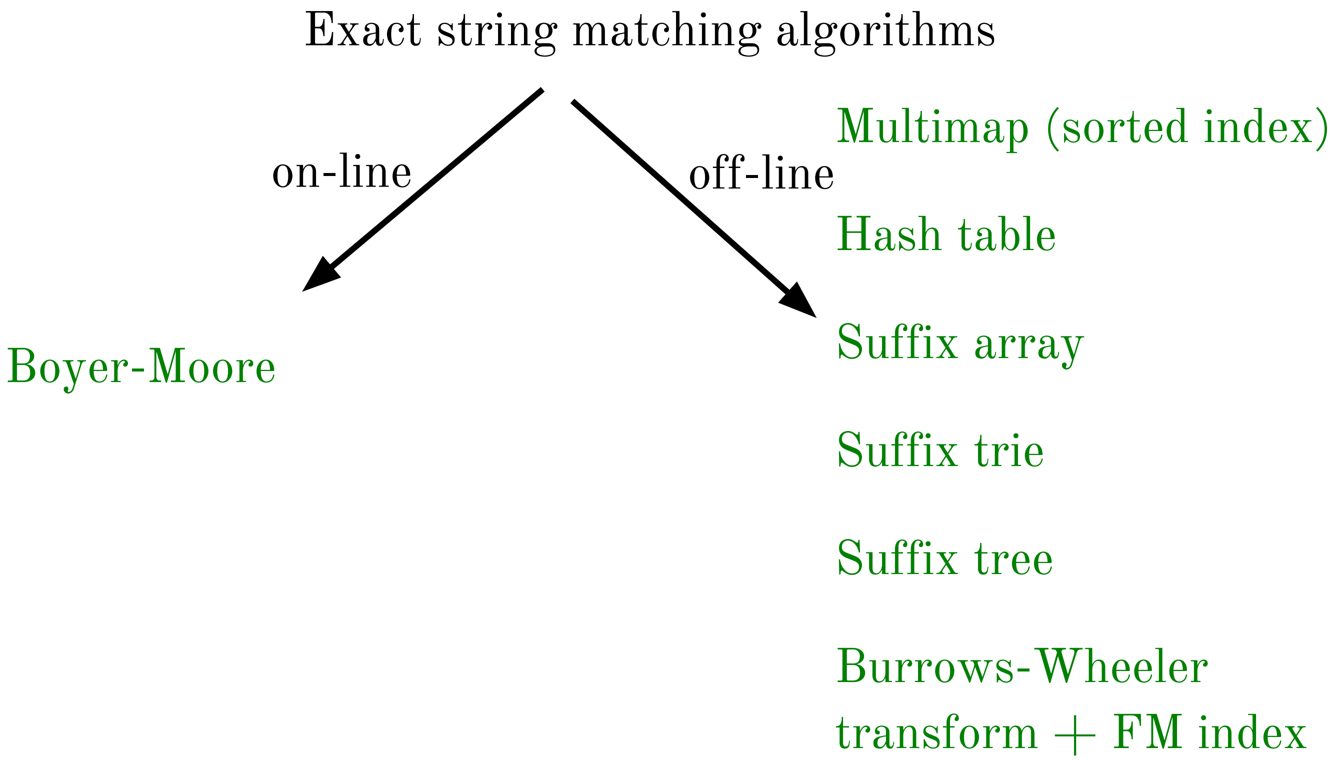
1. Bioinformatic workflows.
2. Variant calling.
3. Cancer analysis.

Lesson 05

Recapitulation



Exact string matching algorithms

on-line

off-line

Boyer-Moore

Multimap (sorted index)

Hash table

Suffix array

Suffix trie

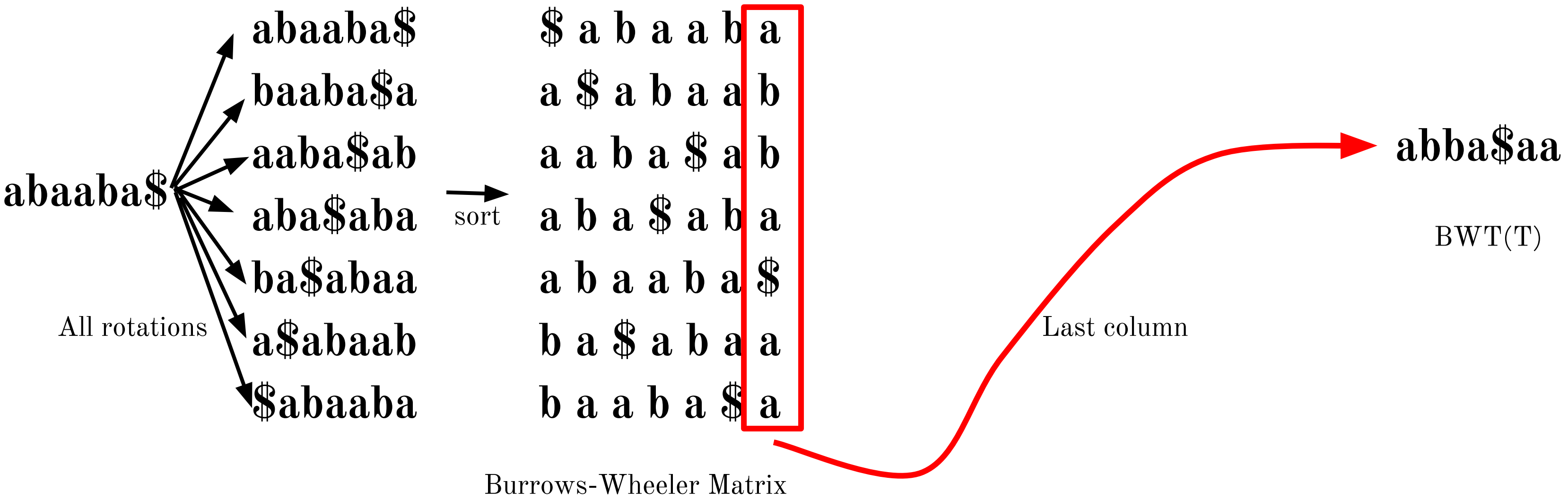
Suffix tree

Burrows-Wheeler transform + FM index

Slide adapted from Ben Langmead

Burrows-Wheeler Transform

Reversible permutation of the characters of a string, used originally for compression



abaaba$

All rotations

abaaba$
baaba$a
aaba$ab
aba$aba
ba$abaa
a$abaab
$abaaba

sort

$ a b a a b a
a $ a b a a b
a a b a $ a b
a b a $ a b a
a b a a b a $
b a $ a b a a
b a a b a $ a

Burrows-Wheeler Matrix

Last column

abba$aa

BWT(T)

How is it useful for compression?

How is it reversible?

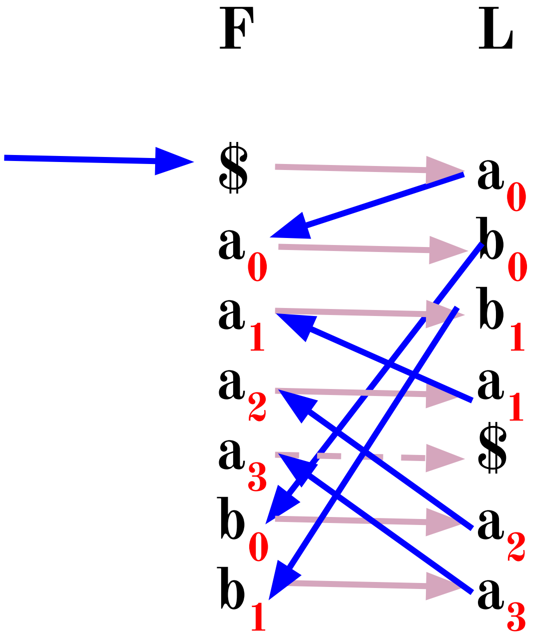
How is it an index?

Slide adapted from Ben Langmead

Burrows-Wheeler Transform: reversing

Reverse BWT(T) starting at right-hand-side of T and moving left

Start in first row. F must have \$.
L contains character just prior to \$: a_0
...



Reverse of chars we visited = $\mathbf{a}_3\ \mathbf{b}_1\ \mathbf{a}_1\ \mathbf{a}_2\ \mathbf{b}_0\ \mathbf{a}_0\ \mathbf{\$}$ = T

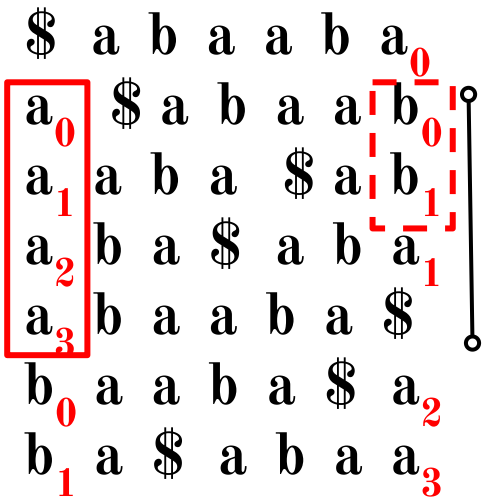
Slide adapted from Ben Langmead

FM Index: querying

Look for range of rows of BWM(T) with P as prefix

Do this for P’s shortest suffix, then extend to successively longer suffixes until range becomes empty or we’ve exhausted P

P = aba



$\$\ a\ b\ a\ a\ b\ a_0$

$a_0\ \$\ a\ b\ a\ a\ b_0$

$a_1\ a\ b\ a\ \$\ a\ b_1$

$a_2\ b\ a\ \$\ a\ b\ a_1$

$a_3\ b\ a\ a\ b\ a\ \$$

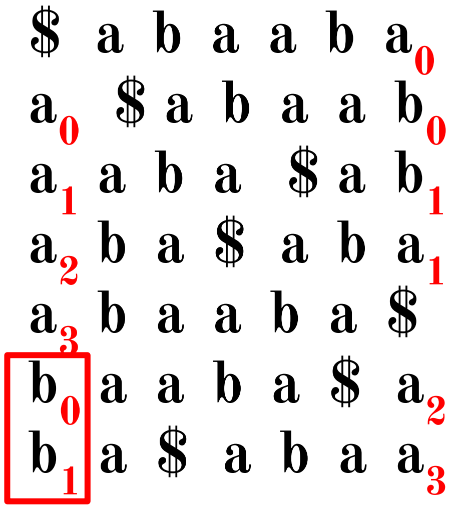
$b_0\ a\ a\ b\ a\ \$\ a_2$

$b_1\ a\ \$\ a\ b\ a\ a_3$

Look at those rows in L.

b_0, b_1 are b-s occurring just to left.

P = aba



$\$\ a\ b\ a\ a\ b\ a_0$

$a_0\ \$\ a\ b\ a\ a\ b_0$

$a_1\ a\ b\ a\ \$\ a\ b_1$

$a_2\ b\ a\ \$\ a\ b\ a_1$

$a_3\ b\ a\ a\ b\ a\ \$$

$b_0\ a\ a\ b\ a\ \$\ a_2$

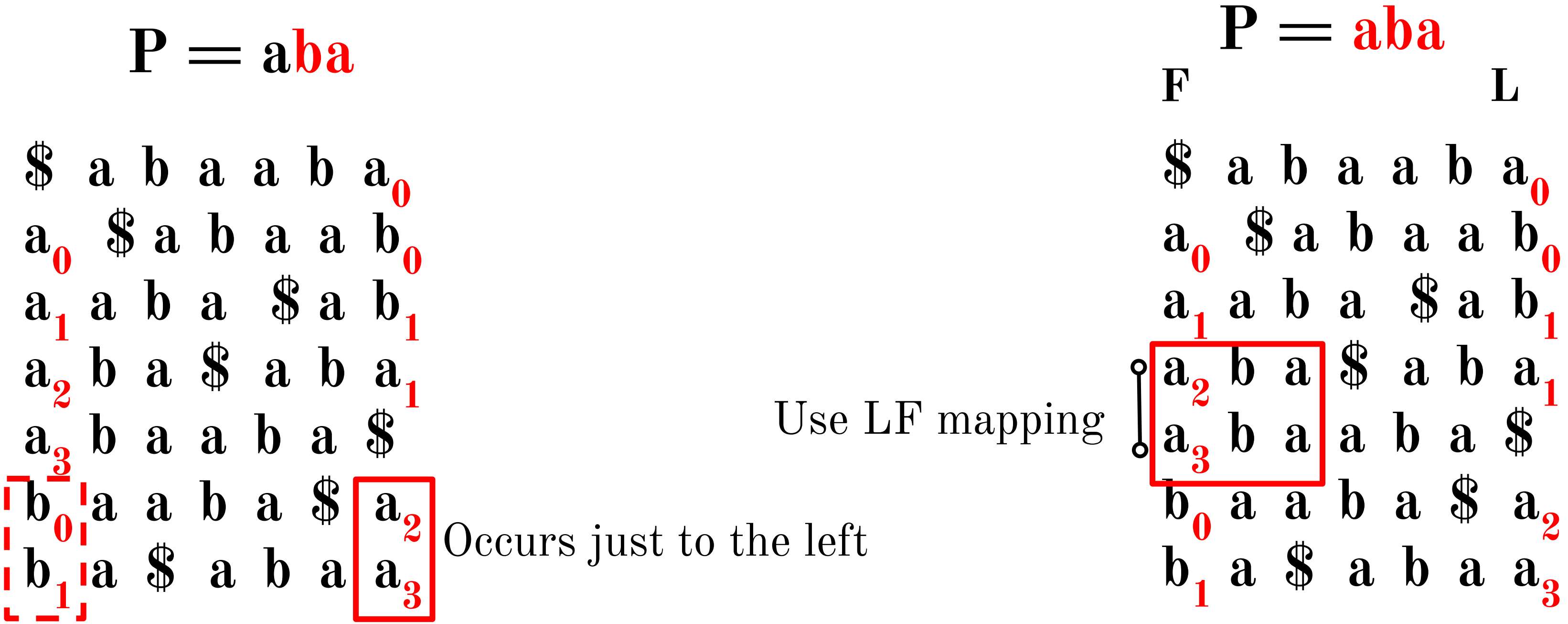
$b_1\ a\ \$\ a\ b\ a\ a_3$

Use LF Mapping. Let new range delimit those b-s

Slide adapted from Ben Langmead

FM Index: querying

We have rows beginning with **ba**, now we seek rows beginning with **aba**



P = aba

$\$\ a\ b\ a\ a\ b\ a_0$

$a_0\ \$\ a\ b\ a\ a\ b_0$

$a_1\ a\ b\ a\ \$\ a\ b_1$

$a_2\ b\ a\ \$\ a\ b\ a_1$

$a_3\ b\ a\ a\ b\ a\ \$$

$b_0\ a\ a\ b\ a\ \$\ a_2$

$b_1\ a\ \$\ a\ b\ a\ a_3$

Occurs just to the left

P = aba

F **L**

$\$\ a\ b\ a\ a\ b\ a_0$

$a_0\ \$\ a\ b\ a\ a\ b_0$

$a_1\ a\ b\ a\ \$\ a\ b_1$

$a_2\ b\ a\ \$\ a\ b\ a_1$

$a_3\ b\ a\ a\ b\ a\ \$$

$b_0\ a\ a\ b\ a\ \$\ a_2$

$b_1\ a\ \$\ a\ b\ a\ a_3$

Use LF mapping

Now we have the rows with prefix **aba**

Slide adapted from Ben Langmead

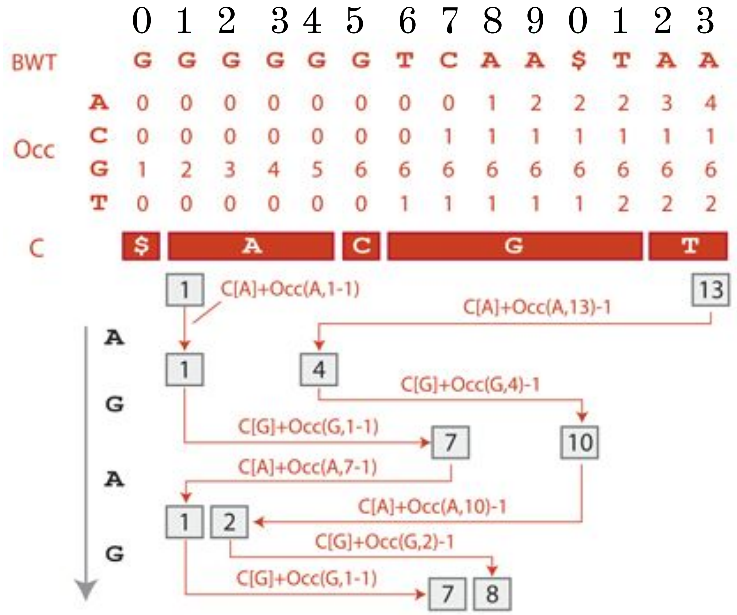
FM Index

1. L = BWT(T)
2. First column (number of appearances of each character)
3. Suffix Array (or SA Sample)
4. Tally (rank, occurrences) matrix

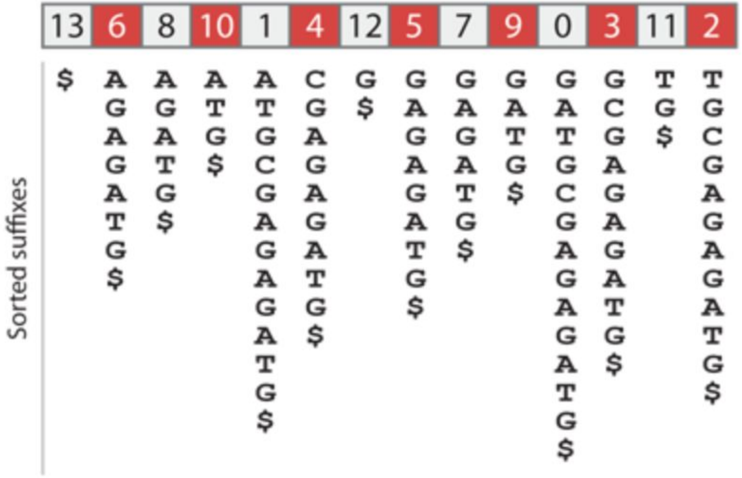
Slide adapted from Ben Langmead

FM Index: Example

Search for:
GAGA



		0	1	2	3	4	5	6	7	8	9	0	1	2	3
BWT		G	G	G	G	G	G	T	C	A	A	$	T	A	A
Occ	A	0	0	0	0	0	0	0	0	1	2	2	2	3	4
	C	0	0	0	0	0	0	0	1	1	1	1	1	1	1
	G	1	2	3	4	5	6	6	6	6	6	6	6	6	6
	T	0	0	0	0	0	0	1	1	1	1	1	2	2	2
C		$	A				C	G					T		



Sorted suffixes

13	6	8	10	1	4	12	5	7	9	0	3	11	2
$	AGAGATG$	AGATG$	ATG$	ATGCGAGAGATG$	CGAGAGATG$	G$	GAGAGATG$	GAGATG$	GATG$	GATGCGAGAGATG$	GCGAGAGATG$	TG$	TGCGAGAGATG$

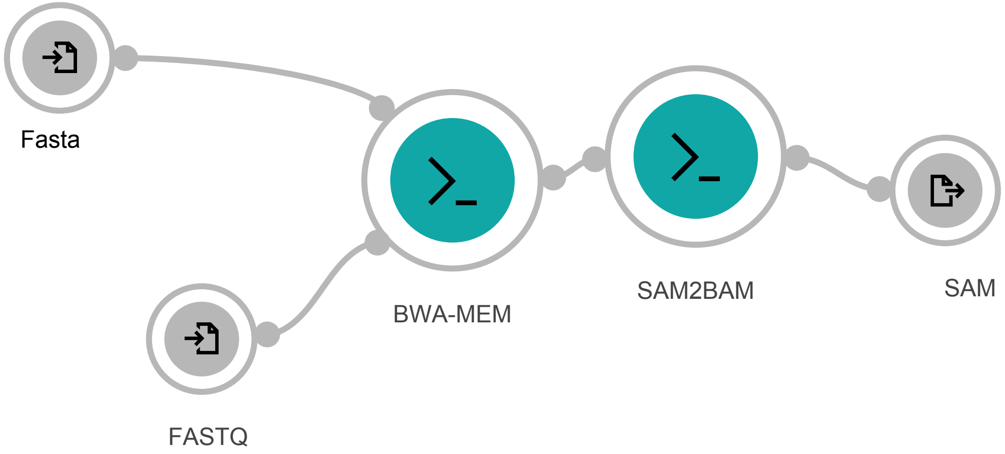
Source

Bioinformatic workflows and cloud computing

Lesson 05.1

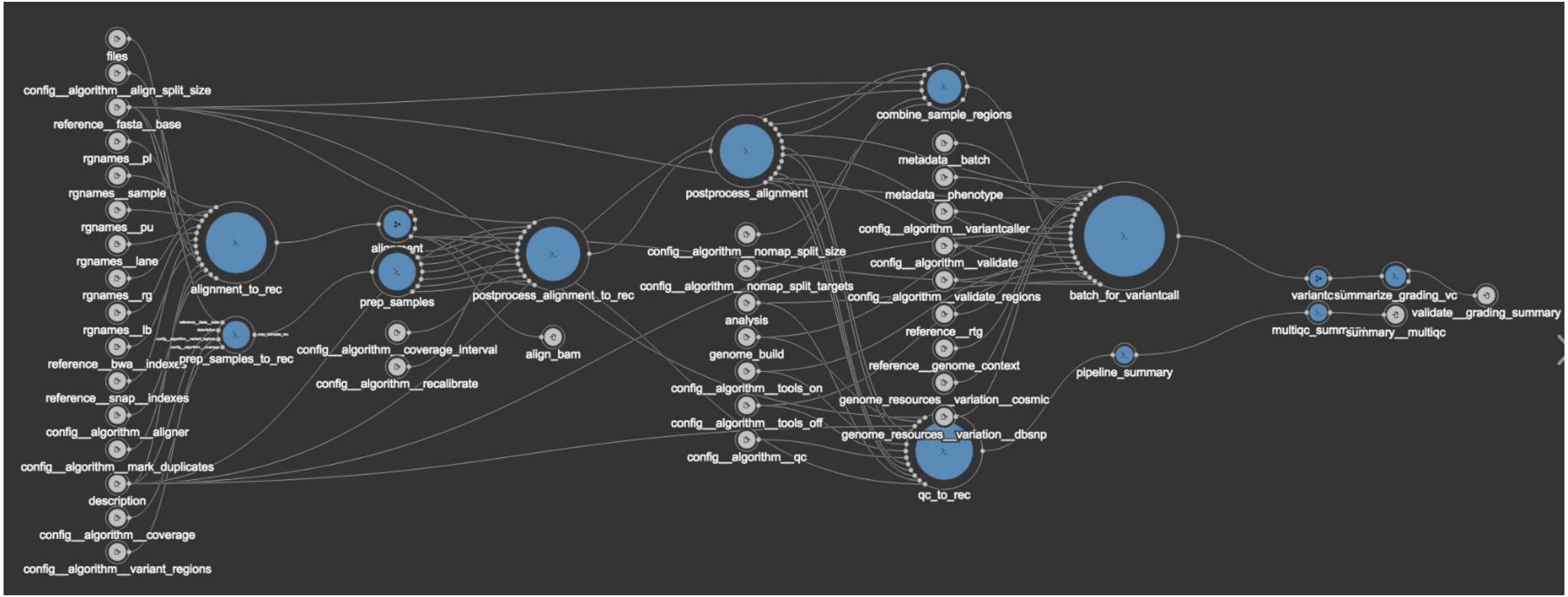
What is a workflow?

- Acyclic graph of tools connected to perform some analysis
- Workflow’s nodes are:
  - Inputs (file or parameter)
  - Tools
  - Outputs
  - Workflow



```
bwa mem ref.fa read1.fq read2.fq >
aln.sam
sam2bam aln.sam > aln.bam
```

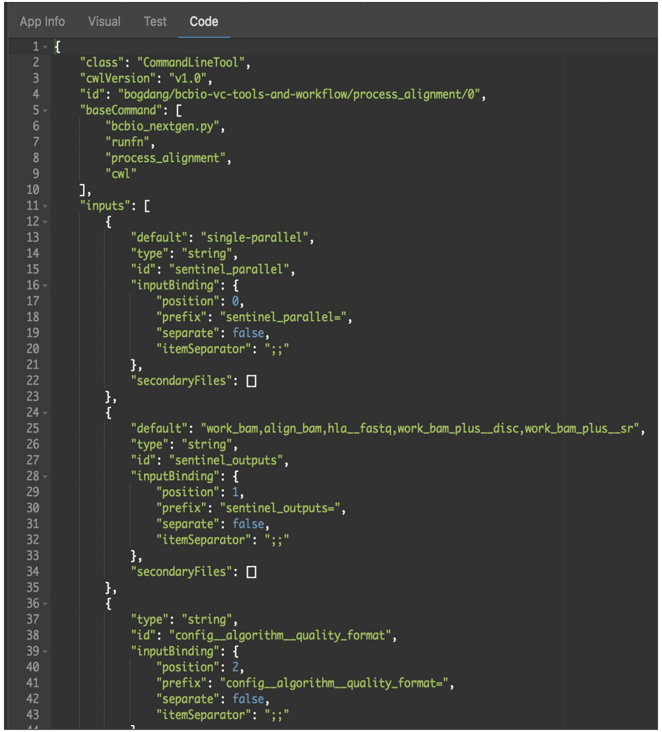
Why we need a workflow?


files
config__algorithm__align_split_size
reference__fasta__base
rgnames__pl
rgnames__sample
rgnames__pu
rgnames__lane
rgnames__rg
rgnames__lb
reference__bwa__indexes
reference__snap__indexes
config__algorithm__aligner
config__algorithm__mark_duplicates
description
config__algorithm__coverage
config__algorithm__variant_regions
alignment_to_rec
prep_samples_to_rec
alignment
prep_samples
config__algorithm__coverage_interval
config__algorithm__recalibrate
postprocess_alignment_to_rec
align_bam
postprocess_alignment
config__algorithm__nomap_split_size
config__algorithm__nomap_split_targets
analysis
genome_build
config__algorithm__tools_on
config__algorithm__tools_off
config__algorithm__qc
combine_sample_regions
metadata__batch
metadata__phenotype
config__algorithm__variantcaller
config__algorithm__validate
config__algorithm__validate_regions
reference__rtg
reference__genome_context
genome_resources__variation__cosmic
genome_resources__variation__dbsnp
qc_to_rec
batch_for_variantcall
pipeline_summary
variantcsummarize_grading_vc
validate__grading_summary
multiqc_summsummary__multiqc

Common Workflow Language

COMMON
WORKFLOW
LANGUAGE

- Define inputs and outputs of a software, runtime and requirements
- Define how to connect software, creating a workflow
- Ensure reproducibility and portability
- Think of CWL as a detailed recipe!



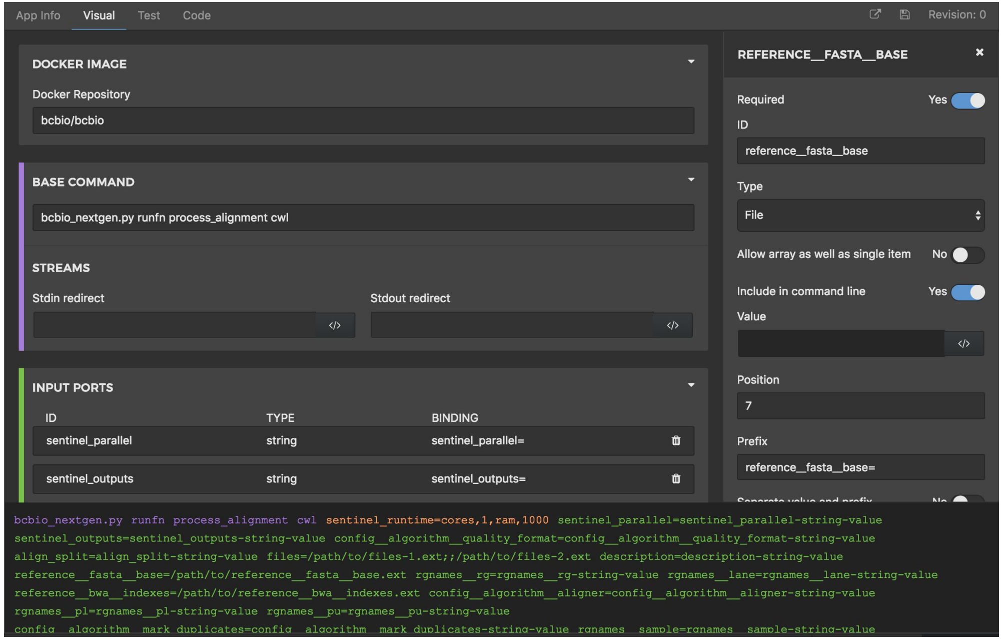
```
{
    "class": "CommandLineTool",
    "cwlVersion": "v1.0",
    "id": "bogdang/bcbio-vc-tools-and-workflow/process_alignment/0",
    "baseCommand": [
        "bcbio_nextgen.py",
        "runfn",
        "process_alignment",
        "cwl"
    ],
    "inputs": [
        {
            "default": "single-parallel",
            "type": "string",
            "id": "sentinel_parallel",
            "inputBinding": {
                "position": 0,
                "prefix": "sentinel_parallel=",
                "separate": false,
                "itemSeparator": ";;"
            },
            "secondaryFiles": []
        },
        {
            "default": "work_bam,align_bam,hla__fastq,work_bam_plus__disc,work_bam_plus__sr",
            "type": "string",
            "id": "sentinel_outputs",
            "inputBinding": {
                "position": 1,
                "prefix": "sentinel_outputs=",
                "separate": false,
                "itemSeparator": ";;"
            },
            "secondaryFiles": []
        },
        {
            "type": "string",
            "id": "config__algorithm__quality_format",
            "inputBinding": {
                "position": 2,
                "prefix": "config__algorithm__quality_format=",
                "separate": false,
                "itemSeparator": ";;"
```

Common Workflow Language

COMMON WORKFLOW LANGUAGE

- **Reproducible** analyses (standard)
- Scalable execution
- Metadata & file registry integration
- **Portability** - deployable on multiple platforms
- Revision management and versioning

  User management / permissions



DOCKER IMAGE

Docker Repository: bcbio/bcbio

BASE COMMAND

bcbio_nextgen.py runfn process_alignment cwl

STREAMS

Stdin redirect

Stdout redirect

INPUT PORTS

ID	TYPE	BINDING
sentinel_parallel	string	sentinel_parallel=
sentinel_outputs	string	sentinel_outputs=

REFERENCE__FASTA__BASE

Required: Yes

ID: reference__fasta__base

Type: File

Allow array as well as single item: No

Include in command line: Yes

Value

Position: 7

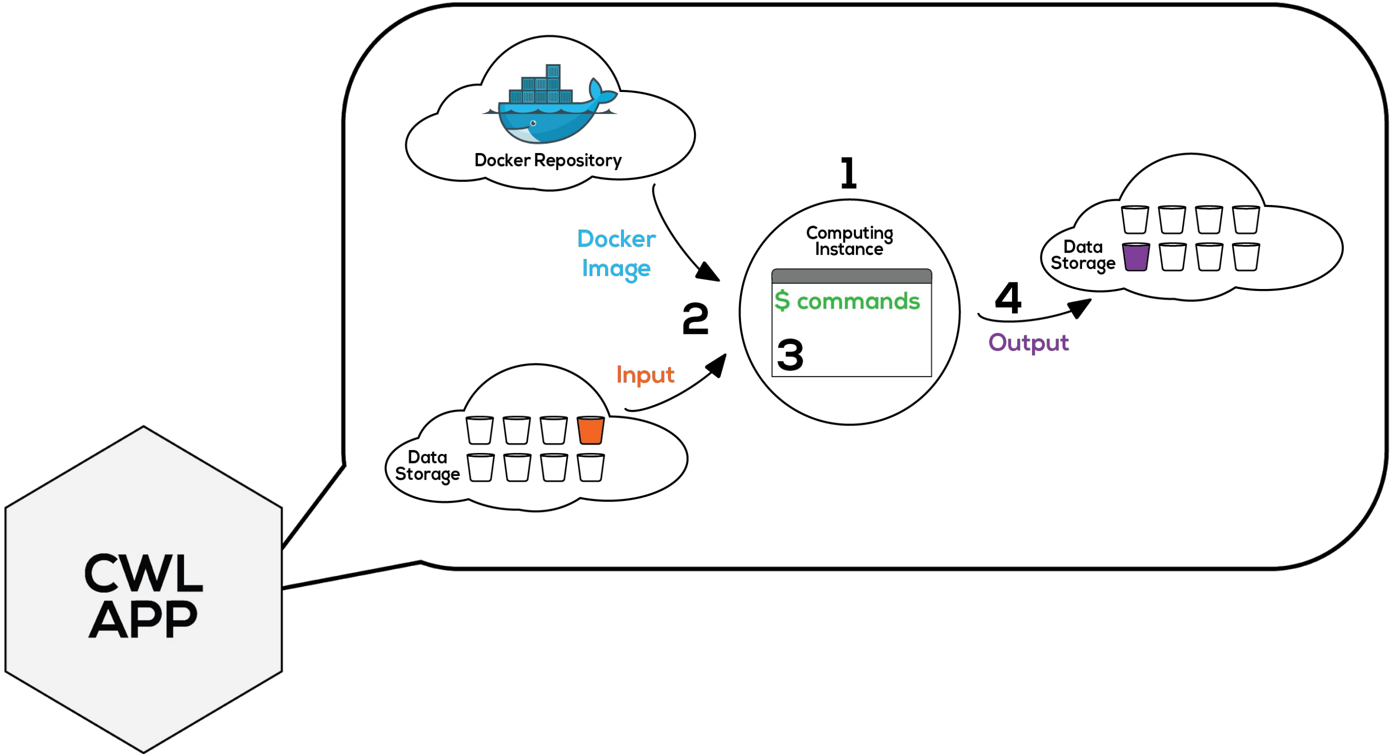
Prefix: reference__fasta__base=

Separate value and prefix: No

```
bcbio_nextgen.py runfn process_alignment cwl sentinel_runtime=cores,1,ram,1000 sentinel_parallel=sentinel_parallel-string-value
sentinel_outputs=sentinel_outputs-string-value config__algorithm__quality_format=config__algorithm__quality_format-string-value
align_split=align_split-string-value files=/path/to/files-1.ext;;/path/to/files-2.ext description=description-string-value
reference__fasta__base=/path/to/reference__fasta__base.ext rgnames__rg=rgnames__rg-string-value rgnames__lane=rgnames__lane-string-value
reference__bwa__indexes=/path/to/reference__bwa__indexes.ext config__algorithm__aligner=config__algorithm__aligner-string-value
rgnames__pl=rgnames__pl-string-value rgnames__pu=rgnames__pu-string-value
config__algorithm__mark_duplicates=config__algorithm__mark_duplicates-string-value rgnames__sample=rgnames__sample-string-value
```

CWL @ Cloud

COMMON
WORKFLOW
LANGUAGE

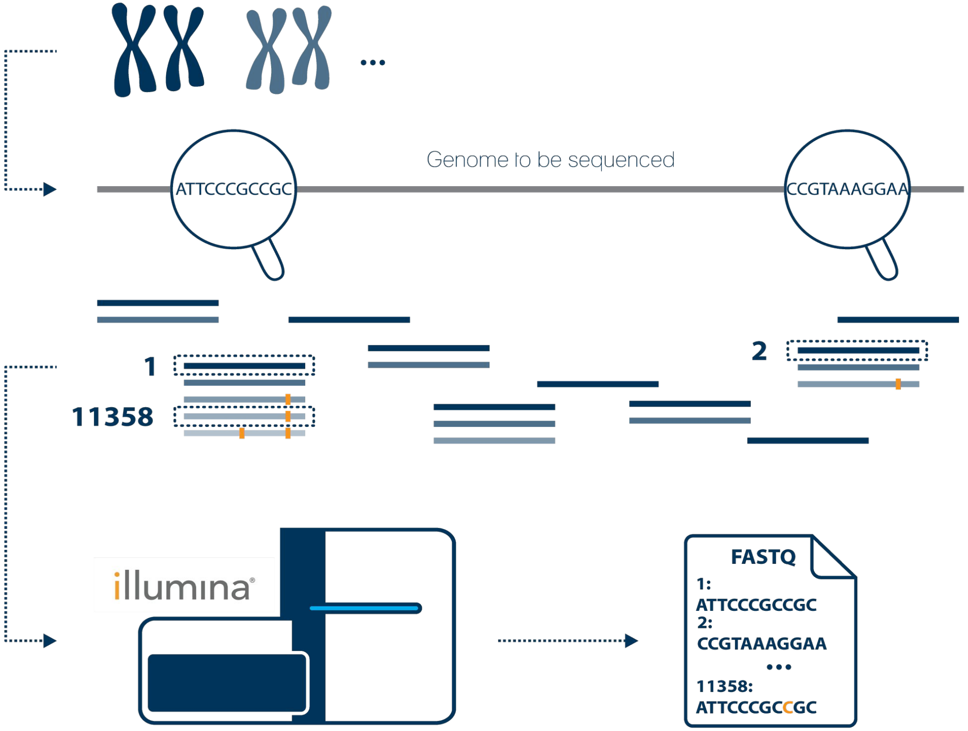


Variant Calling

Lesson 05.2

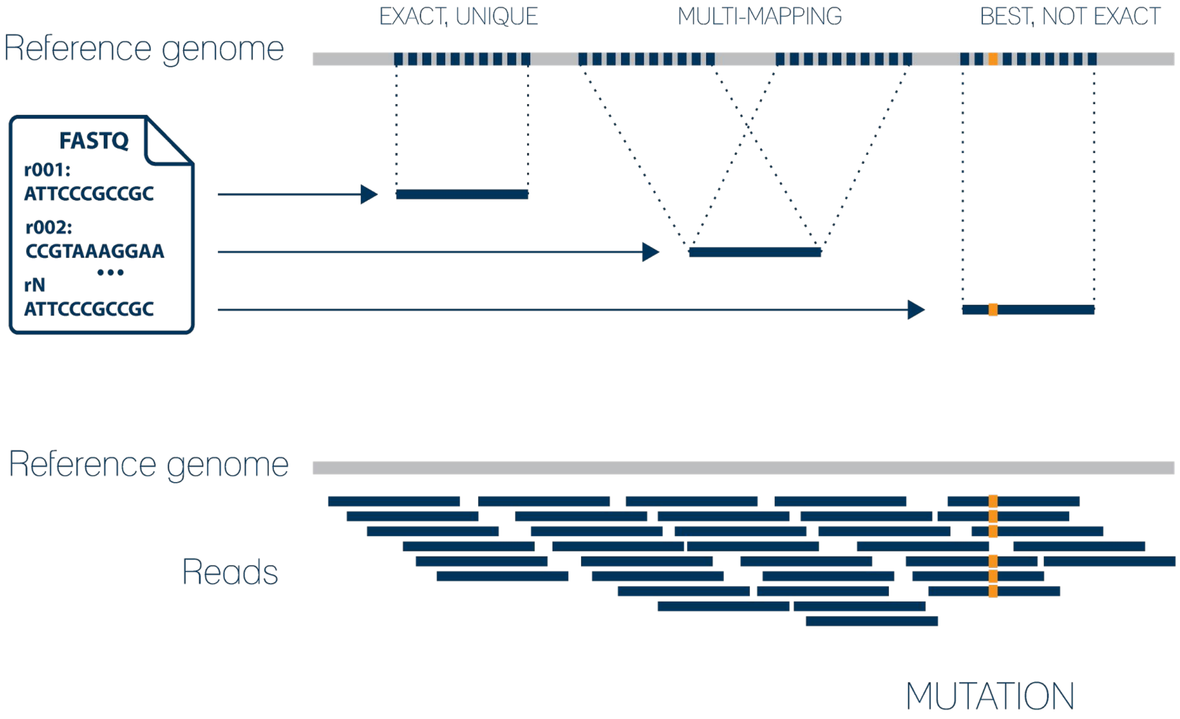
Reminder: DNA Sequencing

We got a FASTQ files with the “reads” – little pieces of the genome.



[redacted]

Reminder: Alignment



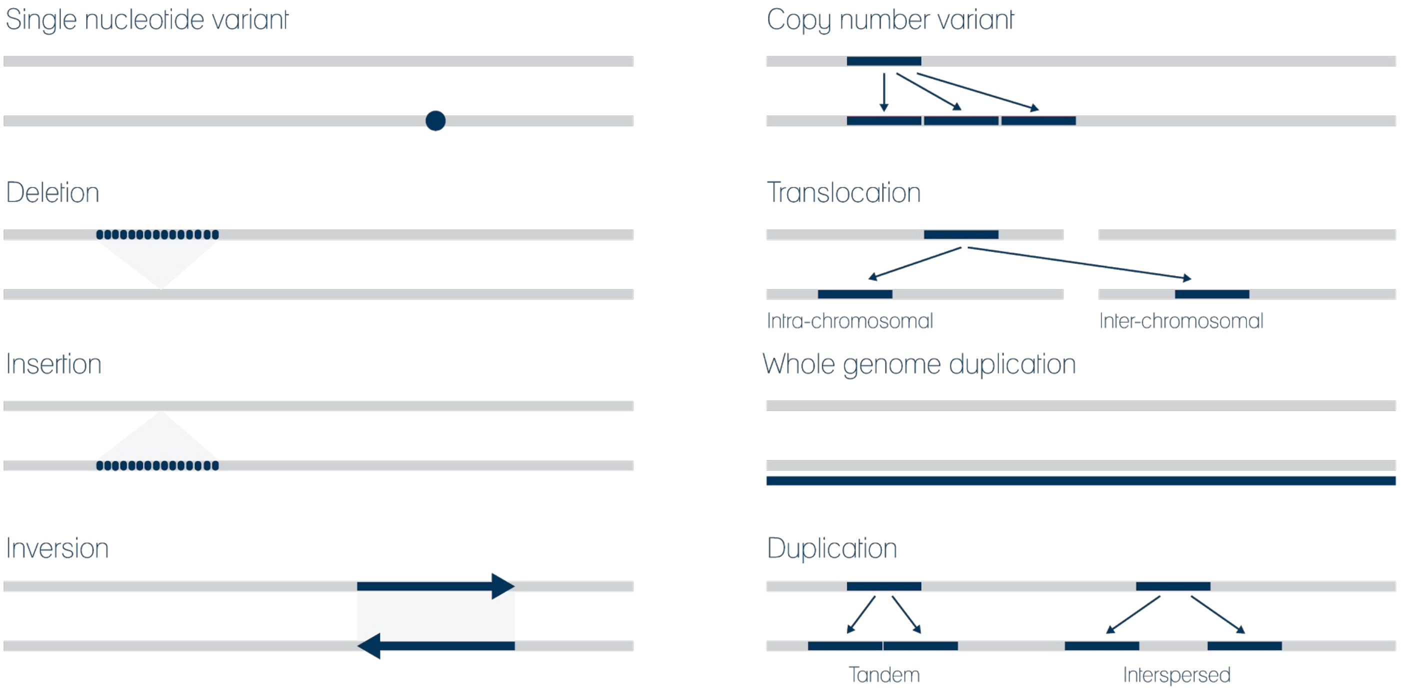
Introduction to Variant calling

- Variant calling is the process of finding differences between reference genome and observed sample

- We need aligned reads to the reference genome so we can find – “call” variants

- Different types of genomic variants

Genomic Variants



Single nucleotide variant

Deletion

Insertion

Inversion

Copy number variant

Translocation

Intra-chromosomal

Inter-chromosomal

Whole genome duplication

Duplication

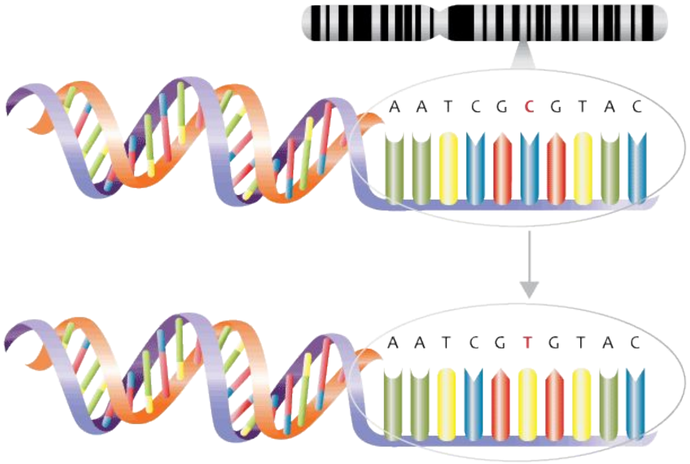
Tandem

Interspersed

Genomic Variants

- SNV (Single Nucleotide Variant)

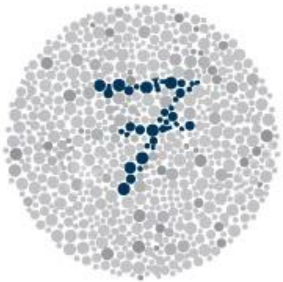
  Simple ones - not a big change on the first look, but...



Genomic Variants

Each of those characteristics causes one SNV

LONGER EYELASHES



DALTONISM

LESS SLEEPING

SUPER STRENGTH

Genomic Variants

Breast Cancer

BRCA2 gene (TS)

SNV id : rs1799954

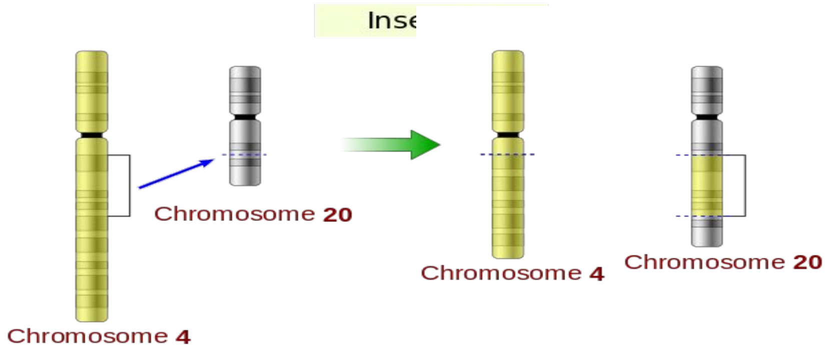
Chromosome 13
Position 32,340,455

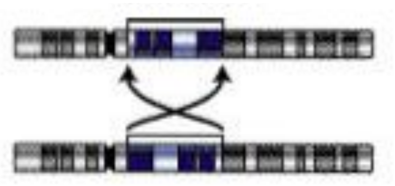
Cancer genotypes: **CC, CT and TT**

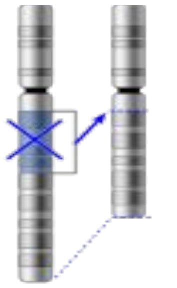
http://www.eupedia.com/genetics/cancer_related_snp.shtml
https://www.snpedia.com/index.php/Rs1799954

Genomic Variants

Deletions, Insertions, Translocations, Inversions, and some others...

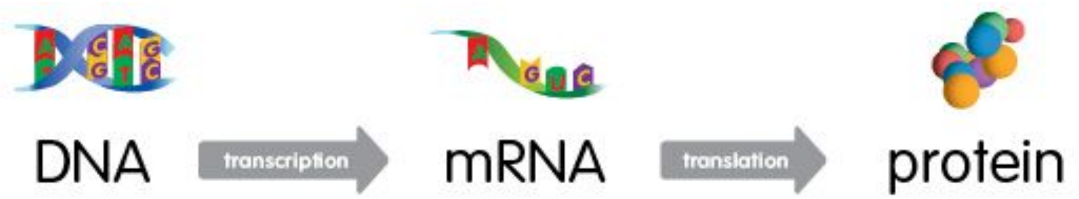


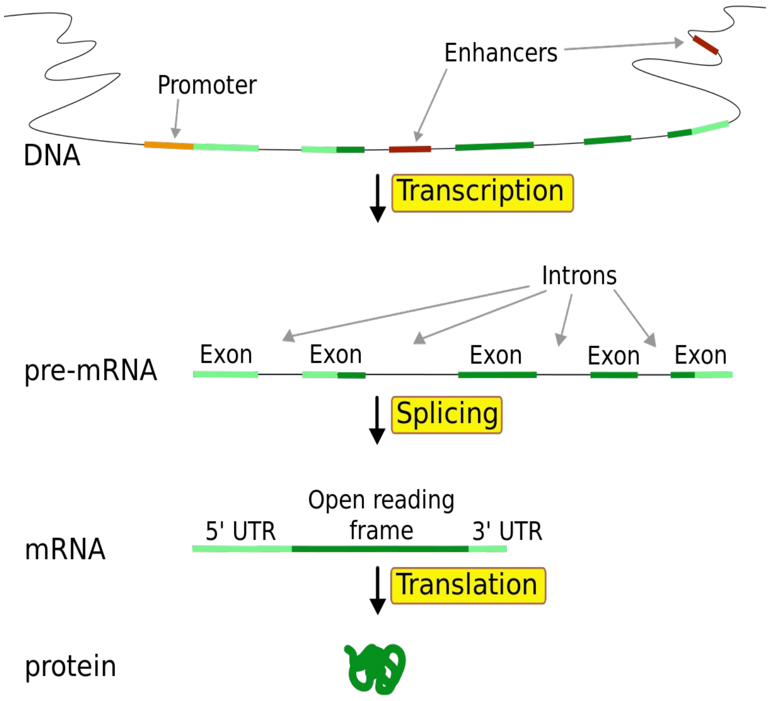




Genomic Variants

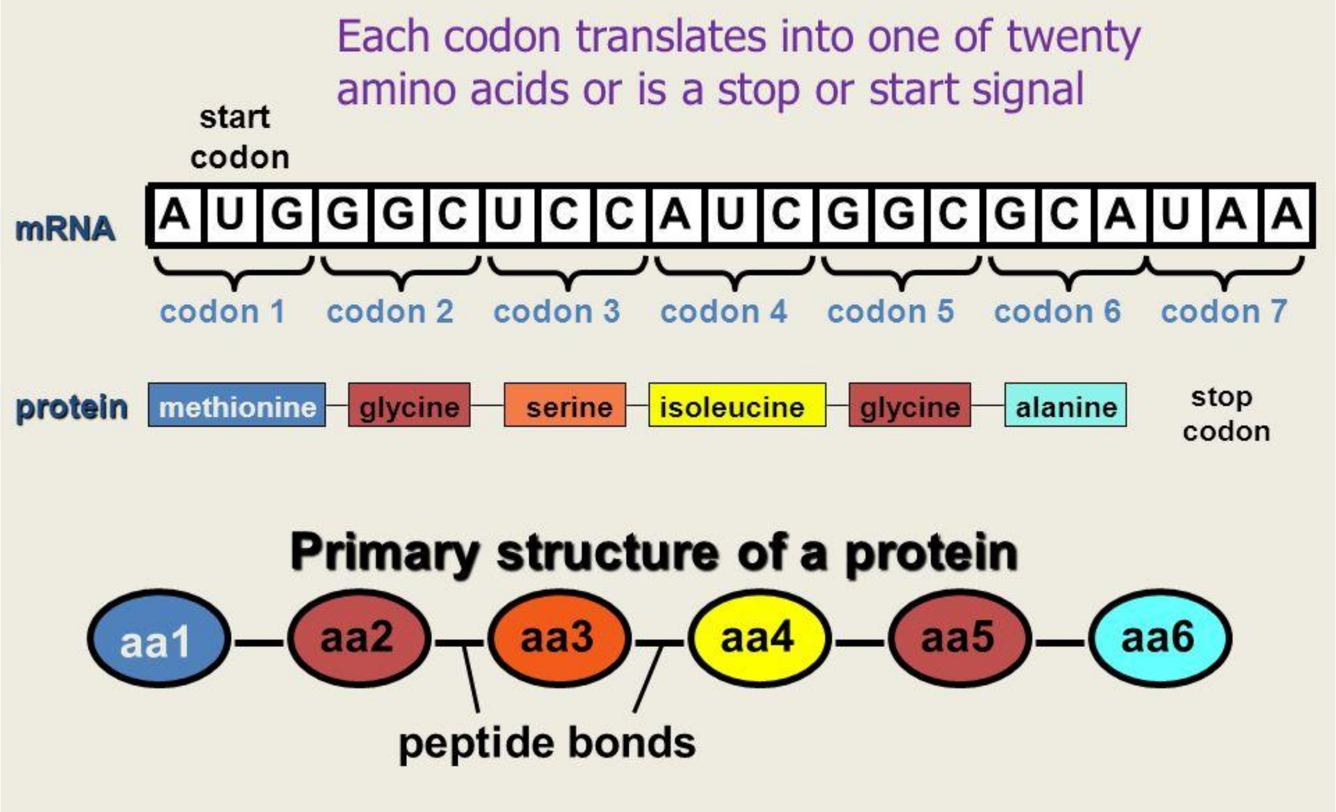
Based on the variant location, we can predict if mutation will have impact.





- Central dogma

RNA to Protein

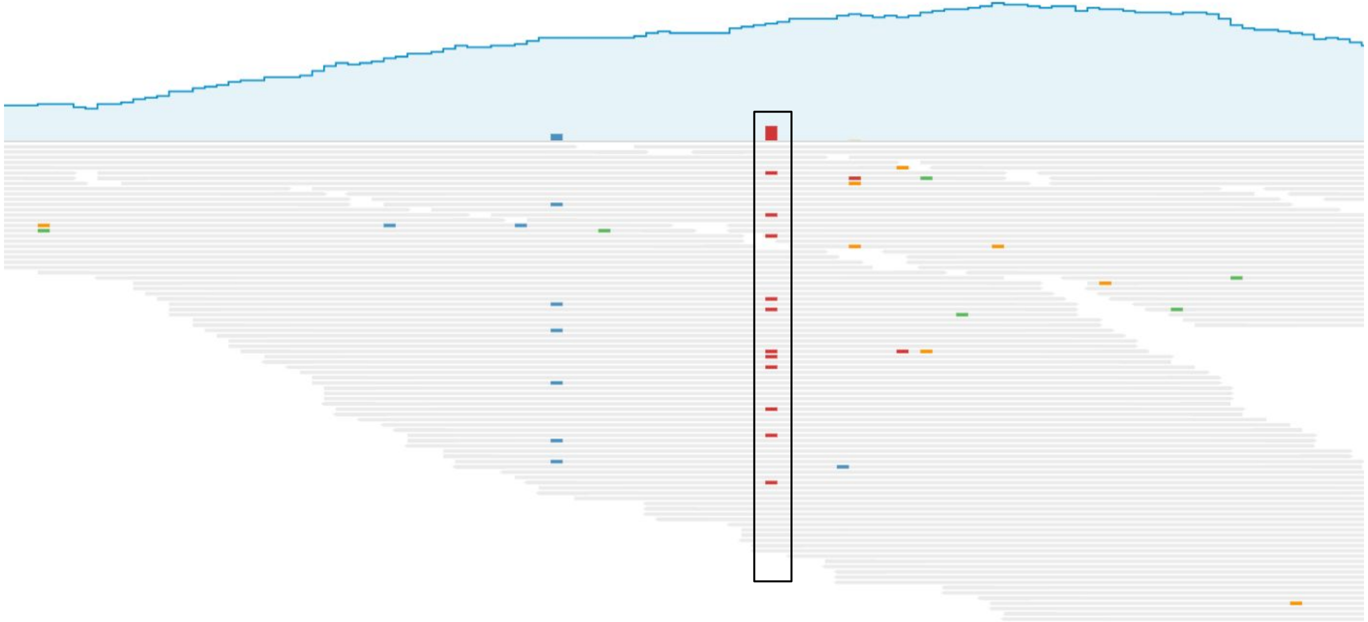


Each codon translates into one of twenty amino acids or is a stop or start signal

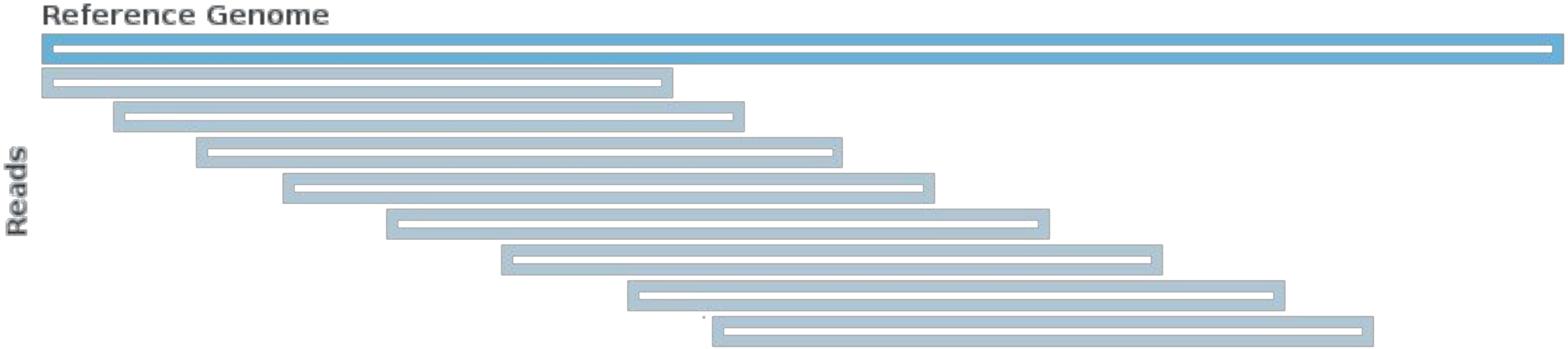
Genomic Variants

- Variants can have different impact on human cells and organism
- Single Nucleotide Variants(**SNV**):
  - Harmless
    - **Silent** – Usually no effect
  - Harmful:
    - **Missense** – Amino acid change
    - **Nonsense**(Start/Stop Gain/Lost) – AUG / UAG, UAA, UGA
  - Depends on the location
    - **Noncoding regions** (Promoter, Enhancer, IncRNA, miRNA...)
- Insertions/Deletions – **INDELS**
  - **In frame**
  - **Out of frame (Frameshift)**

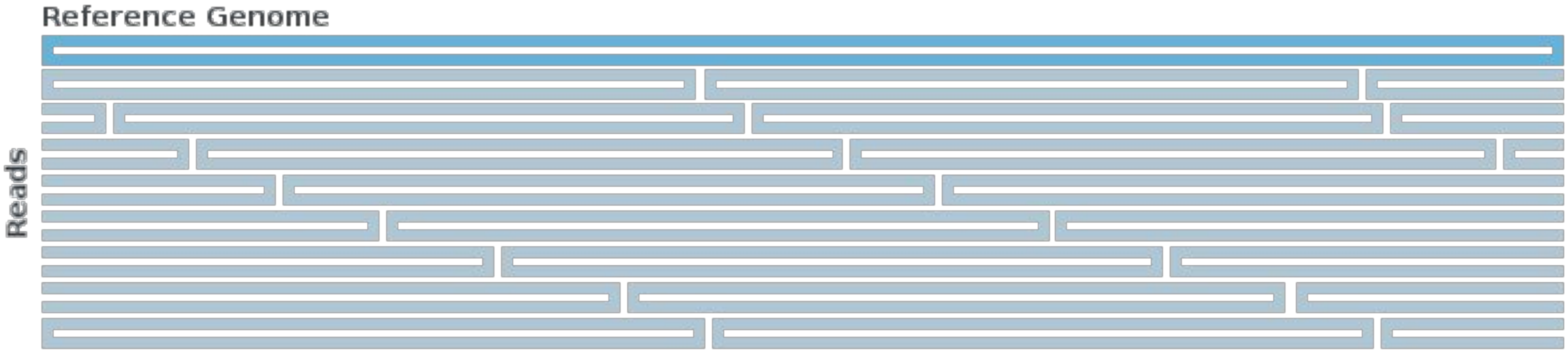
What is the pileup?



Ideal Variant Calling

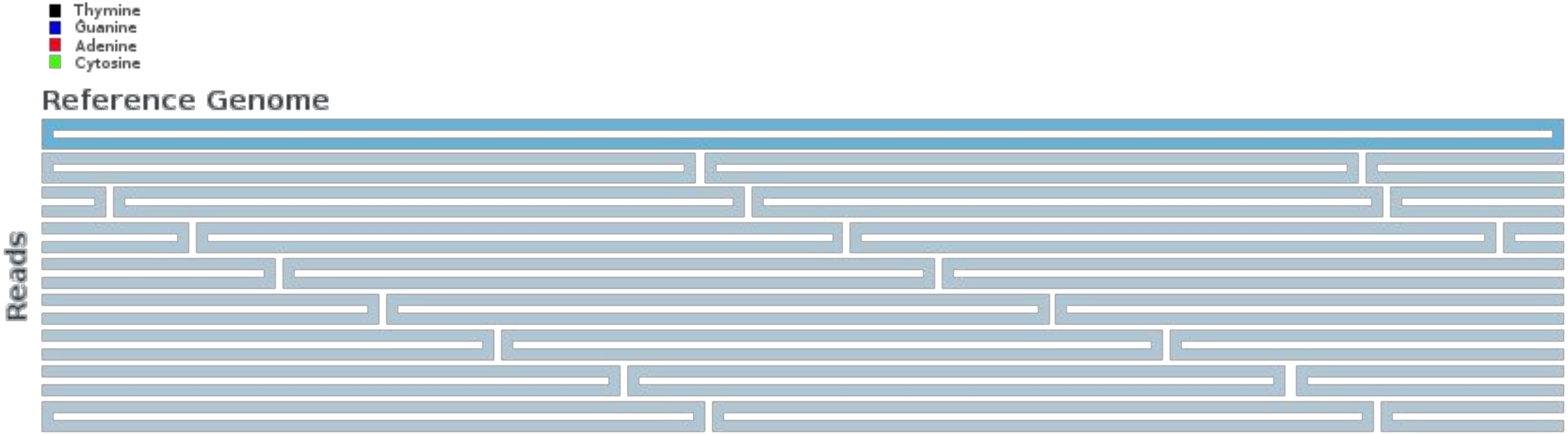


Ideal Variant Calling

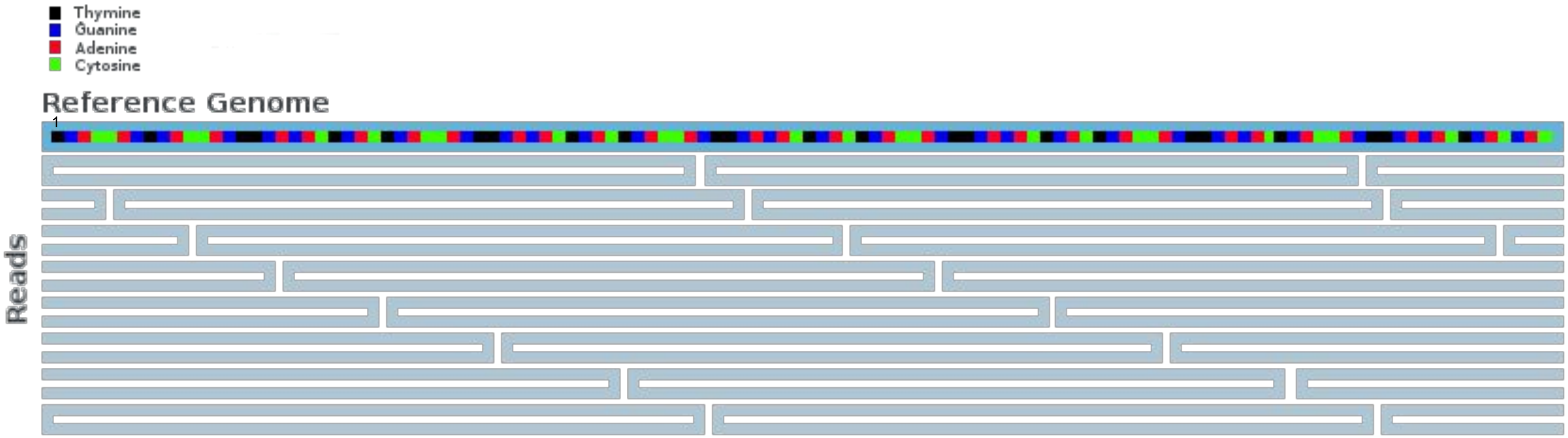


Ideally we will have uniform distribution of reads.

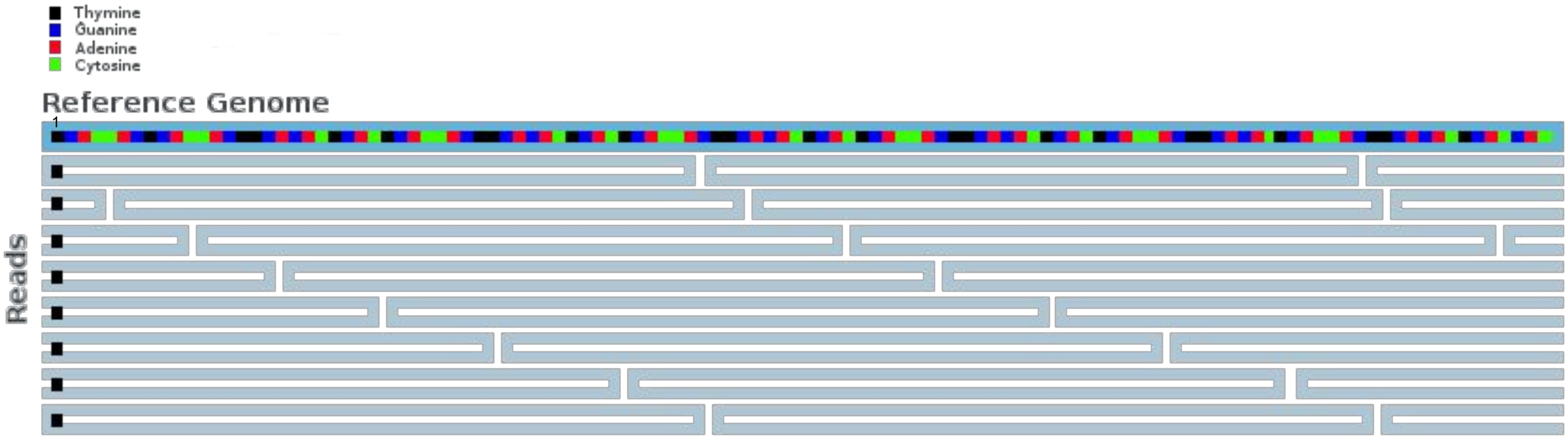
Ideal Variant Calling



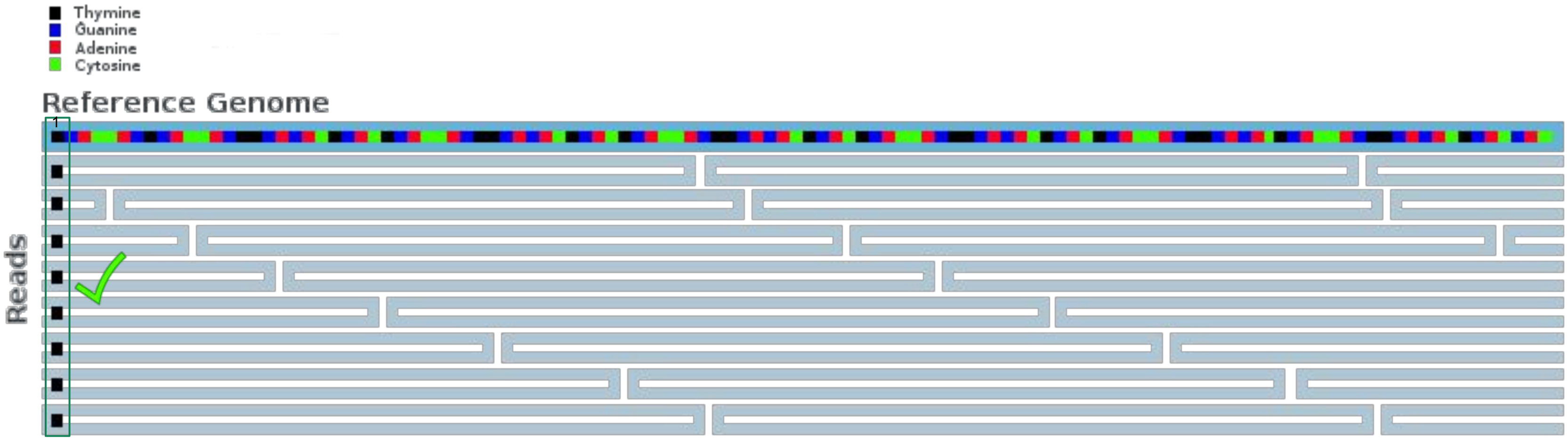
Ideal Variant Calling

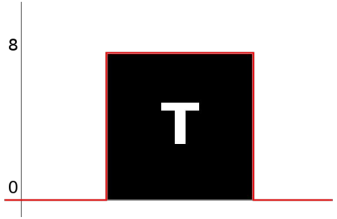


Ideal Variant Calling



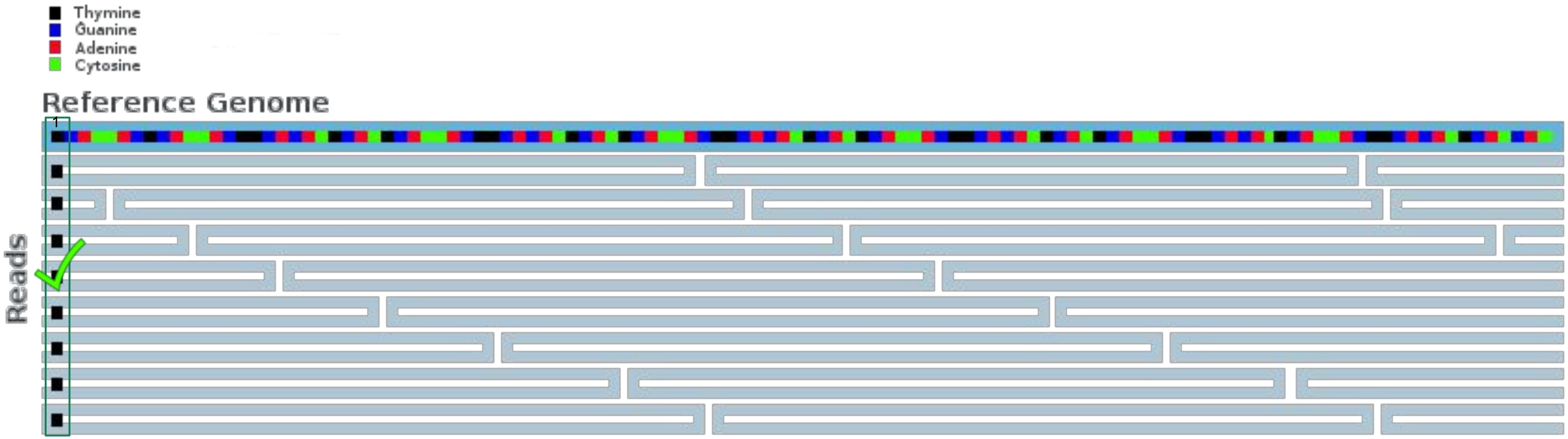
Ideal Variant Calling

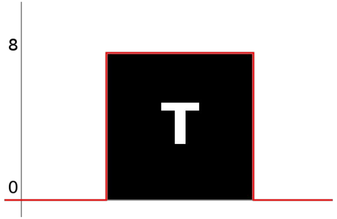




- We have “T” in the all reads covering that position

Ideal Variant Calling

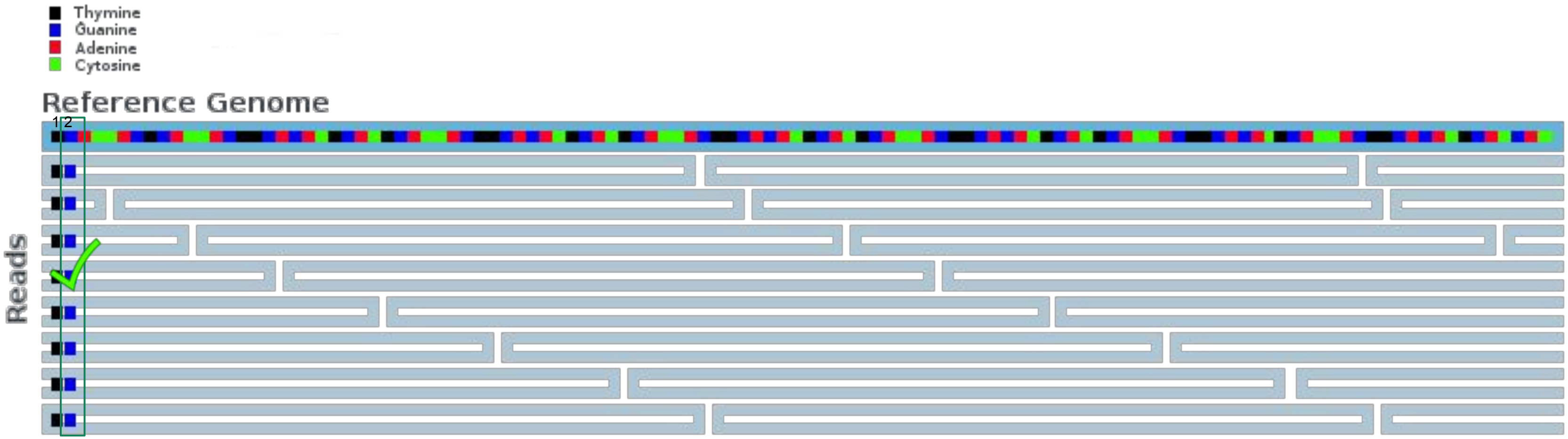


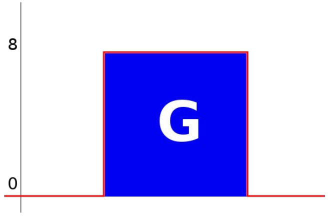


How can we represent what we have observed?

CONTIG	POS	REF	ALT	GT
X	1	T	-	0/0

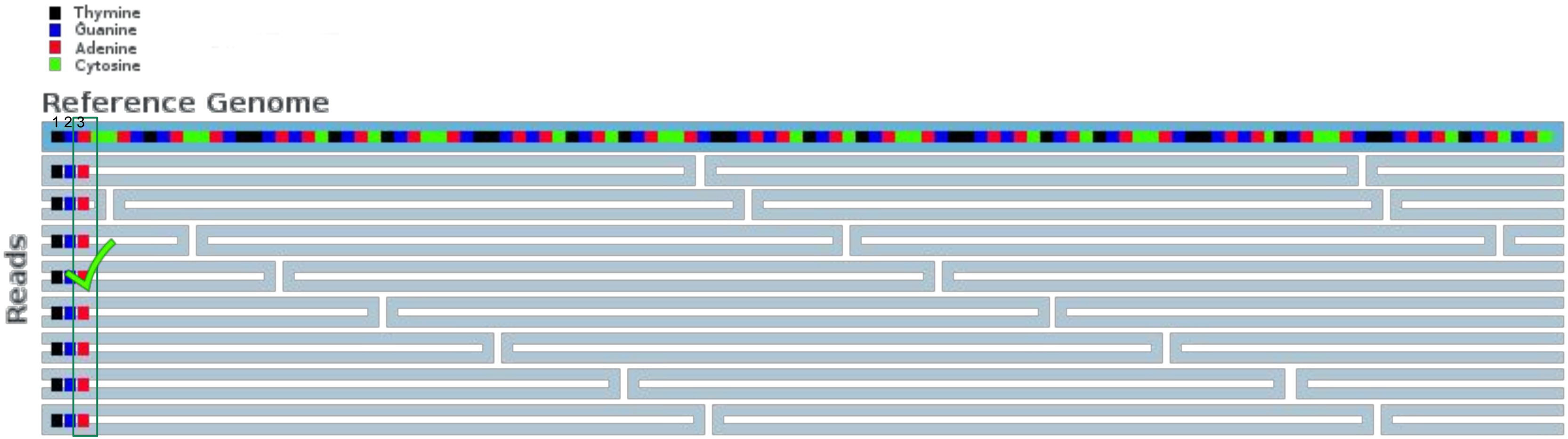
Ideal Variant Calling

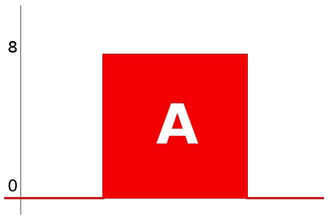




CONTIG	POS	REF	ALT	GT
X	1	T	-	0/0
X	2	G	-	0/0

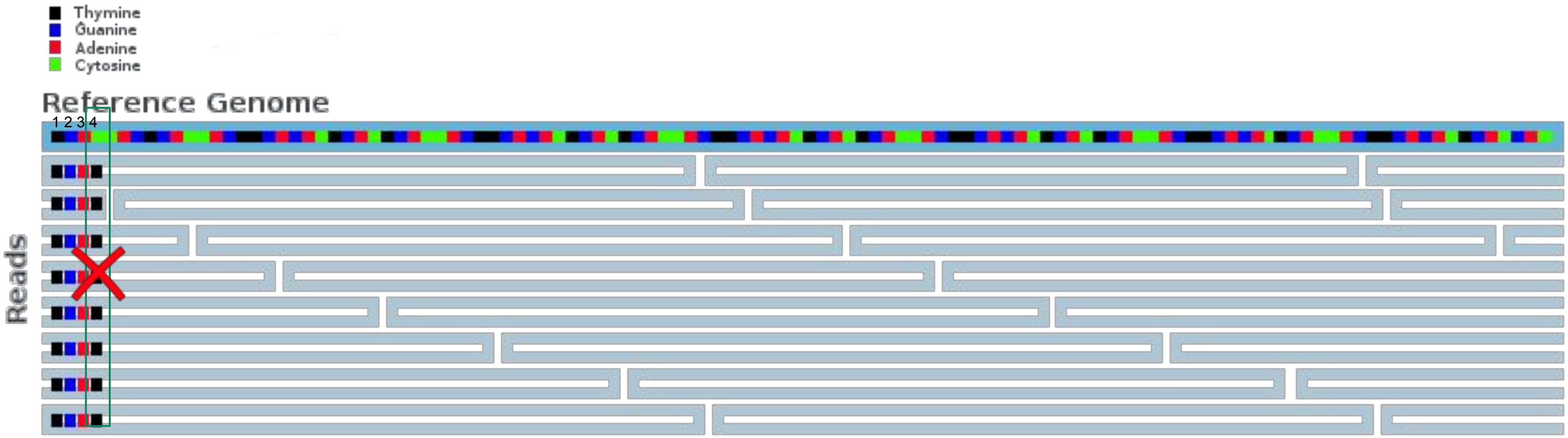
Ideal Variant Calling

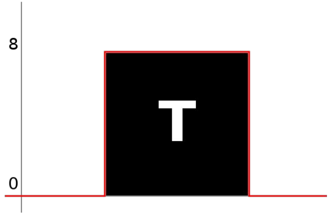




CONTIG	POS	REF	ALT	GT
X	1	T	-	0/0
X	2	G	-	0/0
X	3	A	-	0/0

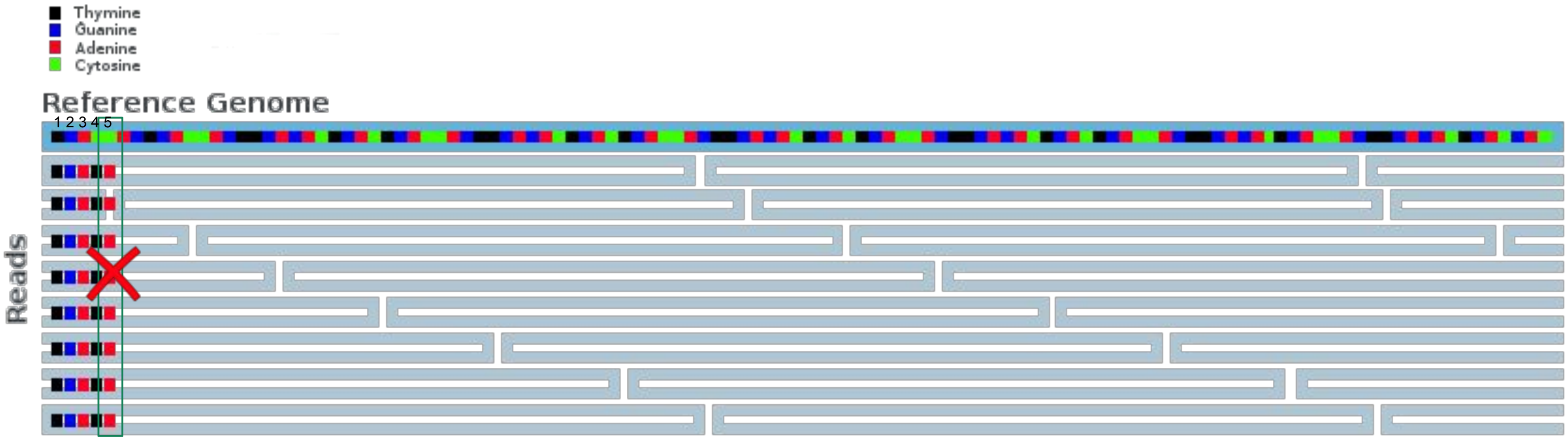
Ideal Variant Calling

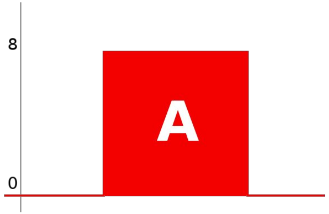




CONTIG	POS	REF	ALT	GT
X	2	G	-	0/0
X	3	A	-	0/0
X	4	**C**	**T**	**1/1**

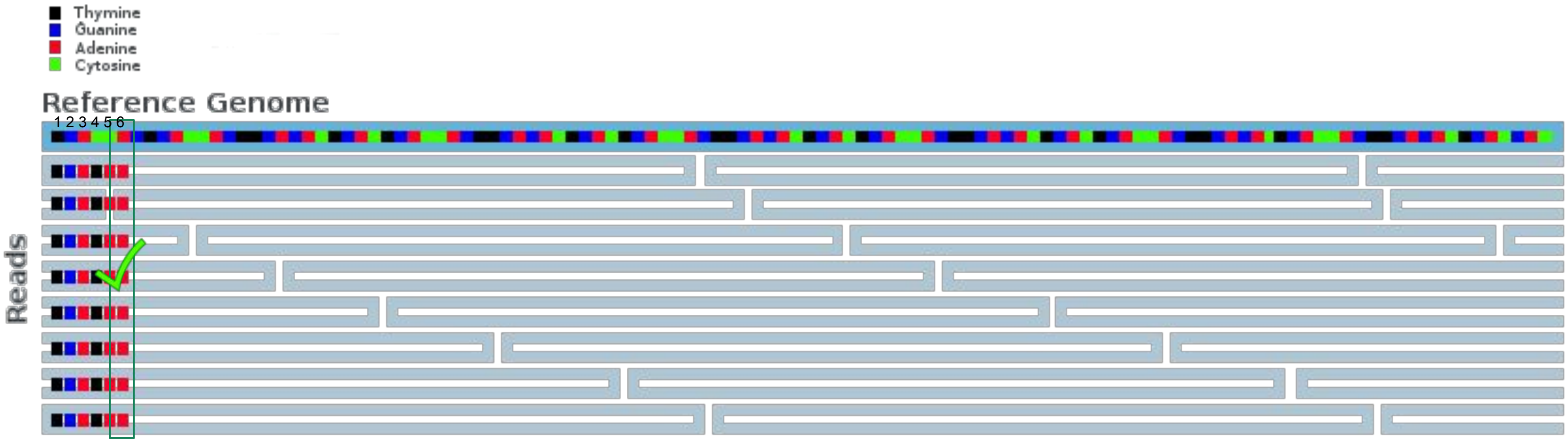
Ideal Variant Calling

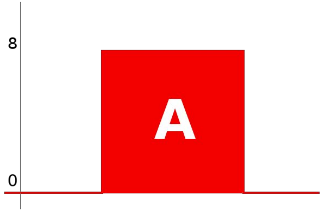




CONTIG	POS	REF	ALT	GT
X	3	A	-	0/0
X	4	**C**	**T**	**1/1**
X	5	**C**	**A**	**1/1**

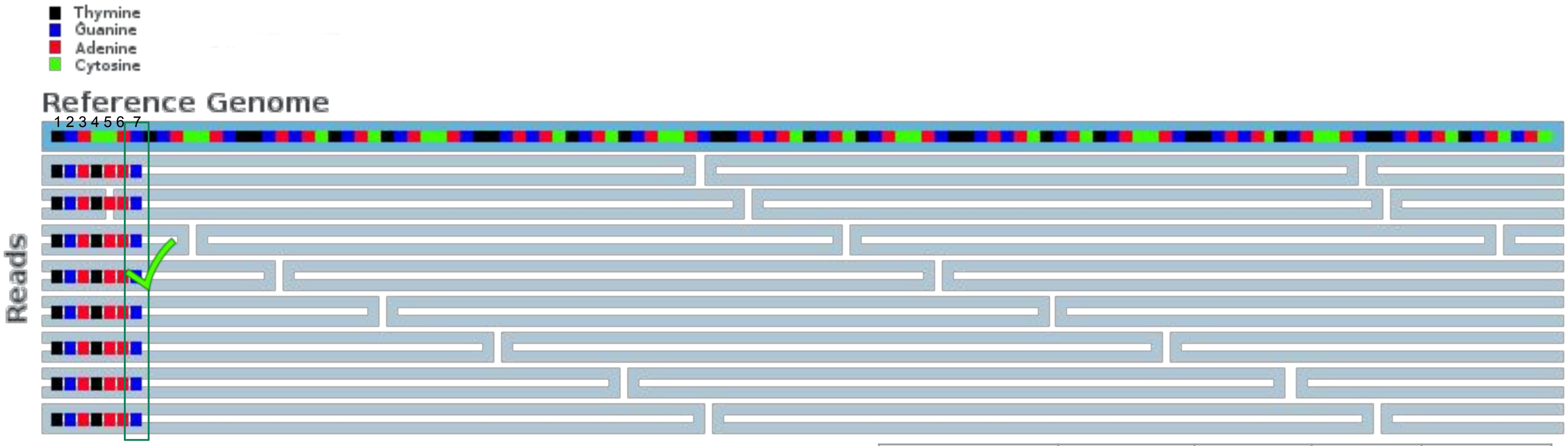
Ideal Variant Calling

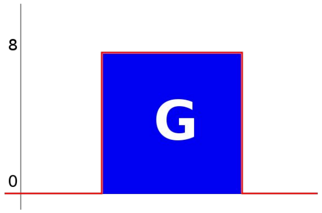




CONTIG	POS	REF	ALT	GT
X	4	**C**	**T**	**1/1**
X	5	**C**	**A**	**1/1**
X	6	A	-	0/0

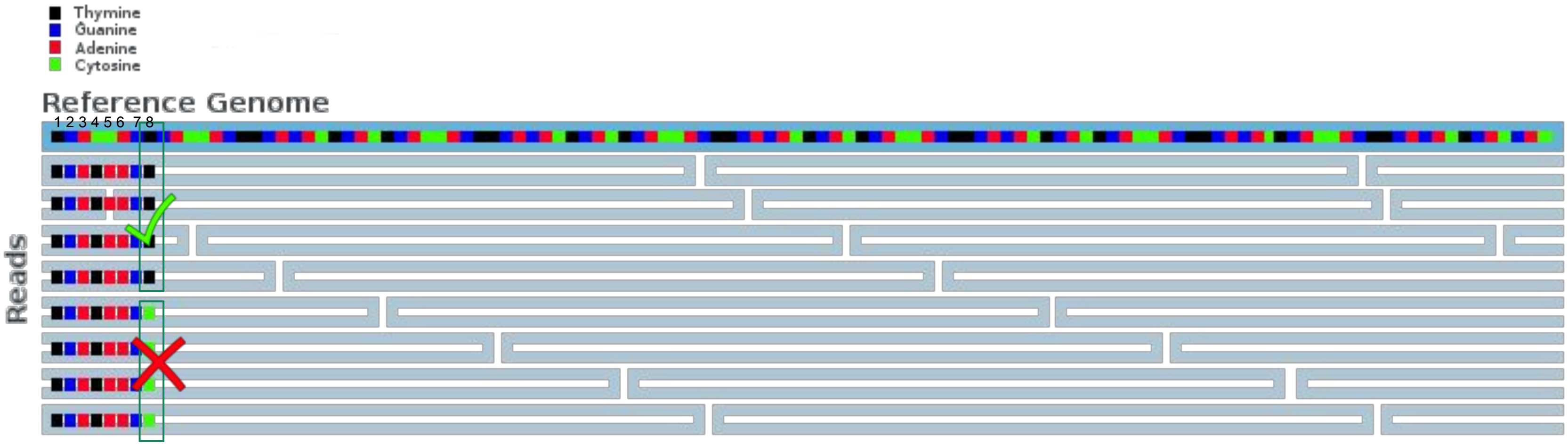
Ideal Variant Calling

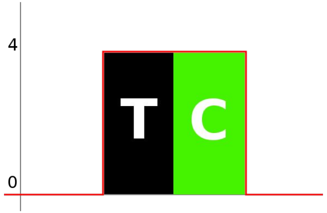




CONTIG	POS	REF	ALT	GT
X	5	**C**	**A**	**1/1**
X	6	A	-	0/0
X	7	G	-	0/0

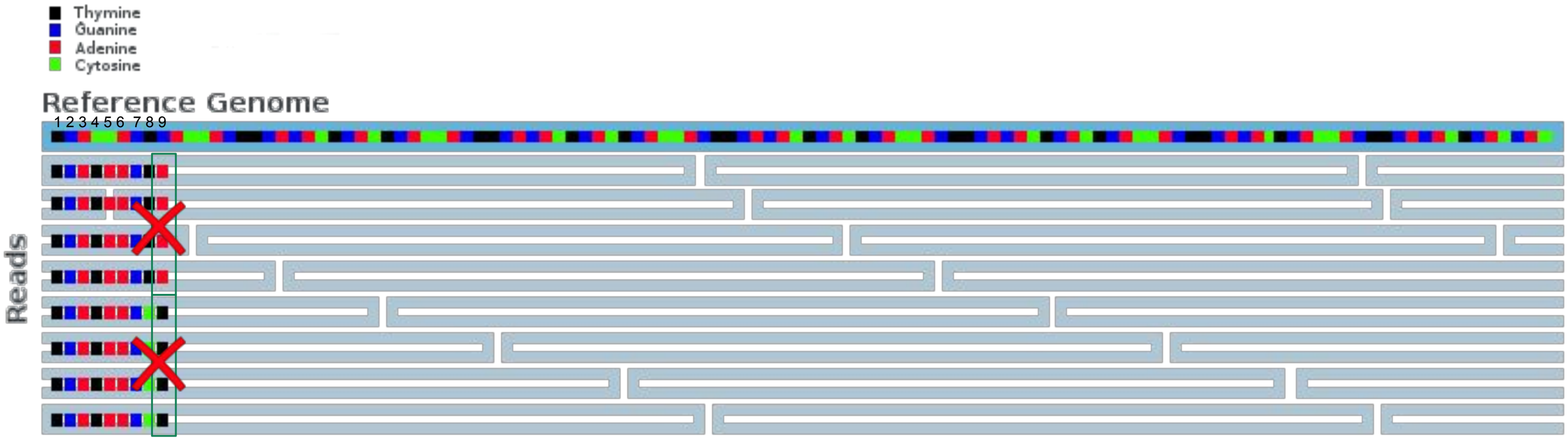
Ideal Variant Calling

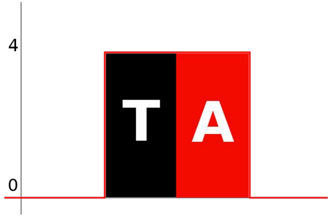




CONTIG	POS	REF	ALT	GT
X	6	A	-	0/0
X	7	G	-	0/0
X	8	**T**	**C**	**0/1**

Ideal Variant Calling



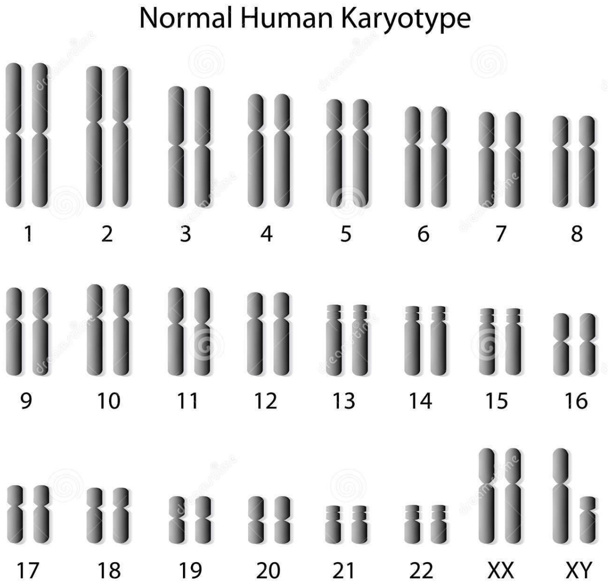


CONTIG	POS	REF	ALT	GT
X	7	G	-	0/0
X	8	**T**	**C**	**0/1**
X	9	**G**	**A,T**	**1/2**

Variant Calling

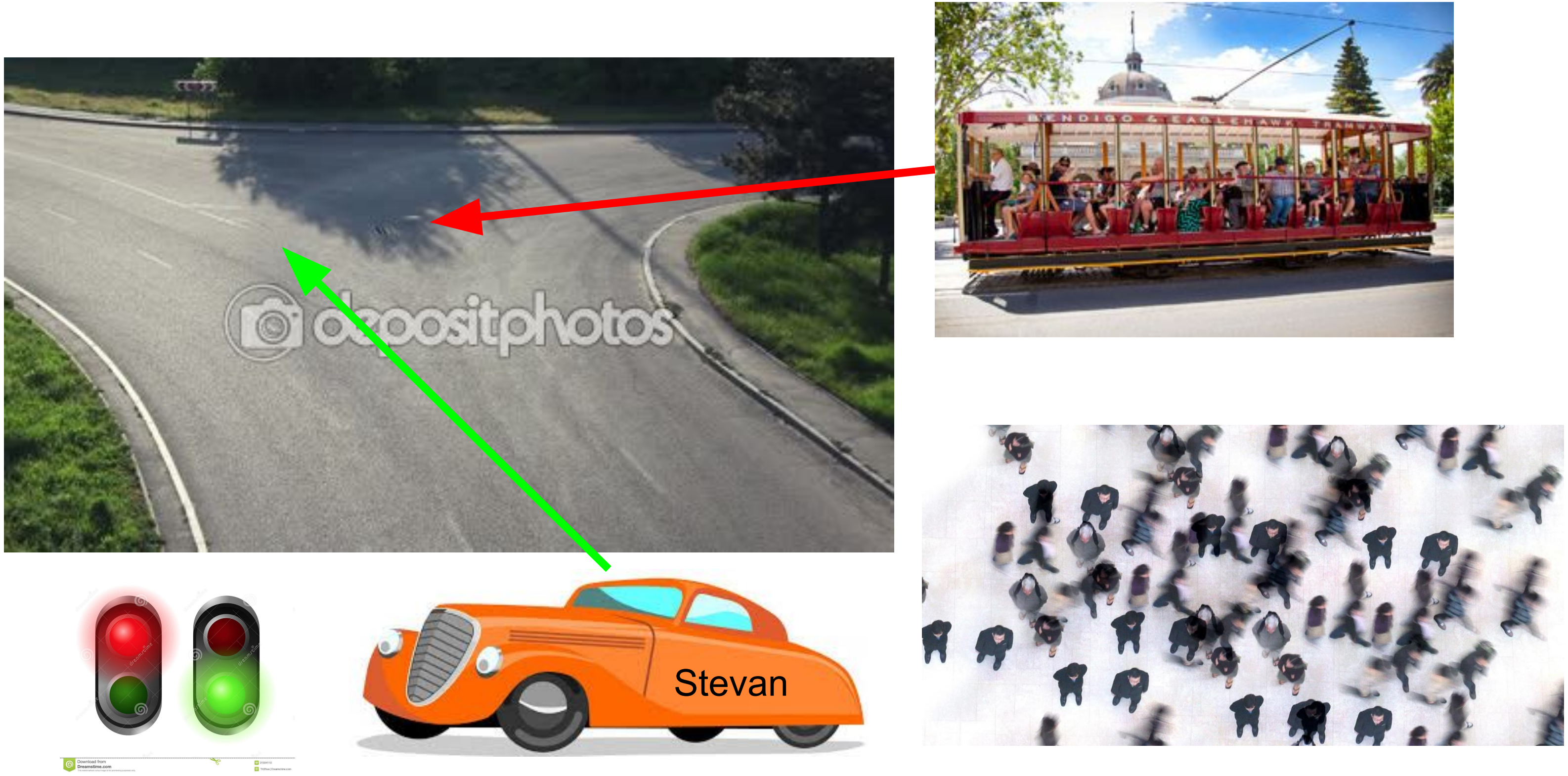
- Two possible cases:
  1. All of the bases in pileup are the same nucleotide [A,T,C,G]
  2. Different nucleotides exist in the pileup

- In the simplest case, assume diploidy
  - There can be only two alleles at a site
  - If there are more than two different letters in the pileup we will only consider the most common two (assume others are errors and discard them)



Hypothetical - traffic light problem

?

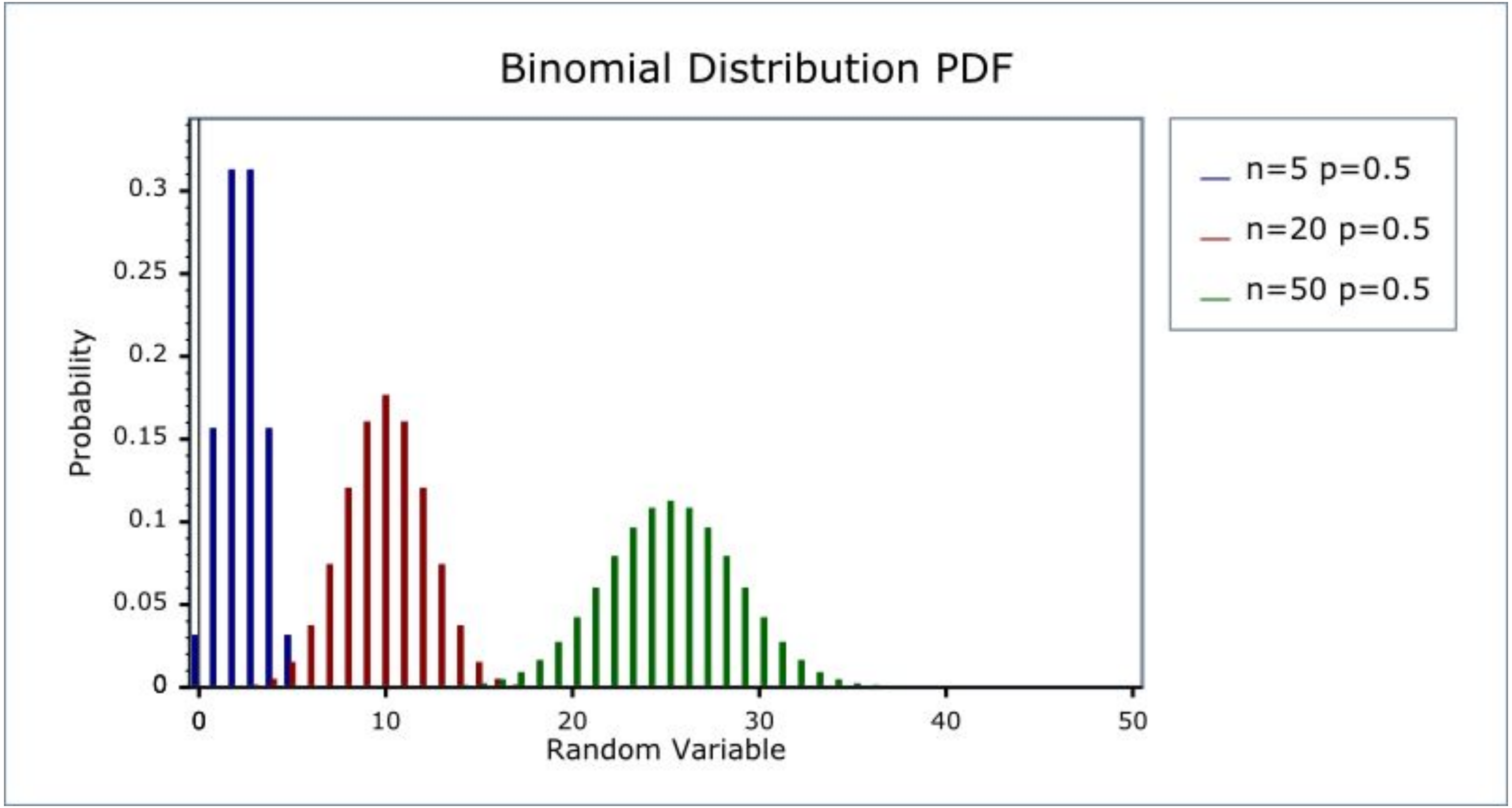


Binomial distribution

- Models the number of successes in a sequence of yes/no experiments
- Parameters:
  - n - number of trials
  - p - probability of a success in a single trial
  - Probability that K out of n trials will be success

$$f(k;n,p) = \Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

Binomial distribution



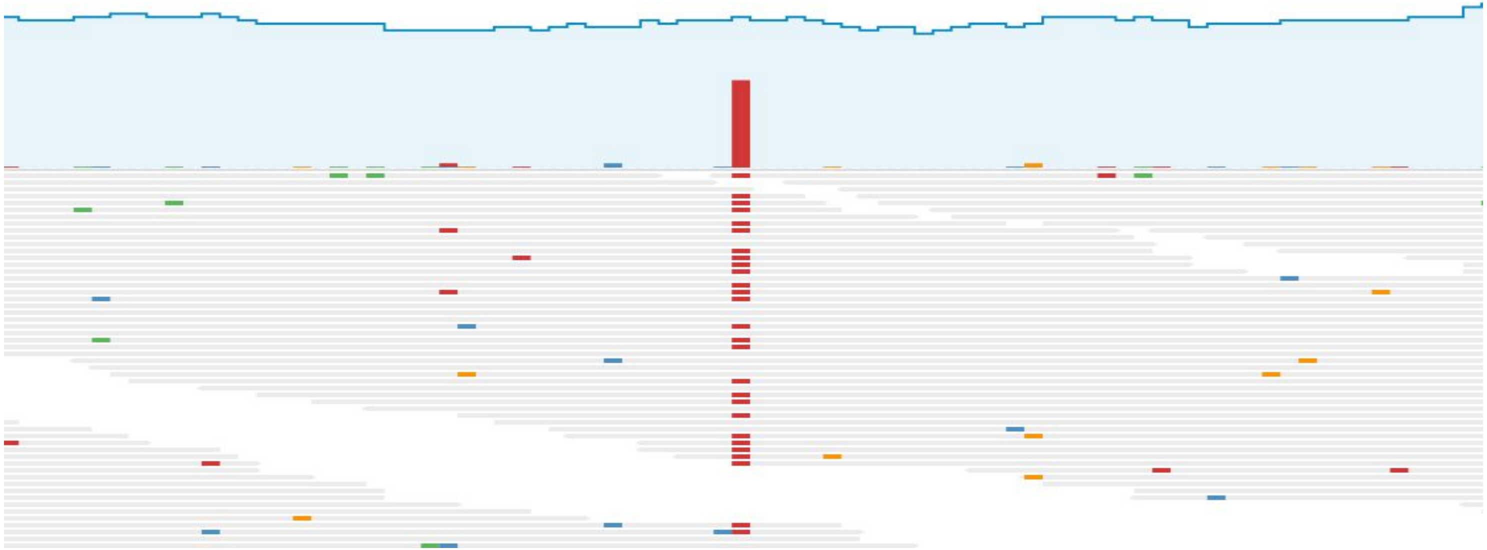
Variant Calling

- So, when we have two letters in the pileup, what should we call?
    - Let’s call the two “letters” **b** and **b’** *(b, b’ ∈ [A, C, T, G])*
    - Let **n** be the total number of bases, and **k** number of b’ bases
    - Three possible explanations for the pileup:
        - Genotype is bb; k bases are errors, n-k are correct
        - Genotype is b’b’; n-k bases are errors, k are correct
        - Genotype is bb’; all n bases are correct
    - Now we need to find the probabilities of these three cases
        - Will pick the most probable one!

Variant Calling – advance

- We assumed a flat error rate
  - But we have Base qualities from the sequencer
  - Machine-specific error profiles
- We can look at mapping qualities
  - Mapping errors are a big source of errors
- We can look at haplotypes
  - Errors don’t segregate nicely
- Population-based methods
  - Separate variant calling from genotyping

Variant calling results – check out BAM file



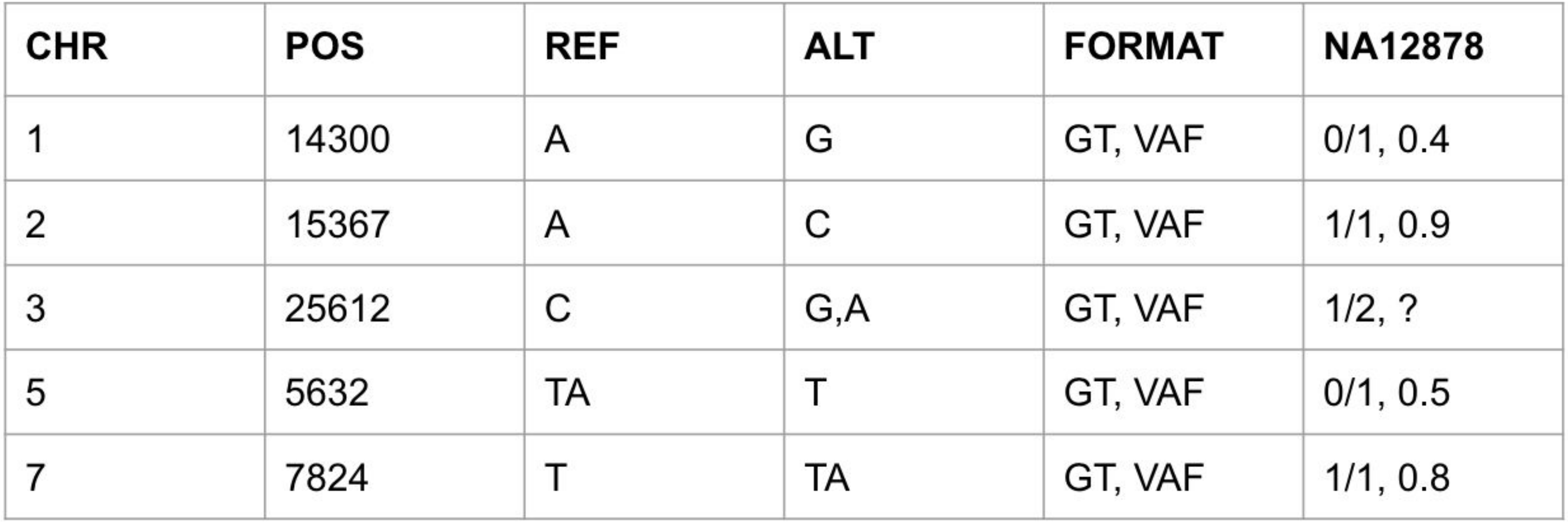
CHR	POS	REF	ALT	FORMAT	NA12877
1	14125	T	C	GT, VAF	0/1, 0.6

Variant calling results

- The result of Variant Calling is a file in VCF format, which contains mutations
- A plain text file format for storing variant data
- A number of line starting with ## -the header
- Main header line:
  #CHROM POS ID REF ALT QUAL FILTER INFO FORMAT SAMPLE1
- This is followed by the actual variant data, one entry per line
  22 10001 . A C 40 PASS DP=14 GT 0/1
- More than one sample can be in one line
- For details: http://samtools.github.io/hts-specs/VCFv4.2.pdf

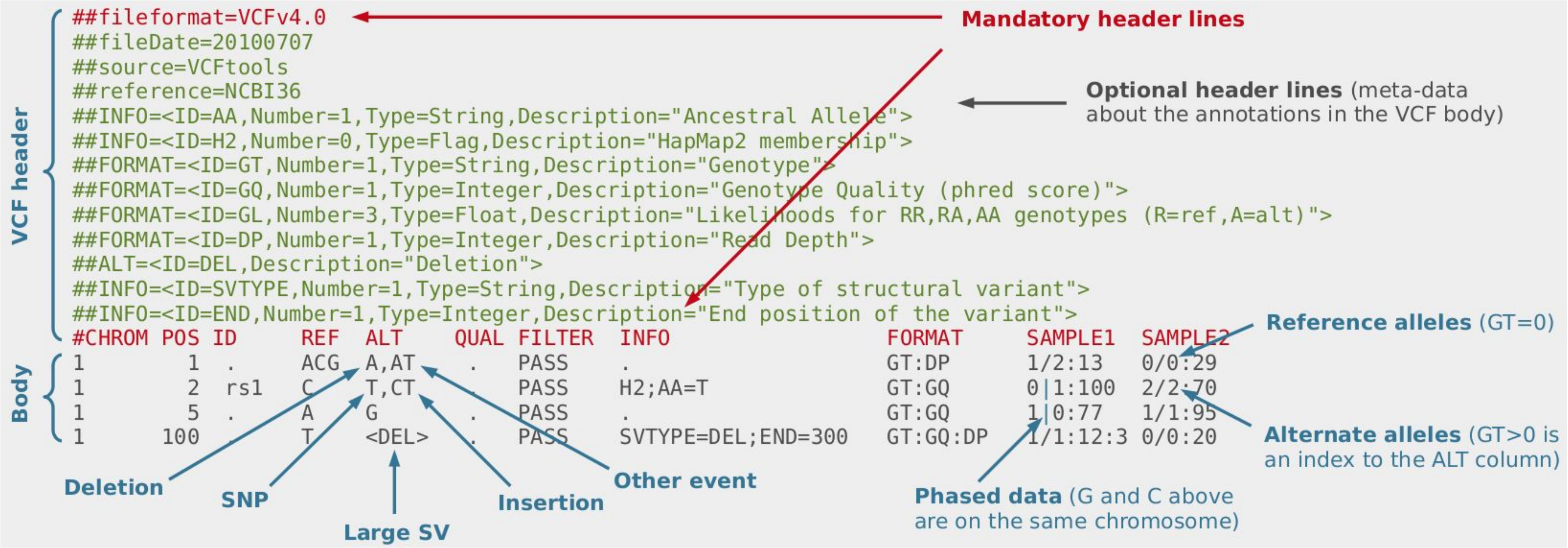
Variant calling results

- Example of VCF format
- Each row represents one mutation



CHR	POS	REF	ALT	FORMAT	NA12878
1	14300	A	G	GT, VAF	0/1, 0.4
2	15367	A	C	GT, VAF	1/1, 0.9
3	25612	C	G,A	GT, VAF	1/2, ?
5	5632	TA	T	GT, VAF	0/1, 0.5
7	7824	T	TA	GT, VAF	1/1, 0.8

Variant Calling Format File



```
##fileformat=VCFv4.0
##fileDate=20100707
##source=VCFtools
##reference=NCBI36
##INFO=<ID=AA,Number=1,Type=String,Description="Ancestral Allele">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality (phred score)">
##FORMAT=<ID=GL,Number=3,Type=Float,Description="Likelihoods for RR,RA,AA genotypes (R=ref,A=alt)">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
##ALT=<ID=DEL,Description="Deletion">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant">
#CHROM POS ID    REF  ALT    QUAL FILTER INFO                FORMAT      SAMPLE1  SAMPLE2
1        1 .     ACG  A,AT    .   PASS   .                   GT:DP       1/2:13   0/0:29
1        2 rs1   C    T,CT    .   PASS   H2;AA=T             GT:GQ       0|1:100  2/2:70
1        5 .     A    G       .   PASS   .                   GT:GQ       1|0:77   1/1:95
1      100 .     T    <DEL>   .   PASS   SVTYPE=DEL;END=300  GT:GQ:DP    1/1:12:3 0/0:20
```

Mandatory header lines

Optional header lines (meta-data about the annotations in the VCF body)

Reference alleles (GT=0)

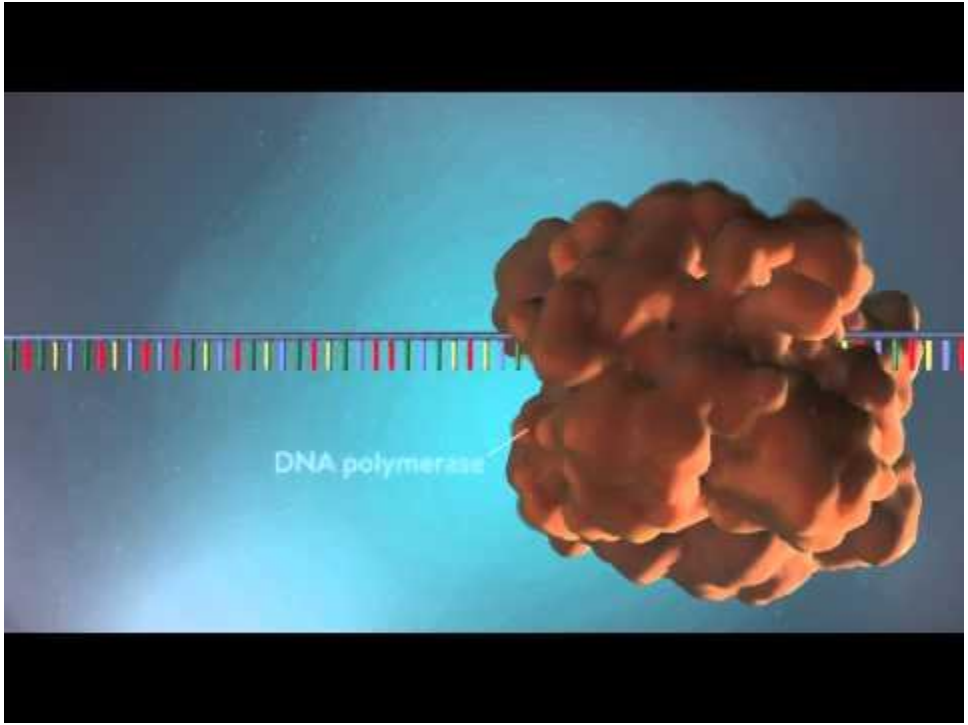
Alternate alleles (GT>0 is an index to the ALT column)

Phased data (G and C above are on the same chromosome)

Computational Cancer Analysis

Lesson 05.3

DNA replication



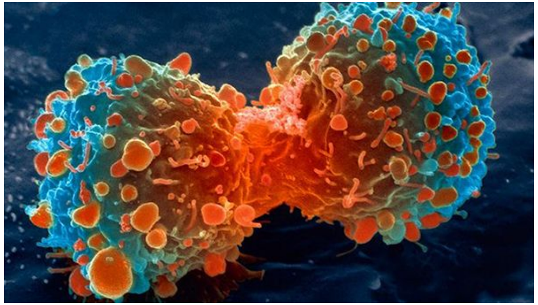
What is cancer?

Mutation during DNA replication can fall to:

1. Intron (no change)
2. Important gene (cell dies, organism lives)
3. Gene that stops cell division (cell lives, organism...)

What causes cancer (increases probability of mutation)?

1. EM radiation
2. Chemical agents
3. Free radicals
4. Genetic factors
5. Infections (viruses)



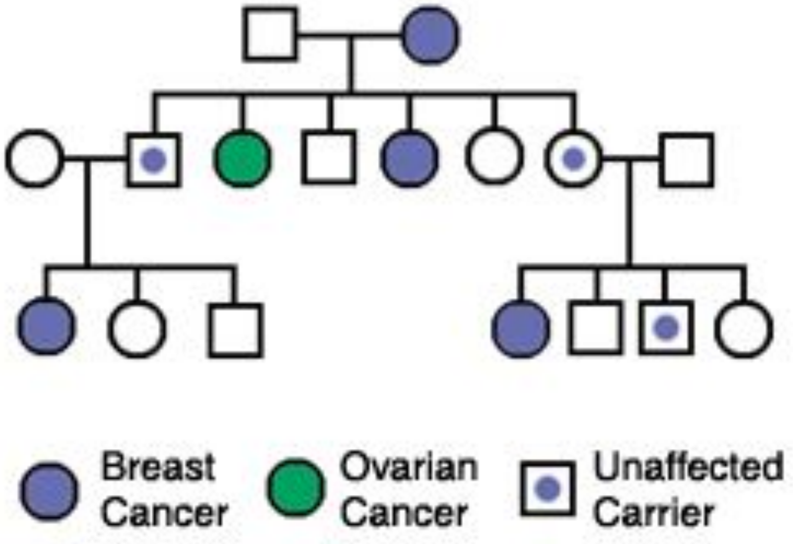
A dividing lung cancer cell.
Credit: National Institutes of Health

Genetic factors

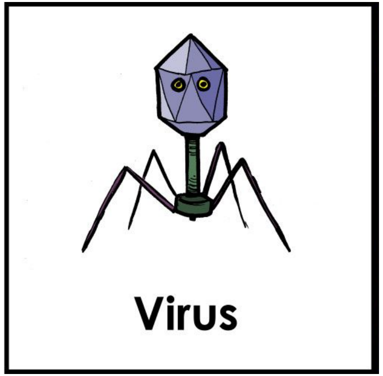
A typical pedigree from a family with a mutation in the BRCA1 (tumor suppressor) gene

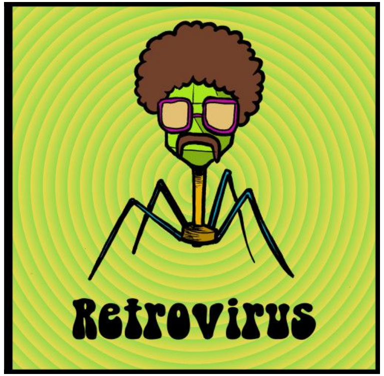
Fathers can be carriers and pass the mutation onto offspring

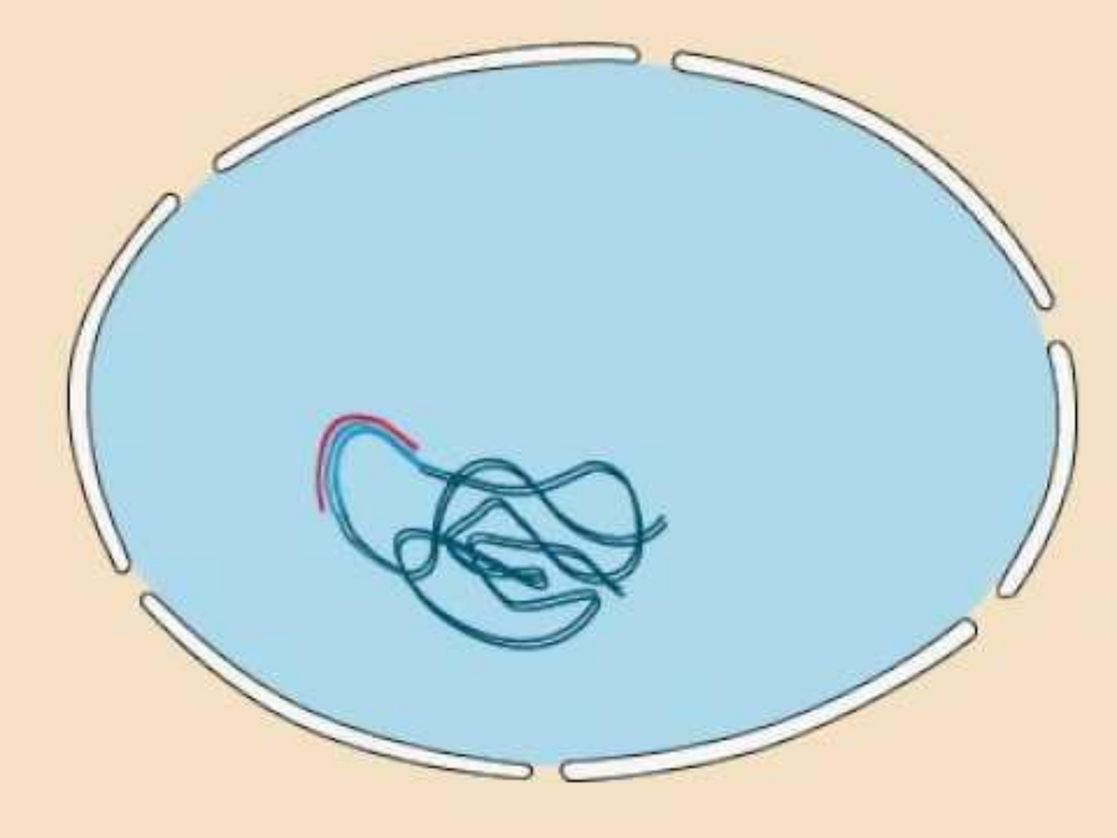
Not all people who inherit the mutation develop the disease, thus patterns of transmission are not always obvious



Viruses and retroviruses



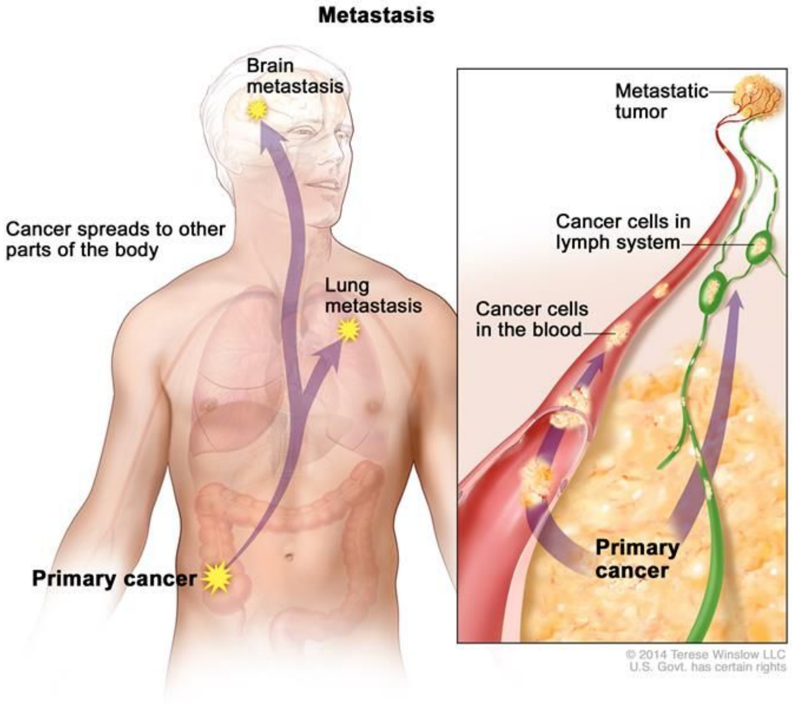




What is metastasis?

Body’s cells begin to divide without stopping and spread into surrounding tissues

Cancer cells - ignore signals that normally tell cells to stop dividing or that begin a process known as programmed cell death, or **apoptosis**, which the body uses to get rid of unneeded cells



Metastasis
Brain metastasis
Metastatic tumor
Cancer spreads to other parts of the body
Cancer cells in lymph system
Lung metastasis
Cancer cells in the blood
Primary cancer
Primary cancer

© 2014 Terese Winslow LLC
U.S. Govt. has certain rights

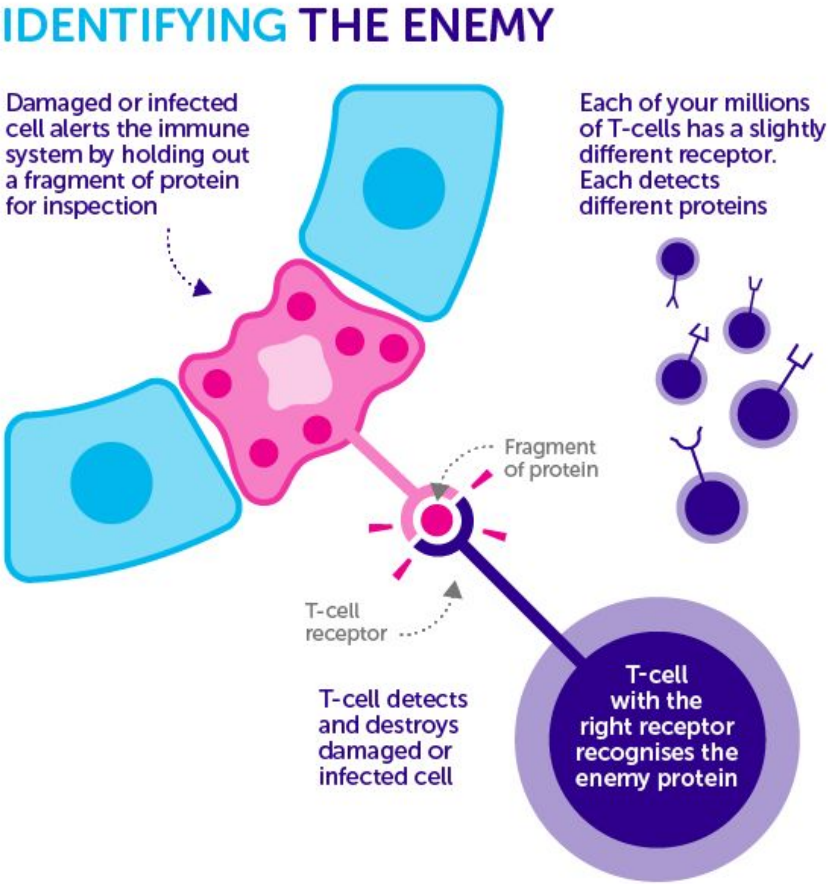
"Drivers" of Cancer

Cancer is a genetic disease that is caused by changes to genes which control the way our cells function, especially how they grow and divide:

1. **Abnormal growth (proto-oncogenes)** Cellular growth mechanism is damaged and cell starts to multiply uncontrollably
2. **Damaged control mechanism** (tumor suppressors) - Cells with certain alterations in tumor suppressor genes may divide in an uncontrolled manner (TP53 - Apoptosis)
3. **Damaged DNA repair mechanism** (Accumulated errors in this group of genes can lead to uncontrollable proliferation)

Cancer cells

Our body develops thousands cancer cells every day! OMG OMG

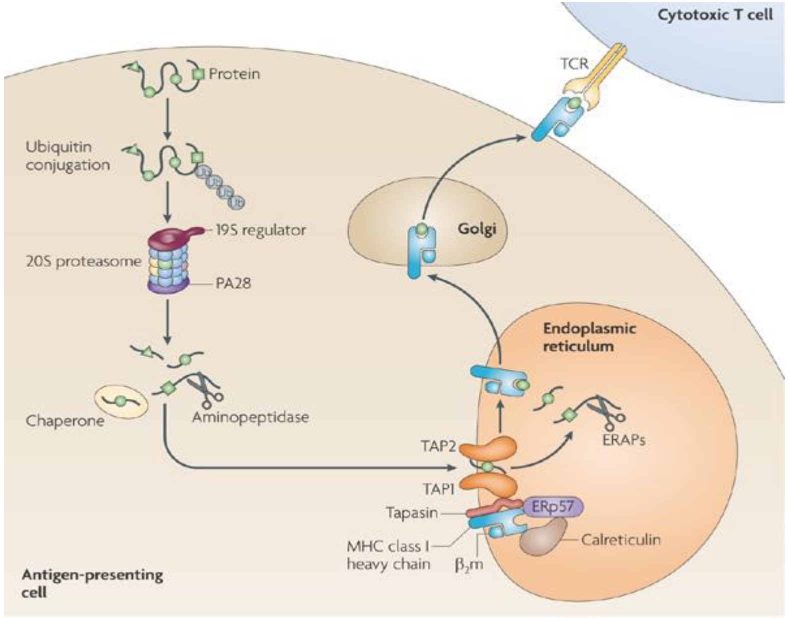

IDENTIFYING THE ENEMY
Damaged or infected cell alerts the immune system by holding out a fragment of protein for inspection
Each of your millions of T-cells has a slightly different receptor. Each detects different proteins
Fragment of protein
T-cell receptor
T-cell detects and destroys damaged or infected cell
T-cell with the right receptor recognises the enemy protein

MHC Complex

MHC is a set of cell surface proteins essential for the acquired immune system to recognize foreign molecules (translated from HLA regions from the genome for humans)

MHC molecules bind to **protein fragments available in the cell**

MHC molecule with **antigen** (MHC complex) is "presented" outside of the cell to cytotoxic T cells and helper T cells



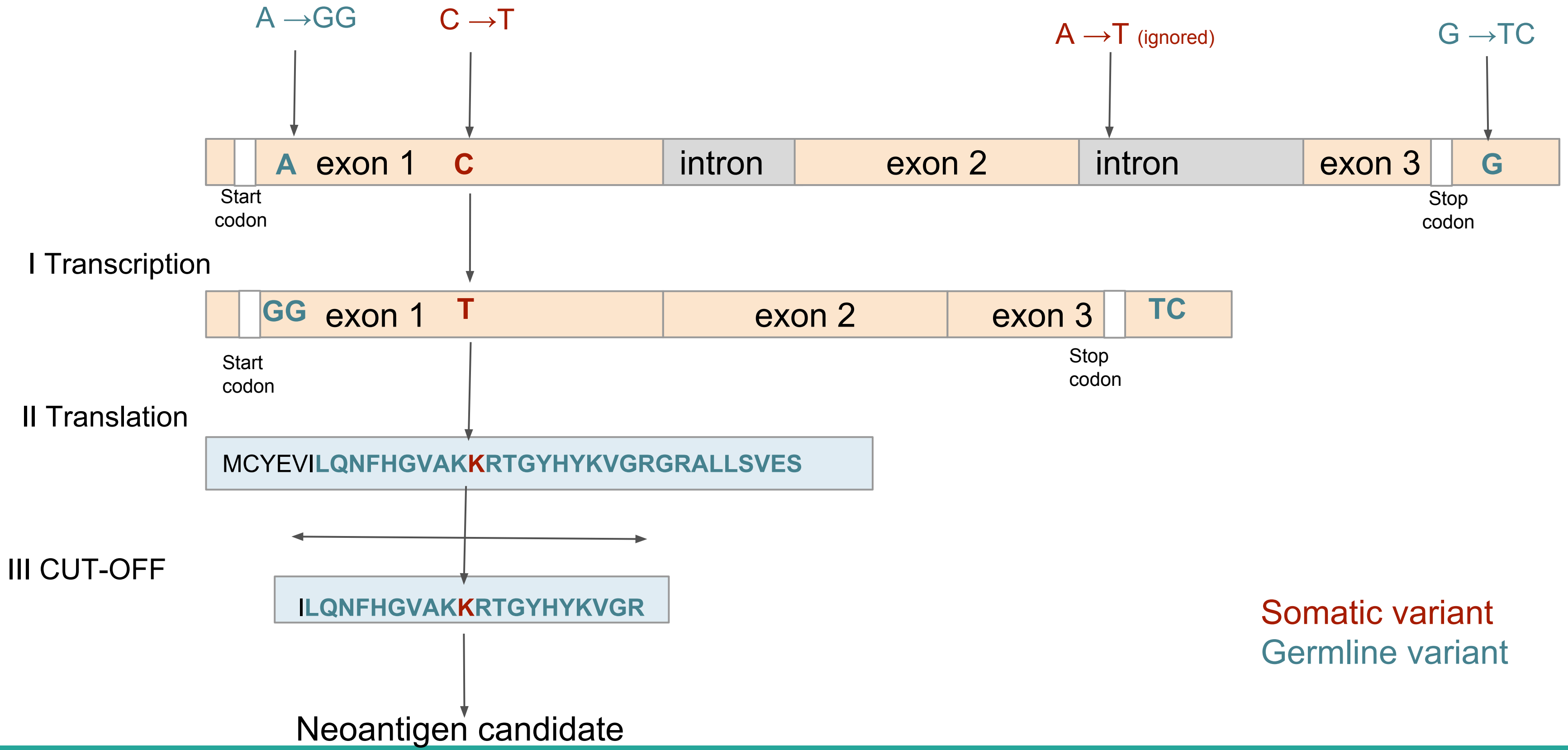
Nature Reviews | Immunology

So, what can be done there?

1. Identify NEOANTIGENS - proteins presented only by cancer cells
2. “Program” T-cells to recognize neoantigens

Compare DNA from Tumor and Normal tissue
Mutations present in tumor - somatic mutations

From DNA somatic mutation to neoantigen



A →GG

C →T

A →T (ignored)

G →TC

A exon 1 C

intron

exon 2

intron

exon 3

G

Start codon

Stop codon

I Transcription

GG exon 1 T

exon 2

exon 3

TC

Start codon

Stop codon

II Translation

MCYEVILQNFHGVAKKRTGYHYKVGRGRALLSVES

III CUT-OFF

ILQNFHGVAKKRTGYHYKVGR

Neoantigen candidate

Somatic variant

Germline variant

References

How the Immune System Works - Lauren Sompayrac