Detecting SNVs with Next-generation-Sequencing

Johannes Köster
Genome Informatics, University Duisburg-Essen

February 25, 2014

Outline

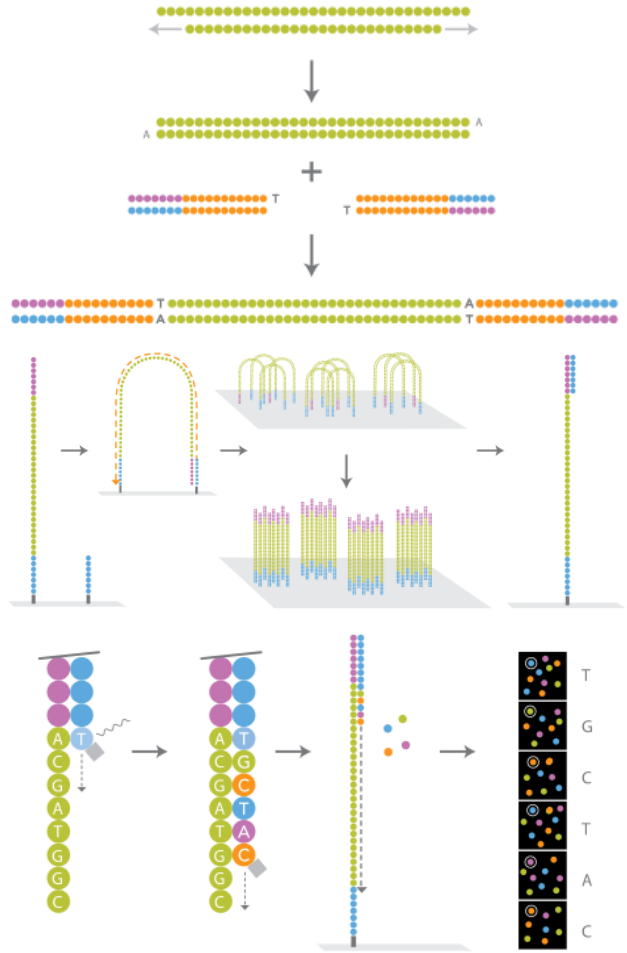
Definitions

Reference	a consensus DNA sequence
Variant	a local difference to the reference in the sequenced sample
SNV	single nucleotide variant
Indel	an insertion or deletion
SNP	single nucleotide polymorphism (a variant that recurs in a population)
Allele	One of all occuring variants at a specific locus

Outline

Next-Generation-Sequencing

1. DNA/RNA is chopped into small fragments.
2. Ligate adapters to both ends.
3. PCR amplify fragments to obtain enough material.
4. Spread fragment solution across a carrier (flowcell) with beads.
5. Amplify fragments into clusters with PCR.
6. Fragments are sequenced from one or both ends by adding fluorescent complementary bases $\rightarrow$ reads.



Illumina, 2013

Detecting SNVs with Next-Generation-Sequencing

Idea:

- Get the difference (in terms of variants) of the sequenced individuum to a reference genome.

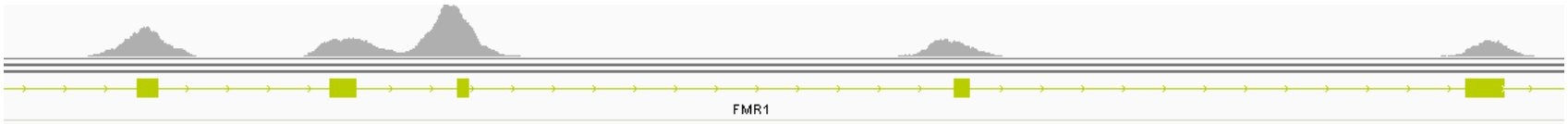
Problems:

- huge amount of small reads, need to find their origin in the reference genome
- PCR-duplicates
- expected sequencing error-rate of 2% $\rightarrow$ genetic variant or error?

Detecting SNVs with Next-Generation-Sequencing

How to distinguish genetic variation from technical errors?

- assuming technical errors are mostly random...
- sequence each portion of the genome as often as possible $\rightarrow$ sequencing depth
- variants common in many reads can be considered true
- sequencing the whole genome at sufficient depth is expensive $\rightarrow$ concentrate on coding regions (exome sequencing)



- missing something? 85% of disease-causing mutations are expected in the exome (Antonarakis et al. 1995).

Outline

Workflow

1. read mapping
2. post-processing reads
3. variant calling
4. variant filtering

- Steps are handled by invoking command line tools on various files.
- The workflow is managed by Snakemake (Köster and Rahmann 2012).
- The variants are called by GATK (DePristo et al. 2011).
- The variants are stored and analysed with Exomate (Martin et al. 2013).

Snakemake

Problem:

- Bioinformatics analyses usually involve many steps.
- Often, command line utilities are used to create output files from input files.
- Want to avoid redoing work by hand when new samples arrive or parameters change.

Our solution:

- Snakemake, a text-based workflow management system.
- Snakemake allows to plug the steps together by matching filenames with wildcards.
- Upon changes/additions of samples, necessary parts of the workflow can be automatically recomputed.

Snakemake

```
SAMPLES = ["500", "501", "502", "503"]
rule all:
  input: expand("{sample}.bam", sample=SAMPLES)

rule sai_to_bam:
  input:  "hg19.fasta", "{sample}.sai", "{sample}.fastq"
  output: "{sample}.bam"
  shell:
    "bwa samse {input} | samtools view -Sbh - > {output}"

rule map_reads:
  input:  "hg19.fasta", "{sample}.fastq"
  output: "{sample}.sai"
  shell:  "bwa aln {input} > {output}"
```

Snakemake

```
SAMPLES = [
rule all:
  input: expand("{sample}.bam", sample=SAMPLES)

rule sai_to_bam:
  input:  "hg19.fasta", "{sample}.sai", "{sample}.fastq"
  output: "{sample}.bam"
  shell:
    "bwa samse {input} | samtools view -Sbh - > {output}"

rule map_reads:
  input:  "hg19.fasta", "{sample}.fastq"
  output: "{sample}.sai"
  shell:  "bwa aln {input} > {output}"
```

rule all
500.bam, 501.bam, 502.bam, 503.bam

Snakemake

```
SAMPLES = [
rule all:
  input: expand("{sample}.bam", sample=SAMPLES)

rule sai_to_bam:
  input:  "hg19.fasta", "{sample}.sai", "{sample}.fastq"
  output: "{sample}.bam"
  shell:
    "bwa samse {input} | samtools view -Sbh - > {output}"

rule map_reads:
  input:  "hg19.fasta", "{sample}.fastq"
  output: "{sample}.sai"
  shell:  "bwa aln {input} > {output}"
```

rule all
500.bam, 501.bam, 502.bam, 503.bam

rule sai_to_bam
500.sai

Snakemake

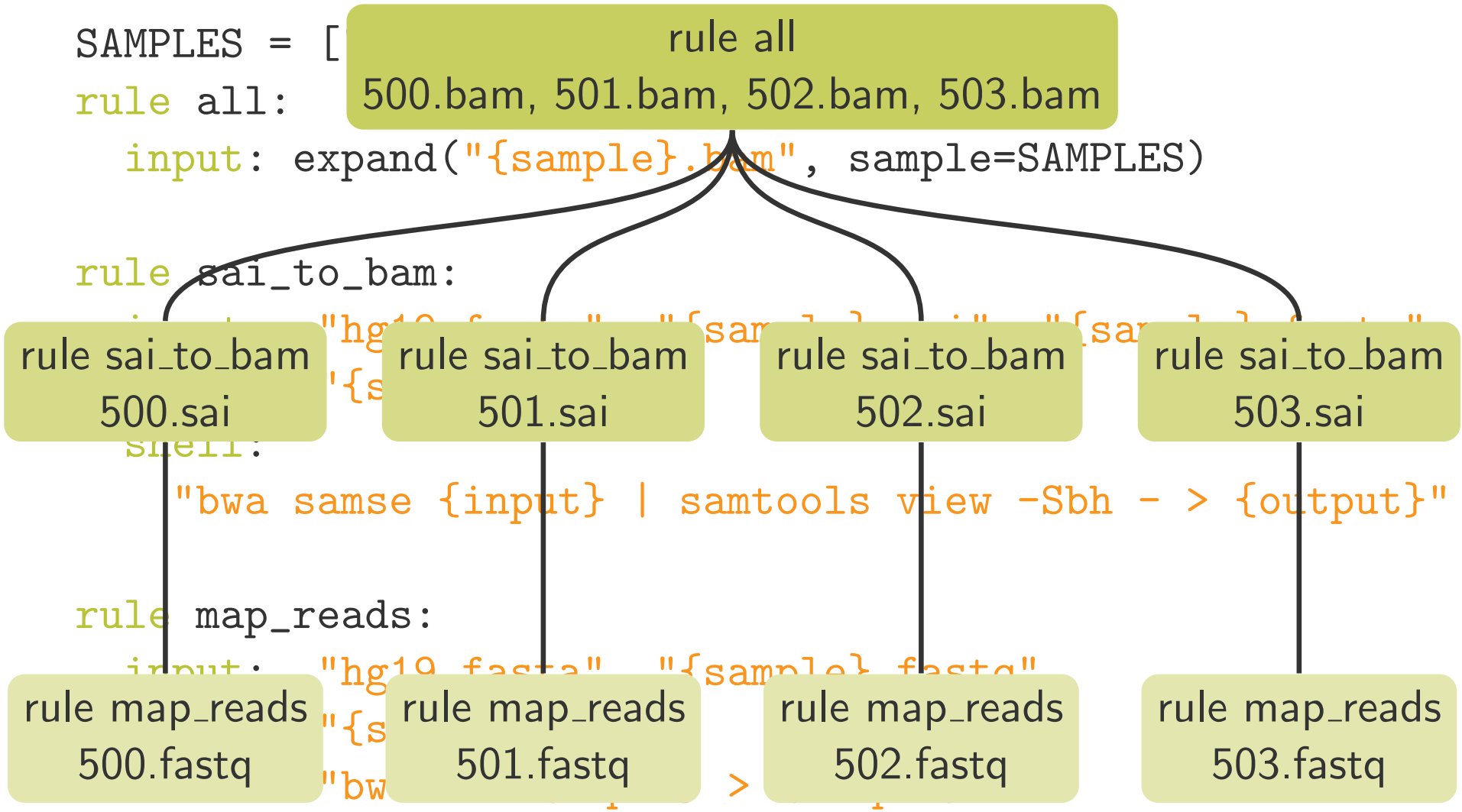
```
SAMPLES = [
rule all:
    input: expand("{sample}.bam", sample=SAMPLES)

rule sai_to_bam:
    input:  "hg19.fasta", "{sample}.sai", "{sample}.fastq"
    output: "{sample}.bam"
    shell:
        "bwa samse {input} | samtools view -Sbh - > {output}"

rule map_reads:
    input:  "hg19.fasta", "{sample}.fastq"
    output: "{sample}.sai"
    shell:  "bwa aln {input} > {output}"
```

rule all
500.bam, 501.bam, 502.bam, 503.bam

rule sai_to_bam
500.sai

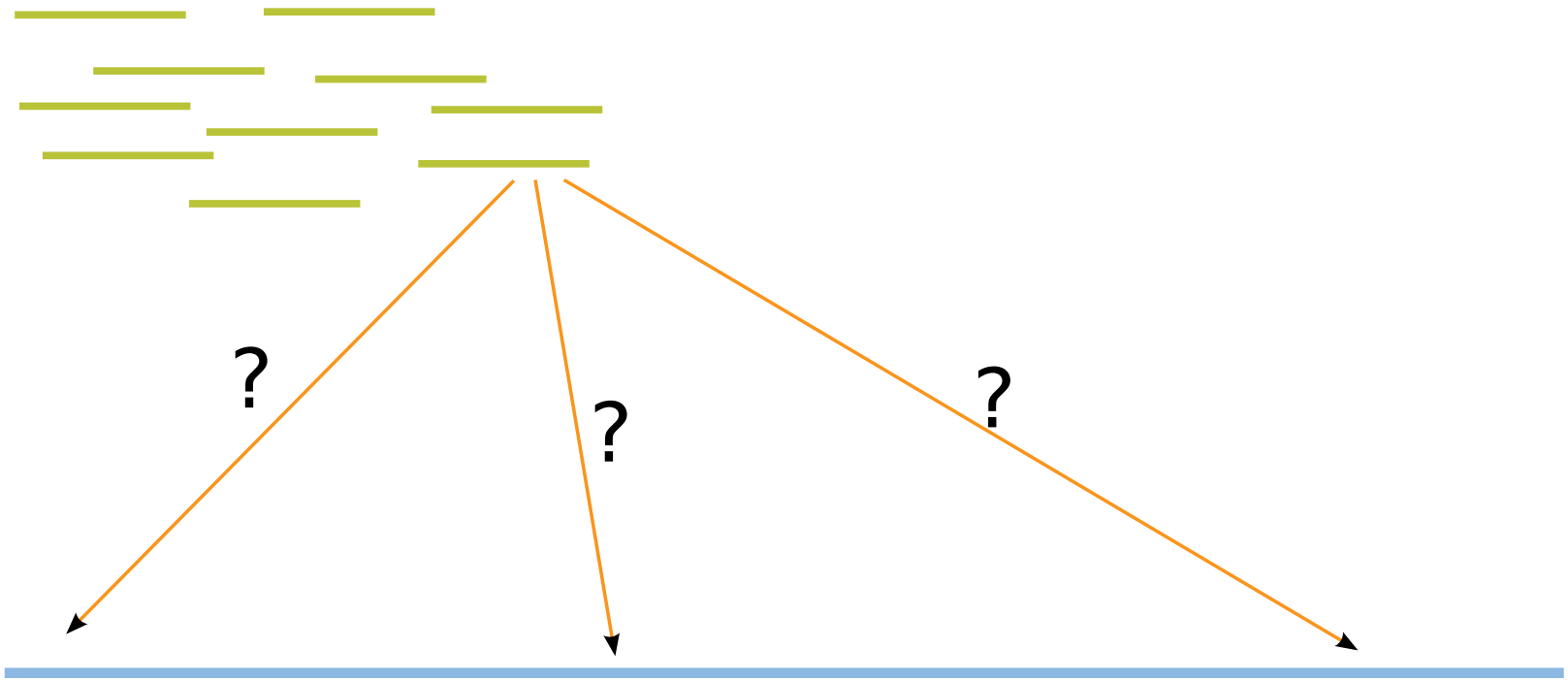
rule map_reads
500.fastq

Snakemake


SAMPLES = [
rule all:
input: expand("{sample}.bam", sample=SAMPLES)
rule sai_to_bam:
"bwa samse {input} | samtools view -Sbh - > {output}"
rule map_reads:
rule all
500.bam, 501.bam, 502.bam, 503.bam
rule sai_to_bam
500.sai
rule sai_to_bam
501.sai
rule sai_to_bam
502.sai
rule sai_to_bam
503.sai
rule map_reads
500.fastq
rule map_reads
501.fastq
rule map_reads
502.fastq
rule map_reads
503.fastq

Outline

Read Mapping



Read Mapping

For each read...
find position(s) with optimal alignment(s) to either strand of the reference:

ACTGTGGACTATCAATGGAC
GGTACTGT CTATCTATGGACCGTTAG

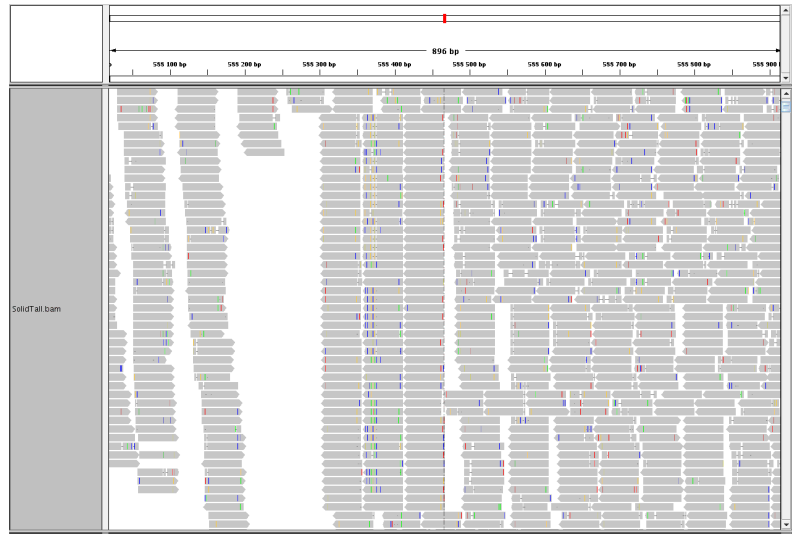
Too slow, therefore heuristics to find anchor positions:

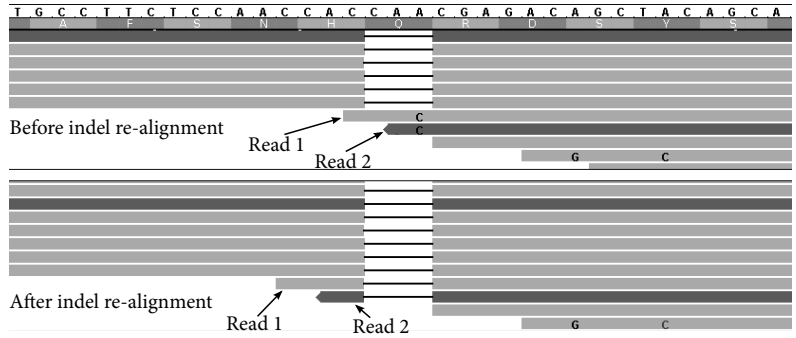
- suffixarray/Burrows-Wheeler-Transformation (BWA, bowtie2)
- q-gram indices
- hashing

Outline

Post-Processing Reads

- Remove PCR duplicates.
- Empirically recalibrate reported base qualities: dinucleotide context, platform specific covariates.
- Realign reads around indels (multiple alignment for all reads during read mapping is infeasible).





Outline

Genome Analysis Toolkit

The Genome Analysis Toolkit (DePristo et al. 2011) is a collection of tools around variant calling.

- Variant calling with GATKs UnifiedGenotyper.
- Calls multiple samples simultaneously.
- Given the reads at each genome position, estimates the probability of having only the reference allele.

Variant Calling

Given a genomic position we assume to have

- for each sample i a set of aligned bases D_i
- a reference allele A
- an alternative allele B

Probability to observe base $D_{i,j}$ under allele X:

$$P(D_{i,j}|X) = \begin{cases} 1 - \epsilon_{i,j} & \text{if } D_{i,j} = X \\ \epsilon_{i,j} \cdot c_{i,j} & \text{else} \end{cases}$$

$\epsilon_{i,j}$ miscall probability given the base-quality of $D_{i,j}$

$c_{i,j}$ probability of X being the true allele given that $D_{i,j}$ was miscalled (technology specific):

Illumina:

	A	C	G	T
A	-	0.58	0.17	0.25
C	0.35	-	0.11	0.54
G	0.32	0.05	-	0.63
T	0.46	0.22	0.32	-

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probabilty $P(D_{i,j}|X)$ to observe $D_{i,j}$ under allele X.

Probabilty to observe base $D_{i,j}$ under genotype $GT = XY$:

$$P(D_{i,j}|GT = XY) = \frac{P(D_{i,j}|X) + P(D_{i,j}|Y)}{2}$$

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probabilty $P(D_{i,j}|GT)$ to observe $D_{i,j}$ genotype GT.

Probability of aligned bases under genotype GT:

$$P(D_i|GT) = \prod_j P(D_{i,j}|GT)$$

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probability $P(D_i|GT)$ of aligned bases under genotype GT.

Probability of pileup D given that the B allele resides on $q = m$ chromosomes in total:

$$P(D|q = m) = \sum_{GT \in \Gamma_m} \prod_i P(D_i|GT_i)$$

q_i the number of B alleles in sample i (0,1,2)

Γ_q the set of genotype assignments such that $\sum_i q_i = q$

$q = 0: \ \{(AA, AA, \dots)\}$

$q = 1: \ \{(AB, AA, AA \dots), (AA, AB, AA \dots), \dots\}$

$q = 2: \ \{(AB, AB, AA \dots), (BB, AA, AA \dots), \dots\}$

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probability $P(D|q = m)$ of all aligned bases under allele frequency $q = m$.

General probabilty of having an allele frequency of $q = m$ for n samples (infinite-site neutral variation model):

$$P(q = m) = \begin{cases} \frac{\Theta}{m} & \text{if } m > 0 \\ 1 - \Theta \sum_{i=1}^{2n} \frac{1}{i} & \text{else} \end{cases}$$

Θ expected heterozygosity (probability to have a non reference allele)

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probability $P(D|q = m)$ of all aligned bases under allele frequency $q = m$.
- General probability $P(q = m)$ of allele frequency $q = m$.

Probability of allele frequency $q = m$ given the aligned bases:

$$P(q = m|D) = \frac{P(q = m)P(D|q = m)}{\sum_{m'} P(q = m')P(D|q = m')}$$

Variant Calling

So far:

- Aligned bases D_i, reference allele A, alternative allele B.
- Probability $P(q = m|D)$ of allele frequency $q = m$ given all aligned bases D.

Phred-scaled probability of having no alternative alleles at the site of interest

$$QUAL = -10 \cdot \log_{10} P(q = 0|D)$$

Consider sites as potentially variable if

- $QUAL \geq 50$ (i.e. p-value $\leq 10^{-5}$) for deep coverage
- $QUAL \geq 10$ (i.e. p-value ≤ 0.1) for shallow coverage

Example

AACTCG
ACTCGCTT
CTCGC
GAACTCGCT
GGAACACGCTTGACT

$$P(q=0) = 1 - 0.001 - \frac{1}{2}0.001 = 0.9985$$

$$P(D|q=0) = P(D_i|AA) = \prod_j \frac{2P(D_{i,j}|A)}{2} = \prod_j \epsilon_{i,j} c_{i,j}$$

$$P(D|q=1) = P(D_i|AT) = \prod_j \frac{P(D_{i,j}|A) + P(D_{i,j}|T)}{2} = \prod_j \frac{\epsilon_{i,j} c_{i,j} + 1 - \epsilon_{i,j}}{2}$$

$$P(D|q=2) = P(D_i|TT) = \prod_j P(D_{i,j}|T) = \prod_j (1 - \epsilon_{i,j})$$

$$P(q=0|D) = \frac{P(q=0)P(D|q=0)}{\sum_{m=0}^{2} P(q=m)P(D|q=m)} = 0.001 \quad \text{with } \epsilon_{i,j} = 0.02 \, \forall i,j$$

Example

GAACTCGCT
GGAACACGCTTGACT

$$P(q=0) = 1 - 0.001 - \frac{1}{2}0.001 = 0.9985$$

$$P(D|q=0) = P(D_i|AA) = \prod_j \frac{2P(D_{i,j}|A)}{2} = \prod_j \epsilon_{i,j} c_{i,j}$$

$$P(D|q=1) = P(D_i|AT) = \prod_j \frac{P(D_{i,j}|A) + P(D_{i,j}|T)}{2} = \prod_j \frac{\epsilon_{i,j} c_{i,j} + 1 - \epsilon_{i,j}}{2}$$

$$P(D|q=2) = P(D_i|TT) = \prod_j P(D_{i,j}|T) = \prod_j (1 - \epsilon_{i,j})$$

$$P(q=0|D) = \frac{P(q=0)P(D|q=0)}{\sum_{m=0}^{2} P(q=m)P(D|q=m)} = 0.96 \quad \text{with } \epsilon_{i,j} = 0.02 \, \forall i,j$$

Quality Covariates

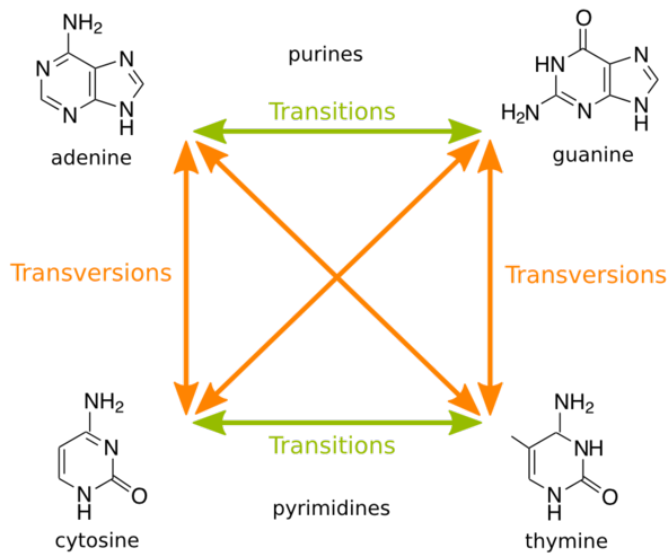
QUAL alone covers only part of the truth...

- base quality
- technology specific miscall rates

Systematic sequencing errors have to be measured with covariates, e.g.

strand bias	Is read strand independent of allele?
mapping qual	Is mapping quality independent of allele?
read positon	Is variant position in the read independent of allele?
haplotype score	Do the reads agree regarding induced haplotypes?

Estimating FDR



Wikipedia, 2013

Investigate the transition-transversion-rate (TiTv-rate).

- random variation (artifacts) would yield a rate of 0.5
- genetic variation is expected to yield a rate $TiTv_{exp} = 3.2$
- transitions are more likely synonymous

Hence, FDR can be estimated as

$$1 - \frac{TiTv_{obs} - 0.5}{TiTv_{exp} - 0.5}$$

Outline

Variant Filtering

SNV types:

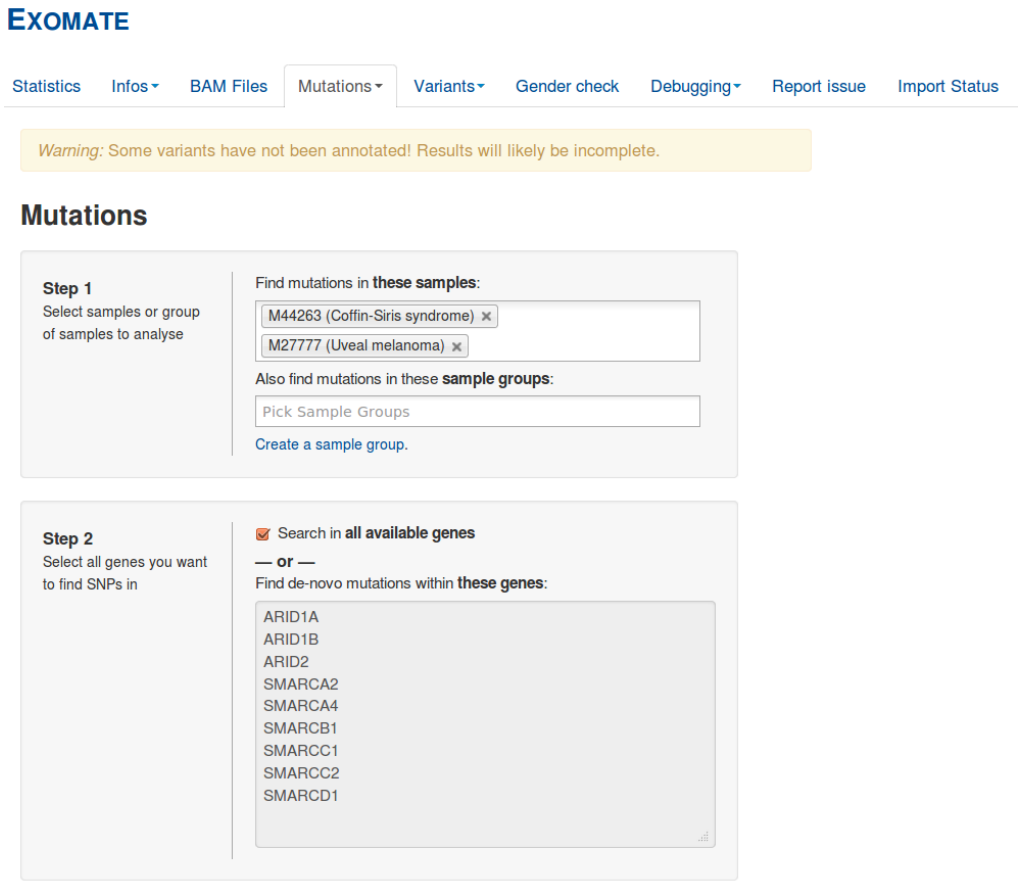
synonymous	encoded amino acid not changed
missense	amino acid changed
nonsense	regular codon becomes stop-codon
read-through	stop-codon becomes regular codon

Discard variants that are ...

- synonymous
- low quality
- known from healthy samples
- tolerated (e.g. according to SIFT (Kumar et al. 2009))

Exomate

- store variants in database
- perform filtering online with adjustable parameters and samples
- provide a web interface

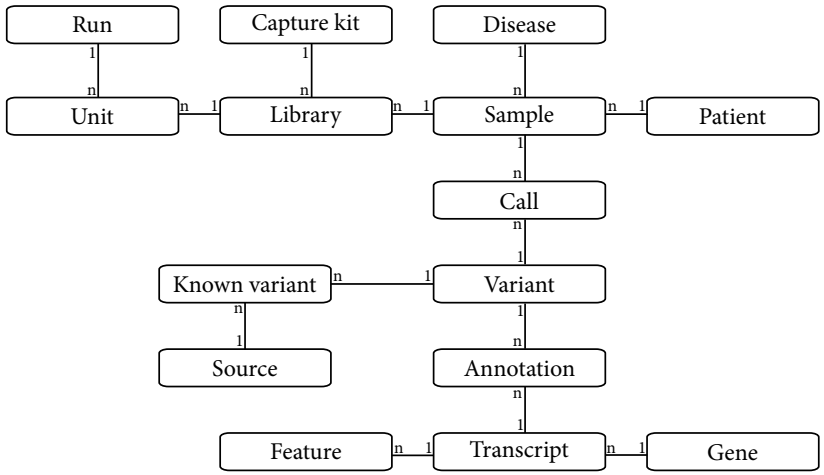


Exomate

Annotate, aggregate, subtract, sort, ...
This is what relational databases and the Structured Query Language (SQL) are designed for:

```
SELECT * FROM calls WHERE qual >= 50;
```

Database layout



Variant id, chrom, pos, ref, alt

Call variant_id, sample_id, qual, strand_bias, ...

Advantages of separating variants from calls:

- avoid redundancy in database
- lightweight filtering of calls without knowledge about the whole variant

Filtering algorithm

Given two sets of samples:

affected those we want to find mutations on

unaffected samples whose variants we want to subtract (e.g. healthy tissue)

1. retrieve all calls from unaffected samples above a given quality
2. obtain all calls from affected samples above a given quality and subtract (LEFT OUTER JOIN) above unaffected calls
3. subtract non-clinical known variants (dbsnp, . . .)
4. associate (JOIN) annotations (synonymous, intronic, . . .) to calls

	Gene	Common	Sample	Transcripts	Nucleotide	Codon	Amino acid	Type	Region	Splice	Quality	Location
1	ARID1A	1	M45227	001, 201, 002, more	C → T	CGA→ TGA	R → *	Nonsense	Coding		75.19	1:27106354
2	ARID1B	1	M45224	201, 203, 009, more	C → T	CGA→ TGA	R → *	Nonsense	Coding		37.98	6:157406006
3	SMARCB1	1	M44263	005, 001, 002, 003	G → A	CGG→ CAG	R → Q	Missense	Coding	Splice	224.79	22:24176330

Special thank goes to Marcel Martin, whose PhD thesis *"Algorithms and Tools for the Analysis of High-Throughput DNA Sequencing Data"* served as template for many of the slides.

References

Antonarakis, Stylianos E. et al. “The Nature and Mechanisms of Human Gene Mutation”. In: *The Metabolic and Molecular Bases of Inherited Disease*. Ed. by C. R. Scriver et al. 7th. New York: McGraw-Hill, 1995, 259–291.

DePristo, M.A. et al. “A framework for variation discovery and genotyping using next-generation DNA sequencing data”. In: *Nature genetics* 43.5 (May 2011), pp. 491–498.

Kumar, Prateek et al. “Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm”. en. In: *Nature Protocols* 4.7 (June 2009), pp. 1073–1081.

Köster, Johannes and Sven Rahmann. “Snakemake – a scalable bioinformatics workflow engine”. en. In: *Bioinformatics* 28.19 (Jan. 2012), pp. 2520–2522.

Martin, Marcel et al. “Exome sequencing identifies recurrent somatic mutations in EIF1AX and SF3B1 in uveal melanoma with disomy 3”. en. In: *Nature Genetics* advance online publication (June 2013).