ	33403		Ţ	C	99				=1;NA=1;NS=2;TYPE		GT	0 1	0 1
chrQ	33802		A	G	99				=1;NA=1;NS=2;TYPE		GT	0 1	1 0
chrQ	34450		C	T	99				;NA=1;NS=2;TYPE=		1 1	1 1	
chrQ	34716		G	A	99				N=1;NA=1;NS=2;TYF		GT	0 0	1 0
chrQ	35484		G	A	99				V=1;NA=1;NS=2;TYF		GT	0 0	1 0
chrQ	36547		G	Α	99				N=1;NA=1;NS=2;TYF		GT	0 0	0 1
chrQ	38015		T	A	99				V=1;NA=1;NS=2;TYF		GT	0 0	1 0
chrQ	38281		T	C	99				V=1;NA=1;NS=2;TYF		GT	0 1	0 0
chrQ	40467		A	G	99				N=1;NA=1;NS=2;TYF		GT	1 0	0 0
chrQ	40581		A	G	99				N=1;NA=1;NS=2;TYF		GT	0 0	1 0
chrQ	40601		A	T	99				N=1;NA=1;NS=2;TYF		GT	010	0 1
chr()	43768		G	A	QQ		AC=1:AF=0.	25:AN=4:1 FN	U=1:NA=1:NS=2:TYI	PF=snn	GT	011	919

Genome (FASTA)

Variation (VCF)

Finding variation



Single-position calling

Reference

Haplotype information is lost.

INDEL normalization

Left alignment allows us to ensure that our representation is consistent across alignments and also variant calls.

CGTATGATCTAGCGCGCTAGCTAGCTAGC CGTATGATCTA - - GCGCTAGCTAGCTAGC Left aligned

CGTATGATCTAGCGCGCTAGCTAGCTAGC CGTATGATCTAGC - - GCTAGCTAGCTAGC

CGTATGATCTAGCGCGCTAGCTAGCTAGC CGTATGATCTAGCGC - -TAGCTAGCTAGC example: 1000G Phasel low coverage, chr15:81551110, ref:CTCTC alt:ATATA

ref: TGTCACTCGCTCTCTCTCTCTCTCTATATATATATATTTTGTGCAT

alt: TGTCACTCGCTCTCTCTCTATATATATATATATATATTTTGTGCAT

Interpreted as 3 SNPs

ref: TGTCACTCGCTCTCTCTCTCTCTCTCT----ATATATATATATTTGTGCAT alt: TGTCACTCGCTCTCTCTCTCT----ATATATATATATATATTTGTGCAT

Interpreted as microsatellite expansion/contraction

Complex allele realignment

example: 1000G Phasel low coverage, chr20:708257, ref:AGC alt:CGA

Finding haplotype polymorphisms

Two reads

AGAACCCAGTGCTCTTTCTGCT AGAACCCAGTGGTCTTTCTGCT

a SNP

Another read showing a SNP on the same haplotype as the first

AGAACCCAGTG CTCTTTCTGCT

AGAACCCAGTGCTCTATCTGCT

Their alignmen t

AGAACCCAGTG CTCTA GTCTT TCTGCT A variant locus implied by alignments

Direct detection of haplotypes (FreeBayes)

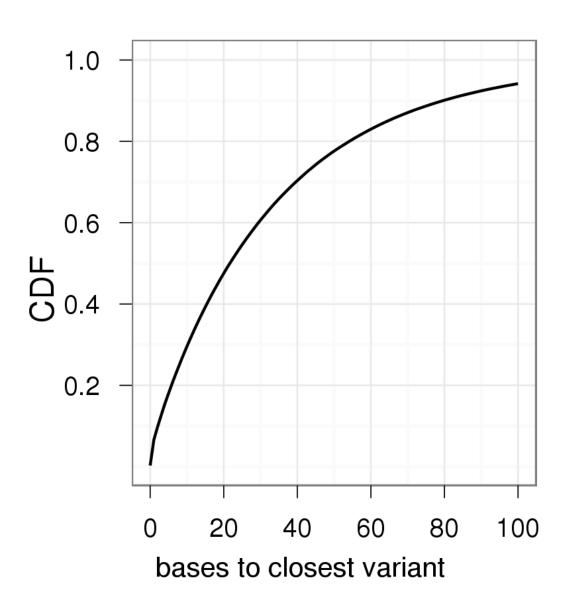


Variant Variant Region Region Ref **TACCGAT** CATTGGATCA CGATTCC...GCATTGC AAAAAA-**GACCGCA** CATTGGATCA TACCGAT CGATTCC...GCATTGC -AAAAA-**GACCGCA** ACCGAT TATTGCATCG CGATTCC...GCATTGC -AAAAAA-**GACCGCA** ds CGATTCC...GCATTGC ACCGAT CATTGGATCA AAAAAA-A **GACCGCA** Rea ACCGAT TATTGGATCG CGATTCC...GCATTGC -AAAAAAA **GACCGCA** CCGAT C-TTGGATCA CGATTCC...GCATTGC AAAAAA-**GACCGCA** CGATTCC...GCATTGC CATGGGATCA AAAAAAA CCGAT **GACCGCA** x10 **8**x CATTGGATCA $(A)_7$ **x9 x**7 $(A)_6$ **TATTGGATCG** x1x1 $(A)_5$ **CTTGGATCA** x1 x1CATGGGATCA (A)₈

Why haplotypes?

- Variants cluster.
- This has functional significance.
- Observing haplotypes lets us be more certain of the local structure of the genome.
- We can improve the detection process itself by using haplotypes rather than point mutations.

Sequence variants cluster



In ~1000 individuals, ½ of variants are within ~22bp of another variant.

Variance to mean ratio (VMR) = 1.4.

The functional effect of variants depends on other nearby variants on the same haplotype

reference: AGG GAG CTG

Arg Glu Leu

OTOF gene – mutations cause profound recessive deafness

apparent: AGG TAG CTG

Arq Ter ---

Apparent nonsense variant, one YRI homozygote

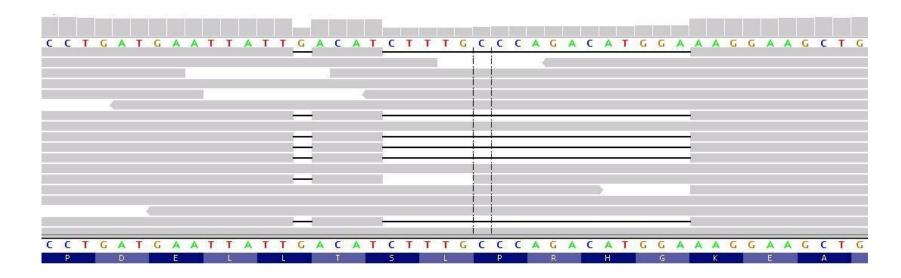
actual: AGG TTG CTG

Arg Leu Leu

Actually a block substitution that results in a missense substitution

(Daniel MacArthur)

Importance of haplotype effects: frame-restoring indels



- Two apparent frameshift deletions in the CASP8AP2 gene (one 17 bp, one 1 bp) on the same haplotype
- Overall effect is in-frame deletion of six amino acids

Frame-restoring indels in 1000 Genomes Phase I exomes

chr6:117113761, GPRC6A (~10% AF in 1000G)

ref: ATTGTAATTCTCA--TA--TT--TGCCTTTGAAAGC

alt: ATTGTAATTCTCAGGTAATTTCCTGCCTTTGAAAGC

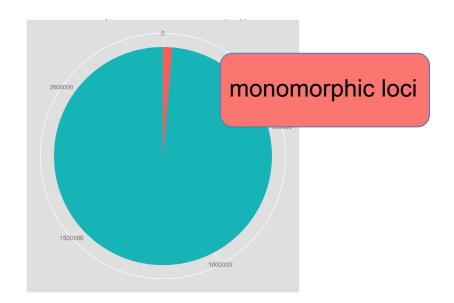
chr6:32551935, HLA-DRB1 (~11% AF in 1000G)

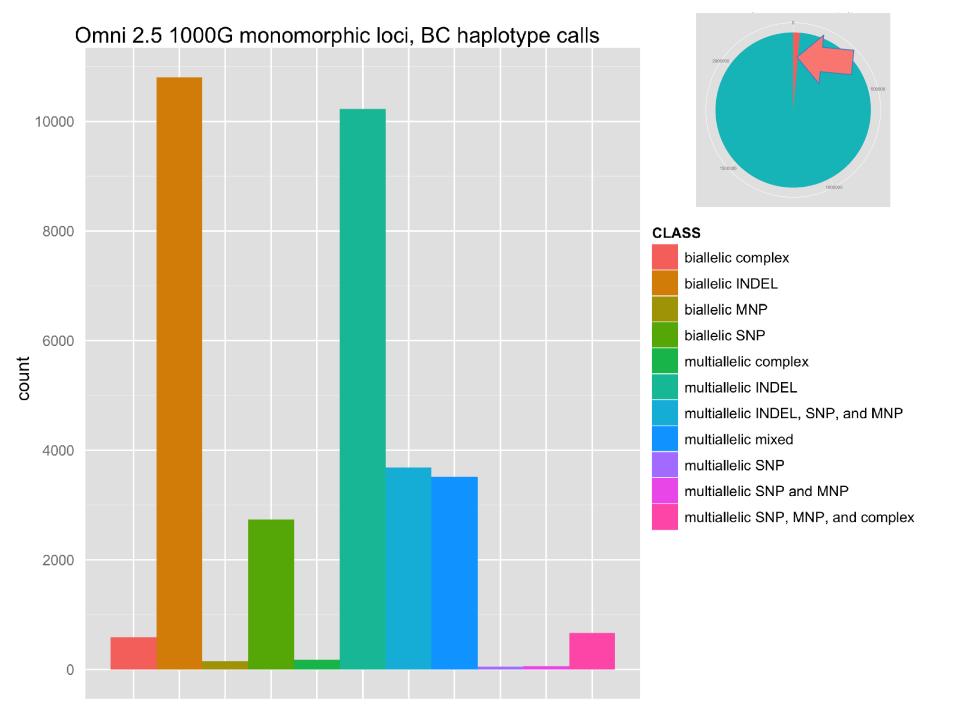
ref: CCACCGCGCCCCGCGCCTG-C-TCCAGGATGTCC

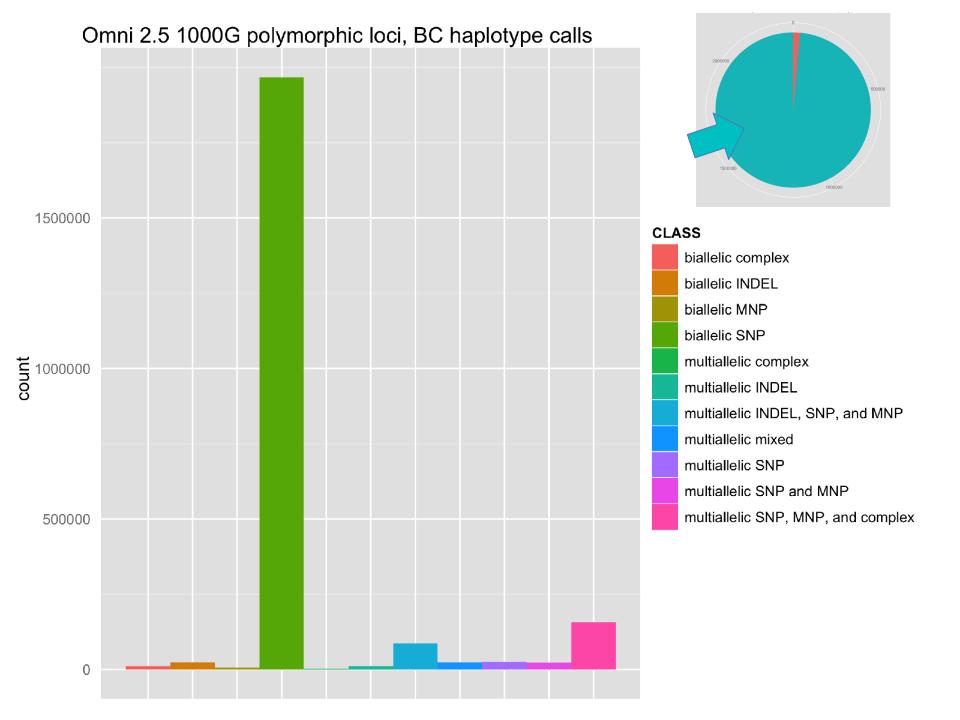
Alt: CCACCGCGG--CGCGCCTGTCTTCCAGGAGGTCC

Impact on genotyping chip design

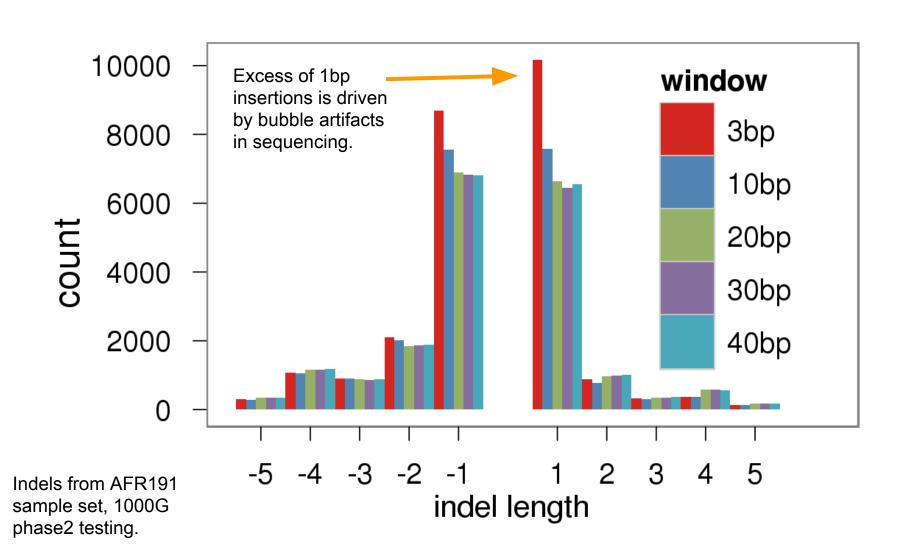
- Biallelic SNPs detected during the 1000 Genomes Pilot project were used to design a genotyping microarray (Omni 2.5).
- When the 1000 Genomes samples were genotyped using the chip, 100k of the 2.5 million loci showed no polymorphism (monomorphs).



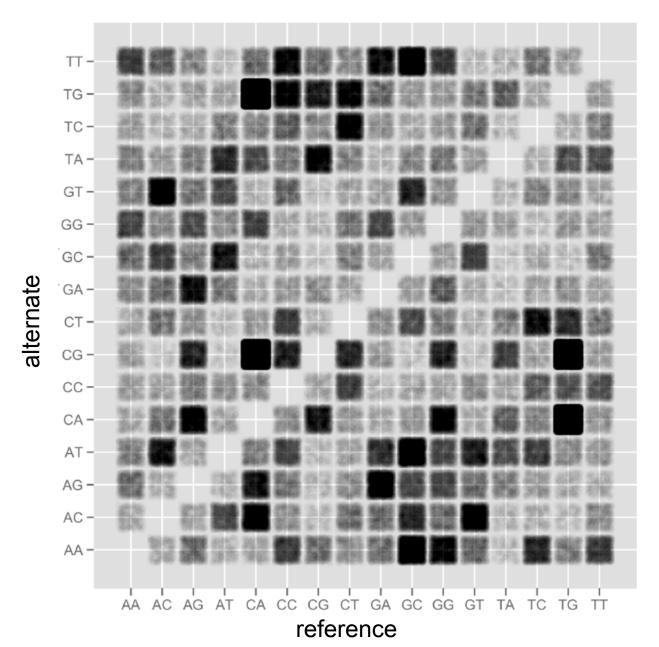




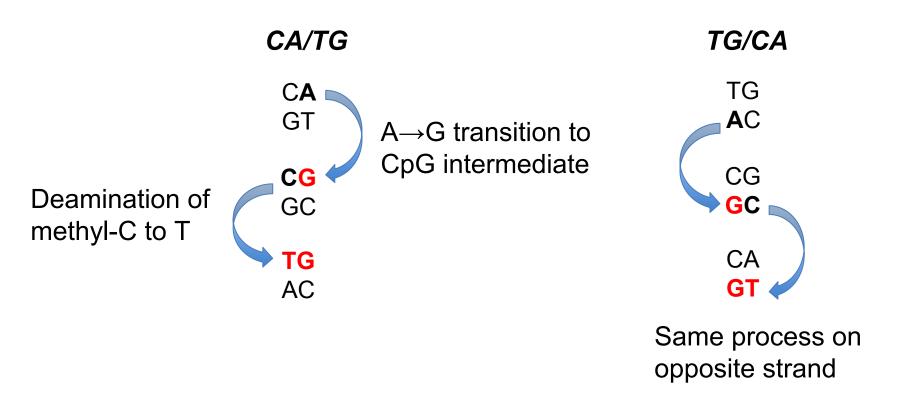
Measuring haplotypes improves specificity



2bp MNPs and dinucleotide intermediates



A distinct mutational mechanism is responsible for the most frequent 2bp MNP



Filtering INDELs

As with SNPs, sequencing error rates are high.

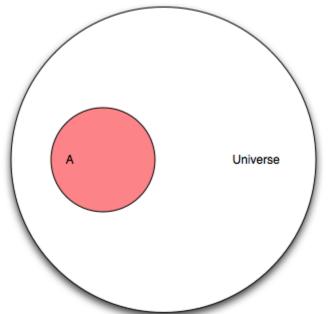
So, we need to filter.

The standard filter of NGS is the Bayesian variant caller.

Combines population-based priors and data from many samples to make high-quality calls.

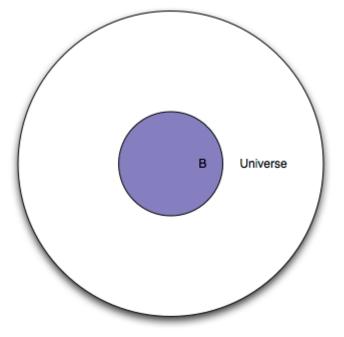
Bayesian (visual) intuition

We have a universe of individuals.



A = samples with a variant at some locus

B = putative observations of variant at some locus



probability(A|B)

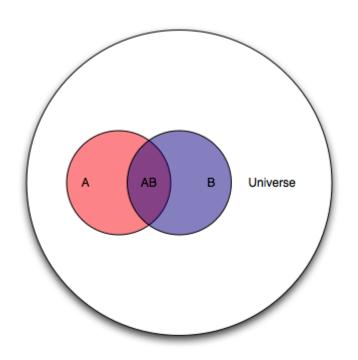
We want to estimate the probability that we have a real polymorphism "A" given "|" that we observed variants in our alignments "B".

$$P(A|B) = \frac{|AB|}{|B|}$$

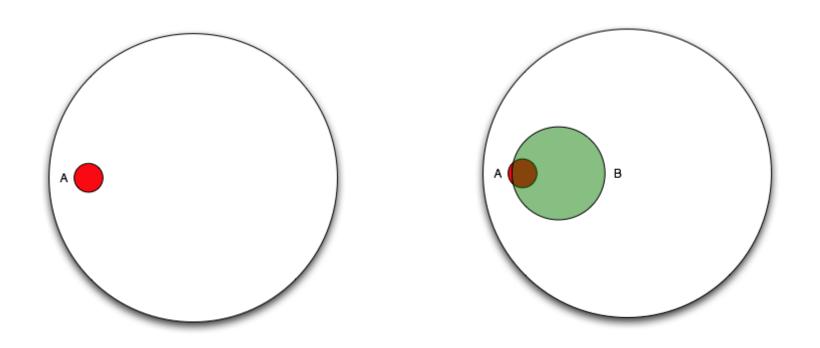
$$P(A|B) = \frac{P(AB)}{P(B)}$$

$$P(B|A) = \frac{P(AB)}{P(A)}$$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

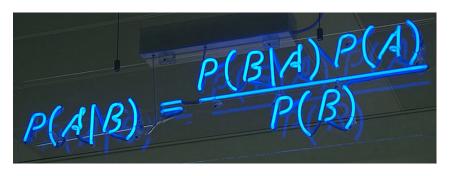


In our case it's a bit more like this...



Observations (B) provide pretty good sensitivity, but poor specificity.

The model



- Bayesian model estimates the probability of polymorphism at a locus given input data and the population mutation rate (~pairwise heterozygosity) and assumption of "neutrality" (random mating).
- Following Bayes theorem, the probability of a specific set of genotypes over some number of samples is:
- Which in FreeBayes we extend to:

 - G = genotypes, R = reads, S = locus is wellcharacterized/mapped
 - P(R|G,S) is our data likelihood, P(G) is our prior estimate of the genotypes, P(S) is our prior estimate of the mappability of the locus, P(R) is a normalizer.

Handling non-biallelic/diploid cases

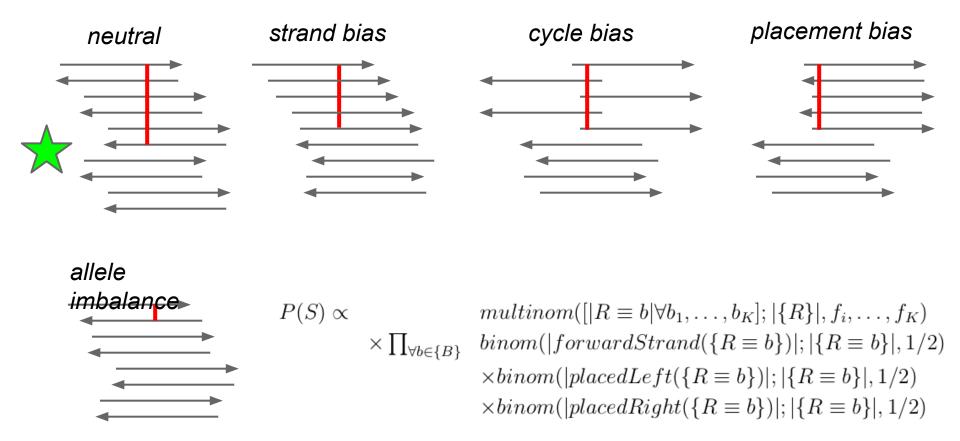
We compose our data likelihoods, **P(Reads|Genotype)** using a discrete multinomial sampling probability:

$$\begin{split} P(reads|genoytpe) = & \begin{pmatrix} |reads| \\ |reads = A|, |reads = B| \ldots \end{pmatrix} \\ & \times & \prod_{\forall alleles \in genotype} freq(allele \in genotype) \\ & \times & \prod_{\forall reads} P(correct(read)) \end{split}$$

Our priors, **P(Genoypes)**, follow the Ewens Sampling Formula and the discrete sampling probability for genotypes.

Are our locus and alleles sequenceable?

In WGS, biases in the way we observe an allele (placement, position, strand, cycle, or balance in heterozygotes) are often correlated with error. We include this in our posterior **P(G,S|R)**, and to do so we need an estimator of **P(S)**.



The detection process

- Parse alleles (small haplotypes) from alignments using CIGAR strings.
- 2. Pick suitable alleles (very weak input filters to improve runtime).
- Build haplotypes across target locus.
- 4. Generate genotype likelihoods.
- Sample a posterior space around the data-likelihood maximum
 - a. update genotype estimates and iterate (hill-climbing posterior search) until convergence on maximum a posteriori genotyping over all samples
- 6. Output a record, and do it again

SVM filtering

INDEL detection is hard.

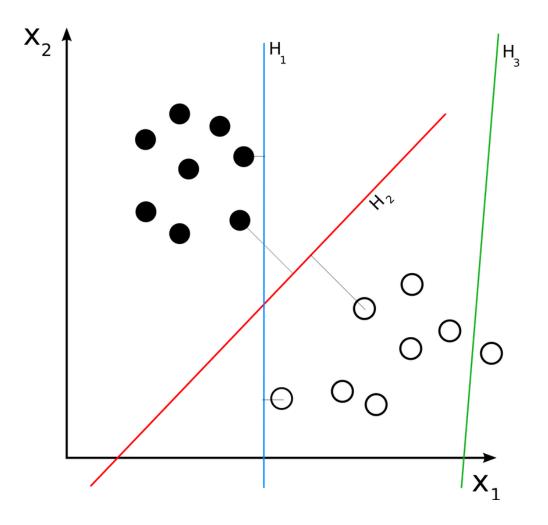
A priori models can't capture all types of error.

It's especially difficult when we try to make a consensus set from lots of input variant callers.

We can use classifiers like Support Vector Machines (SVM) to further improve results.

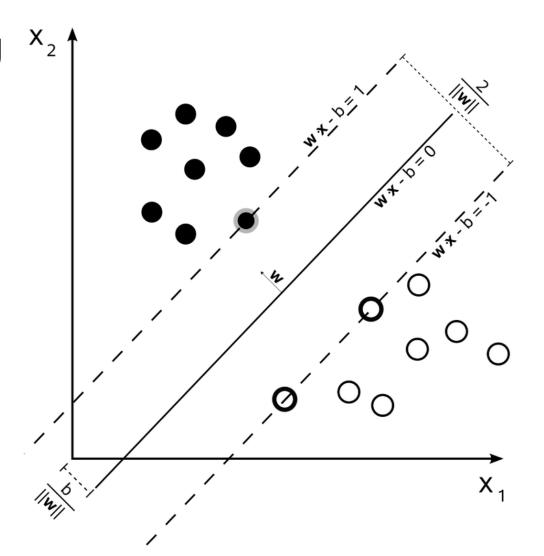
SVM classifier

Find a hyperplane (here a line in 2D) which separates observations.



SVM classifier

The best separating hyperplane is determined by maximum margin between groups we want to classify.



SVM approach for INDEL filtering

Extract features that tend to vary with respect to call quality:

- call QUALity
- read depth
- sum of base qualities
- inbreeding coefficient (heterozygosity)
- entropy of sequence at locus
- mapping quality
- allele frequency in population
- read pairing rate
- etc.

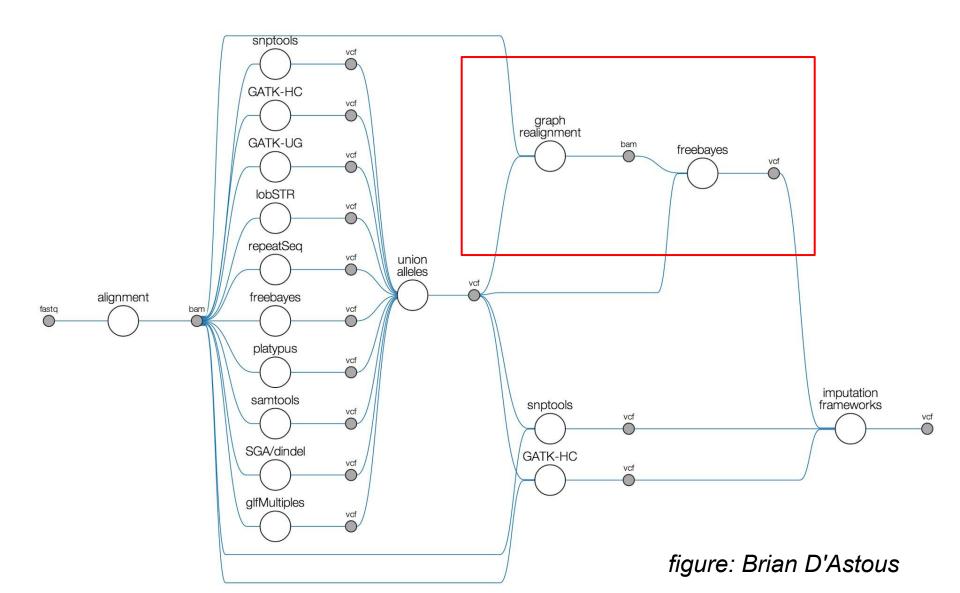
SVM approach for INDEL filtering

Now, use overlaps in validation samples or sites to determine likely errors and true calls.

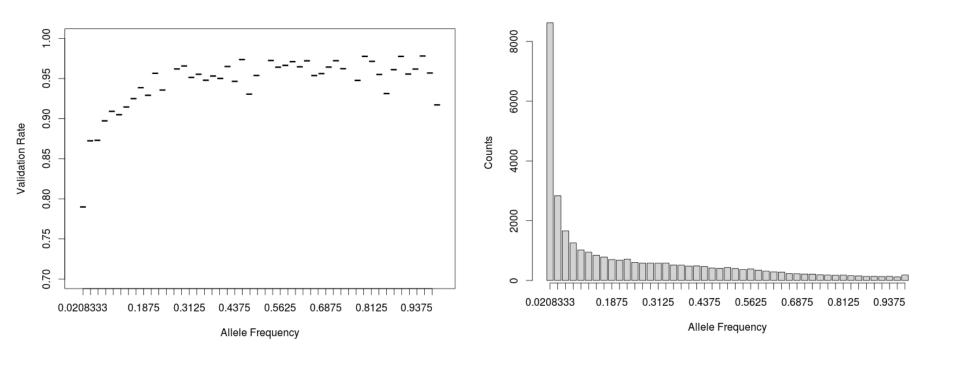
Use this list + annotations of the calls to train an SVM model.

Apply the model to all the calls, filter, and measure validation rate of the whole set.

1000G variant integration pipeline



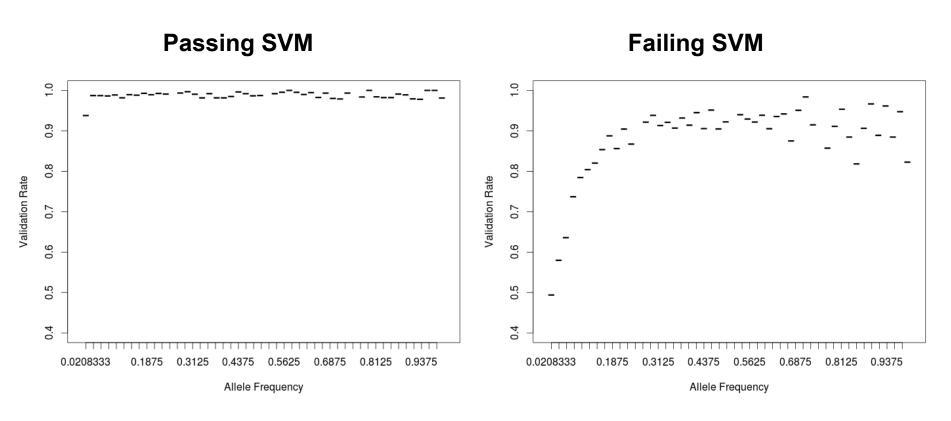
Application of SVM to 1000G INDELs



Raw validation rates of indels in 1000G phase 3, "MVNCall" set.

Tony Marcketta and Adam Auton

Application of SVM to 1000G INDELs



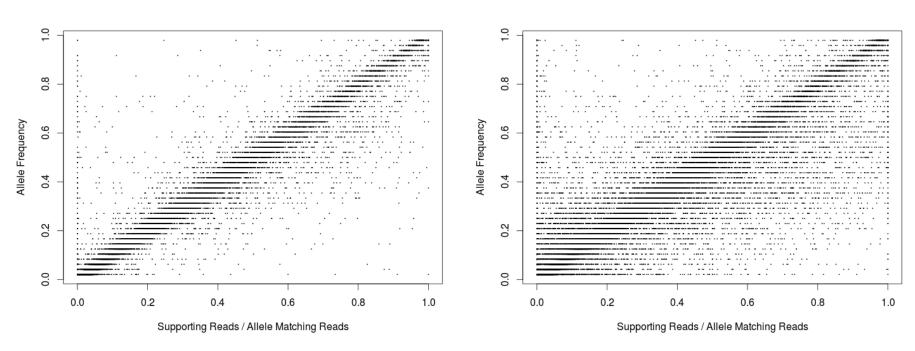
Filtering results, using SVM-based method.

Anthony Marcketta and Adam Auton

Application of SVM to 1000G INDELs

Passing SVM

Failing SVM



Correlation between allele frequency and observation counts.

Anthony Marcketta and Adam Auton

Questions?

. . .