Bioinformatics in Next-Generation Sequencing

Elena Nabieva, Ph.D.

enabieva@gmail.com

Outline

- Review of molecular biology
  - What is DNA sequencing?
- Bioinformatics
  - Read mapping
  - Variant calling
  - Variant evaluation

Molecular biology and genetics

A very brief review

DNA molecule

- “Blueprint” for making our bodies
- Long polymer of repeating nucleotides
  - Adenine, Cytosine, Guanine, Thymine
- Double helix:
  - Adenine – Thymine
  - Guanine – Cytosine

  This enables replication of the molecule
- For computation: string over alphabet {A, C, G, T}
  - Entire string: *genome*

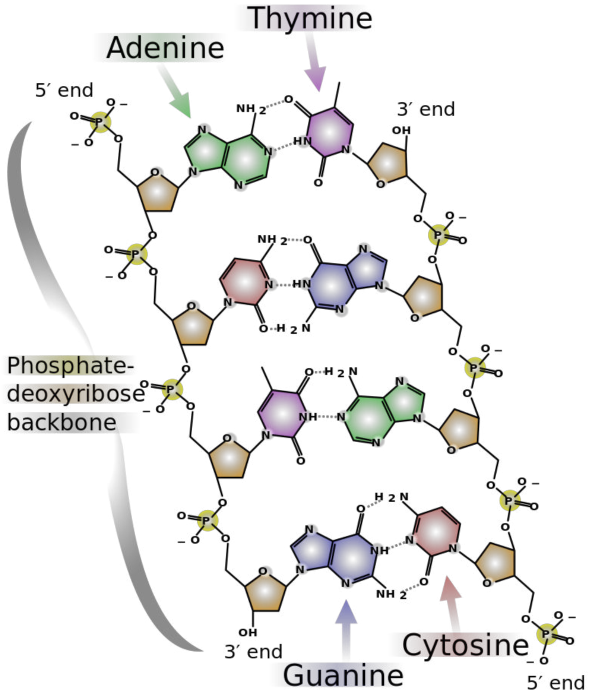
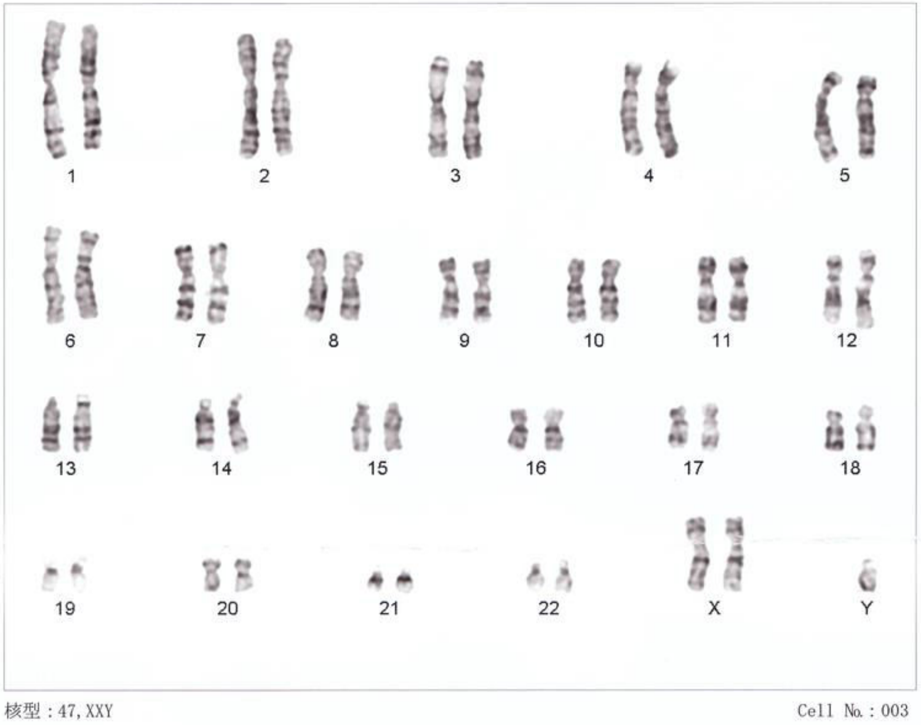


Image by Madprime https://commons.wikimedia.org/w/index.php?curid=1848174

DNA organized into chromosomes

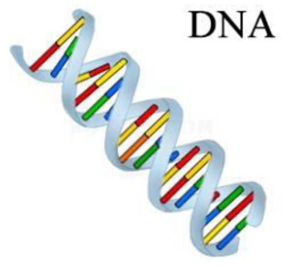
- 22 *autosomes*
  - come in pairs
    - mother’s
    - father’s

  = humans are *diploid*
- Sex chromosomes:
    - X, Y
  - Female: XX
  - Male: XY
- Consequence:
  - Two copies of every autosomal gene
  - Females have two copies of X-chr genes, males have one
- Sometimes: deviation from two copies
  - Down’s syndrome, X-, Y-polysomies
  - cancer



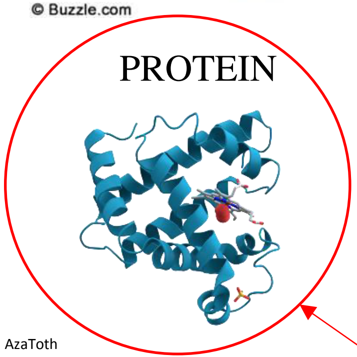
https://commons.wikimedia.org/wiki/File:Human_chromosomesXXY01.png

Central Dogma of Molecular Biology

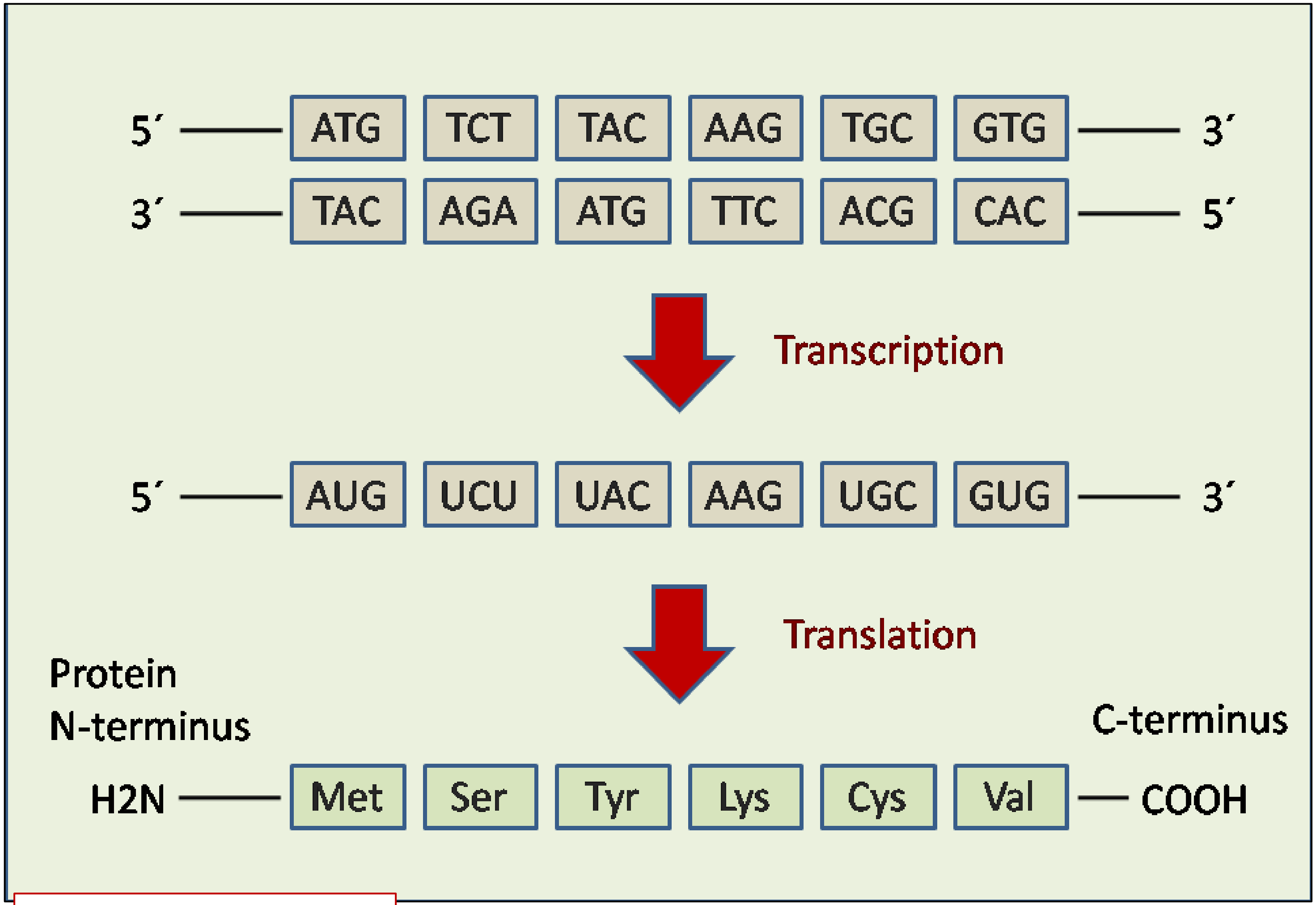


© Buzzle.com

© Buzzle.com



AzaToth
https://commons.wikimedia.org/w/index.php?curid=68596



5′ —— ATG TCT TAC AAG TGC GTG —— 3′

3′ —— TAC AGA ATG TTC ACG CAC —— 5′

Transcription

5′ —— AUG UCU UAC AAG UGC GUG —— 3′

Translation

Protein
N-terminus

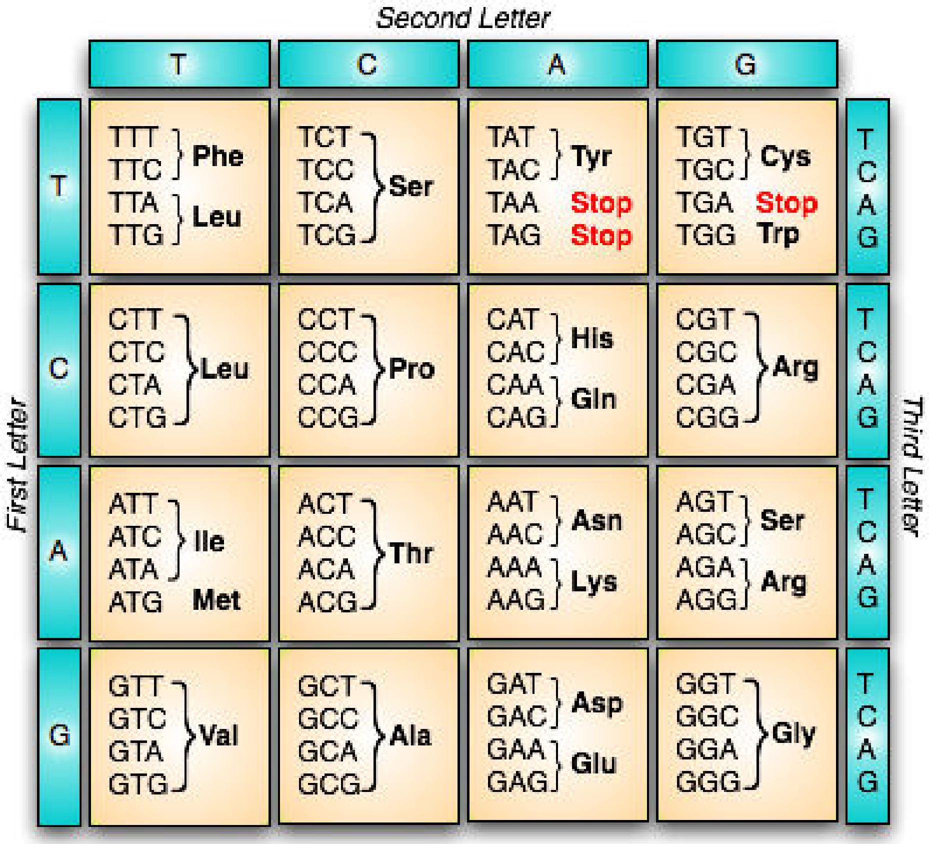
C-terminus

H2N —— Met Ser Tyr Lys Cys Val — COOH

This matters most

http://creationwiki.org/

Genetic code



Second Letter

First Letter	T	C	A	G	Third Letter
T	TTT TTC } Phe TTA TTG } Leu	TCT TCC TCA TCG } Ser	TAT TAC } Tyr TAA Stop TAG Stop	TGT TGC } Cys TGA Stop TGG Trp	T C A G
C	CTT CTC CTA CTG } Leu	CCT CCC CCA CCG } Pro	CAT CAC } His CAA CAG } Gln	CGT CGC CGA CGG } Arg	T C A G
A	ATT ATC ATA } Ile ATG Met	ACT ACC ACA ACG } Thr	AAT AAC } Asn AAA AAG } Lys	AGT AGC } Ser AGA AGG } Arg	T C A G
G	GTT GTC GTA GTG } Val	GCT GCC GCA GCG } Ala	GAT GAC } Asp GAA GAG } Glu	GGT GGC GGA GGG } Gly	T C A G

- Protein code
  - 20 amino acids
  - STOP codon
    - signal to stop translation

⇒Redundancy in genetic code

⇒*reading frame* matters!

Mutations in DNA

- Point mutations (single nucleotide change)
  - Synonymous: no AA change
    - e.g., TCA -> TCC (Ser -> Ser)
  - Nonsynonymous: AA change
    - e.g., TCA->CCA (Ser->Pro)
  - *Nonsense*: introduce new STOP (*) codon

    => premature termination of translation
    - e.g., TCA->TAA (Ser->*)
- Short insertions or deletions (indels)

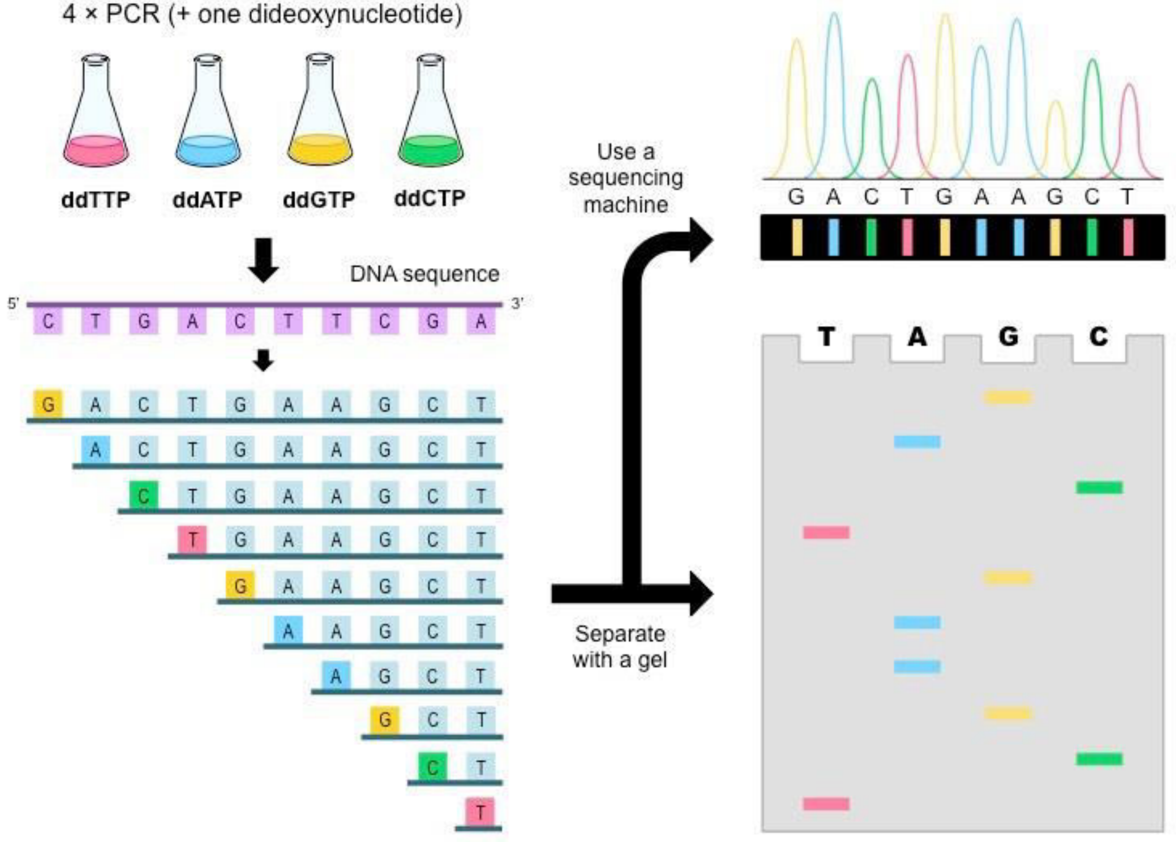
  e.g. TCACCATCG -> TCACATCG
  - *in-frame*: multiple of 3, preserves reading frame
  - *frameshift:* not multiple of 3, disrupts reading frame
- Multinucleotide substitutions
- Large-scale copy number variation

DNA sequencing

Some questions to answer

- What genetic variants does this genome have?
  - What variants does a particular person have?
  - This person has a genetic disease. What is the cause?
  - What mutations do we find in a tumor compared to normal tissue?
- What do these variants mean?
  - Will/did this variant cause disease?
  - Does this mutation in the tumor *drive* the cancer or is it a *passenger?*

Fred Sanger, 1970s



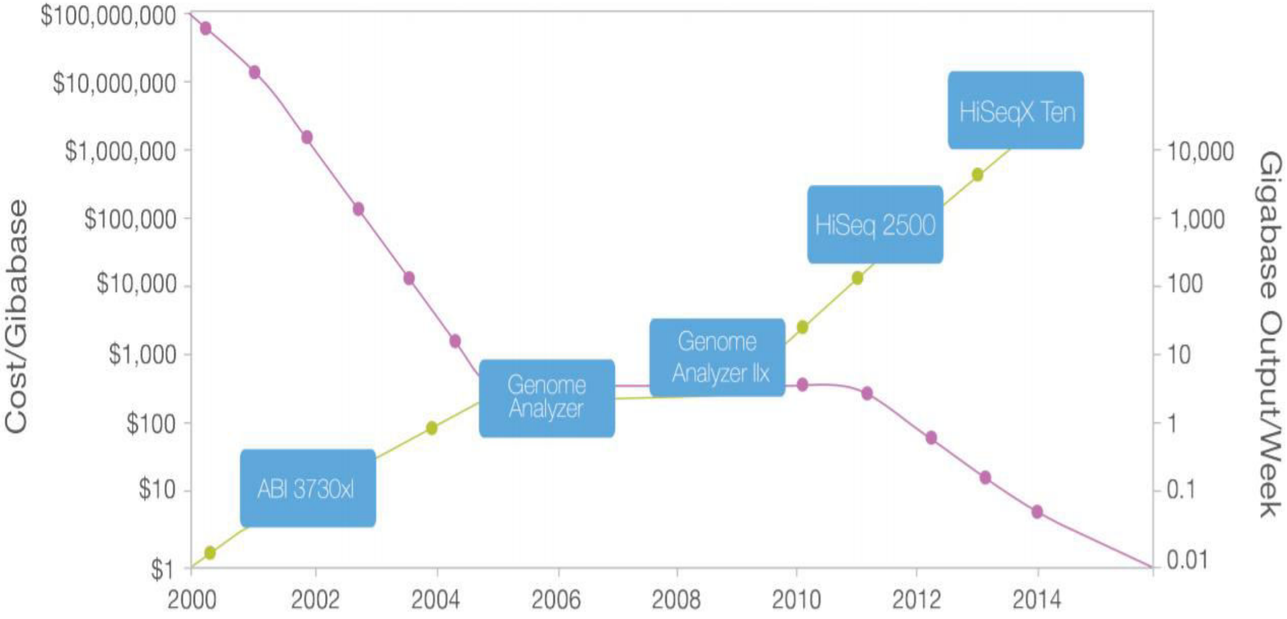
http://ib.bioninja.com.au/

- Can sequence up to 1000 nucleotide-long segments
- But the human genome is over 3 billion nucleotides long
- Note: Sanger sequencing used for NGS variant validation

Next-Generation Sequencing

- Massively parallel

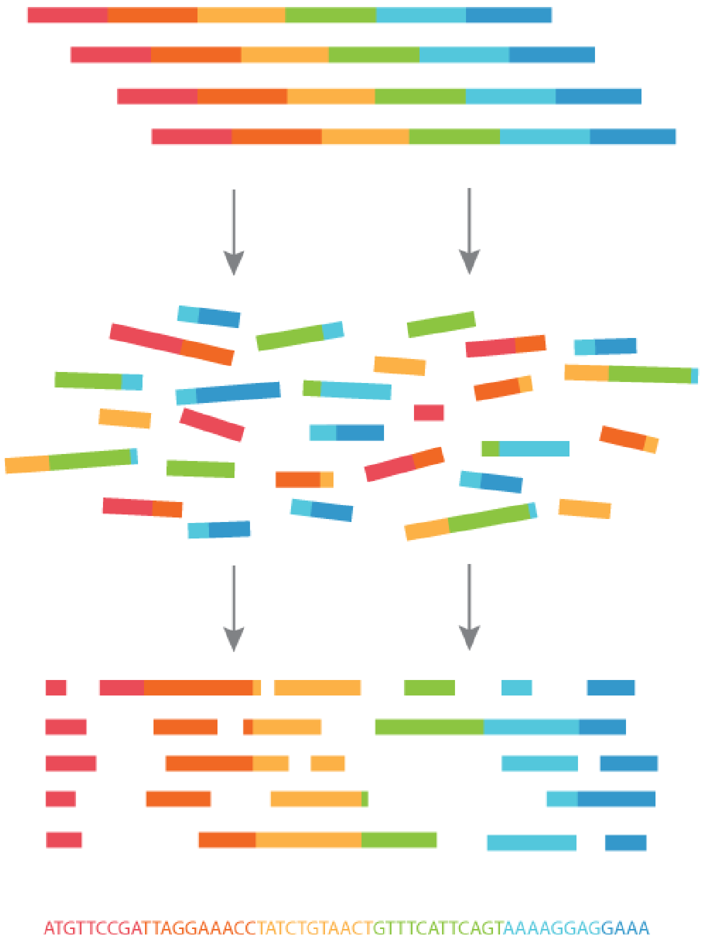
Illumina HiSeq 2000



http://www.illumina.com/technology/next-generation-sequencing.html

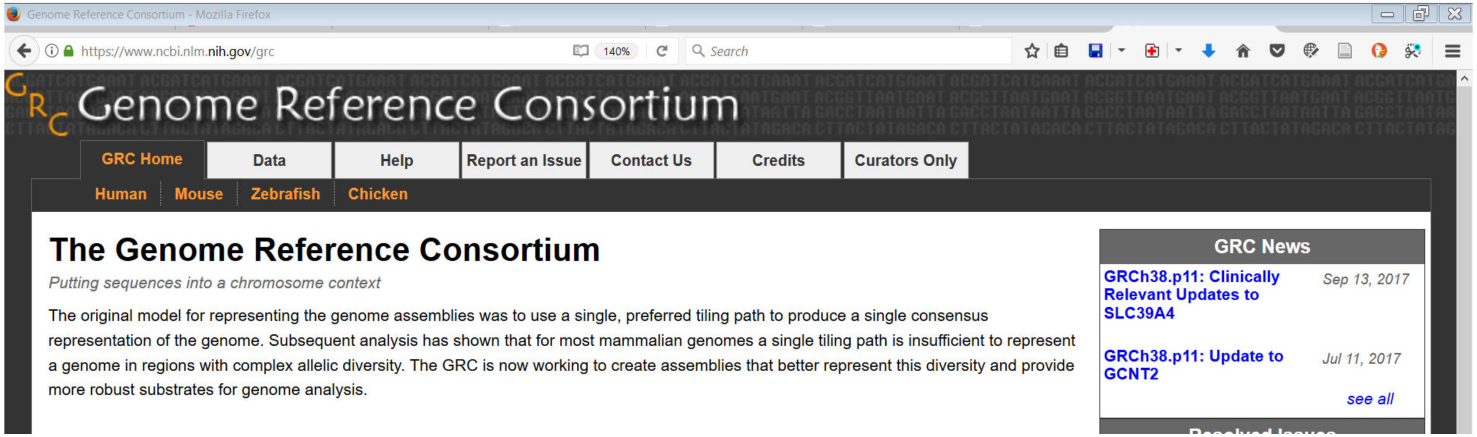
Next-Generation Sequencing

- Randomly “tear” DNA into short fragments
- Sequence fragments (“reads”)
- Do bioinformatics



Sequence assembly

- Many overlapping short reads => try to assemble the genome
- Important bioinformatics problem
  - beyond the scope of this lecture
- For human genome: (mostly) solved
    - though resurfaces in “local assembly” in variant calling
  - *Reference human genome*


Genome Reference Consortium - Mozilla Firefox
https://www.ncbi.nlm.nih.gov/grc
140%
Search
GRC
Genome Reference Consortium
GRC Home
Data
Help
Report an Issue
Contact Us
Credits
Curators Only
Human
Mouse
Zebrafish
Chicken

The Genome Reference Consortium

Putting sequences into a chromosome context

The original model for representing the genome assemblies was to use a single, preferred tiling path to produce a single consensus representation of the genome. Subsequent analysis has shown that for most mammalian genomes a single tiling path is insufficient to represent a genome in regions with complex allelic diversity. The GRC is now working to create assemblies that better represent this diversity and provide more robust substrates for genome analysis.

GRC News

GRCh38.p11: Clinically Relevant Updates to SLC39A4 — *Sep 13, 2017*

GRCh38.p11: Update to GCNT2 — *Jul 11, 2017*

see all

The reference genome

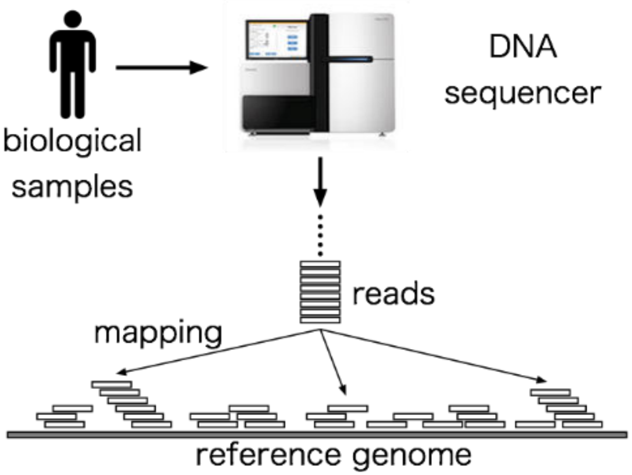
- Haploid (single-copy) sequence
- May be based on multiple individuals
- Not necessarily an “ideal” genome
    - May contain rare/deleterious alleles (versions of genes)
    - GRC tries to replace rare alleles with common ones
- Missing knowledge
    - Chromosomal segments with unknown sequence: "NNNNN"
    - Sequences that could not yet be placed on a chromosome : chr*_random, chrUn*

If we have a reference genome

- *Map* the reads to the reference
- *Call* variants (difference from reference)
- *Evaluate* the functional significance of the variants

Read mapping problem

- For every short read, find its location on the reference genome
- Potential difficulties:
  - Sequencing errors
    - Illumina: ~0.1%
  - Natural variation
    - ~0.1% for European population
  - =>reads may not align exactly



https://julialang.org

A step back: sequence alignment

- Given two sequences, align them optimally
  - given a scoring scheme
- Example

	Penalty
Gap	2
Mismatch	1
Match	0

Sequences:
AACAGTTACC
TAAGGTCA

Seq 1	A	A	C	A	G	T	T	A	C	C
Seq 2	T	A	A	G	G	T	C	A	-	-
Penalty	1	0	1	1	0	0	1	0	2	2

Penalty: 8

Seq 1	A	A	C	A	G	T	T	A	C	C
Seq 2	T	A	-	A	G	G	T	-	C	A
Penalty	1	0	2	0	0	1	0	2	0	1

Penalty: 7

Dynamic programming: Needleman-Wunsch algorithm

```
opt[i][j] = min {
    opt[i+1][j+1] + 0/1,
    opt[i+1][j] + 2,
    opt[i][j+1] + 2 }
```

O(mn)

Variation: *local* alignment Smith-Waterman algorithm

```
           |  0   1   2   3   4   5   6   7   8
       x\y |  T   A   A   G   G   T   C   A   -
-----------------------------------------------
 0    A    |  7   8  10  12  13  15  16  18  20
 1    A    |  6   6   8  10  11  13  14  16  18
 2    C    |  6   5   6   8   9  11  12  14  16
 3    A    |  7   5   4   6   7   9  11  12  14
 4    G    |  9   7   5   4   5   7   9  10  12
 5    T    |  8   8   6   4   4   5   7   8  10
 6    T    |  9   8   7   5   3   3   5   6   8
 7    A    | 11   9   7   6   4   2   3   4   6
 8    C    | 13  11   9   7   5   3   1   3   4
 9    C    | 14  12  10   8   6   4   2   1   2
10    -    | 16  14  12  10   8   6   4   2   0
```

Adapted from http://www.cs.princeton.edu/courses/archive/spr05/cos126/assignments/sequence.html

Alignment of short reads to reference genome (mapping)

- One LONG Reference genome (~3 x10^9)
  - Can preprocess
- MANY short reads
  - 10s/100s of millions
  - Illumina HiSeq 2000: 100nt reads, other technologies produce longer reads

Approaches:

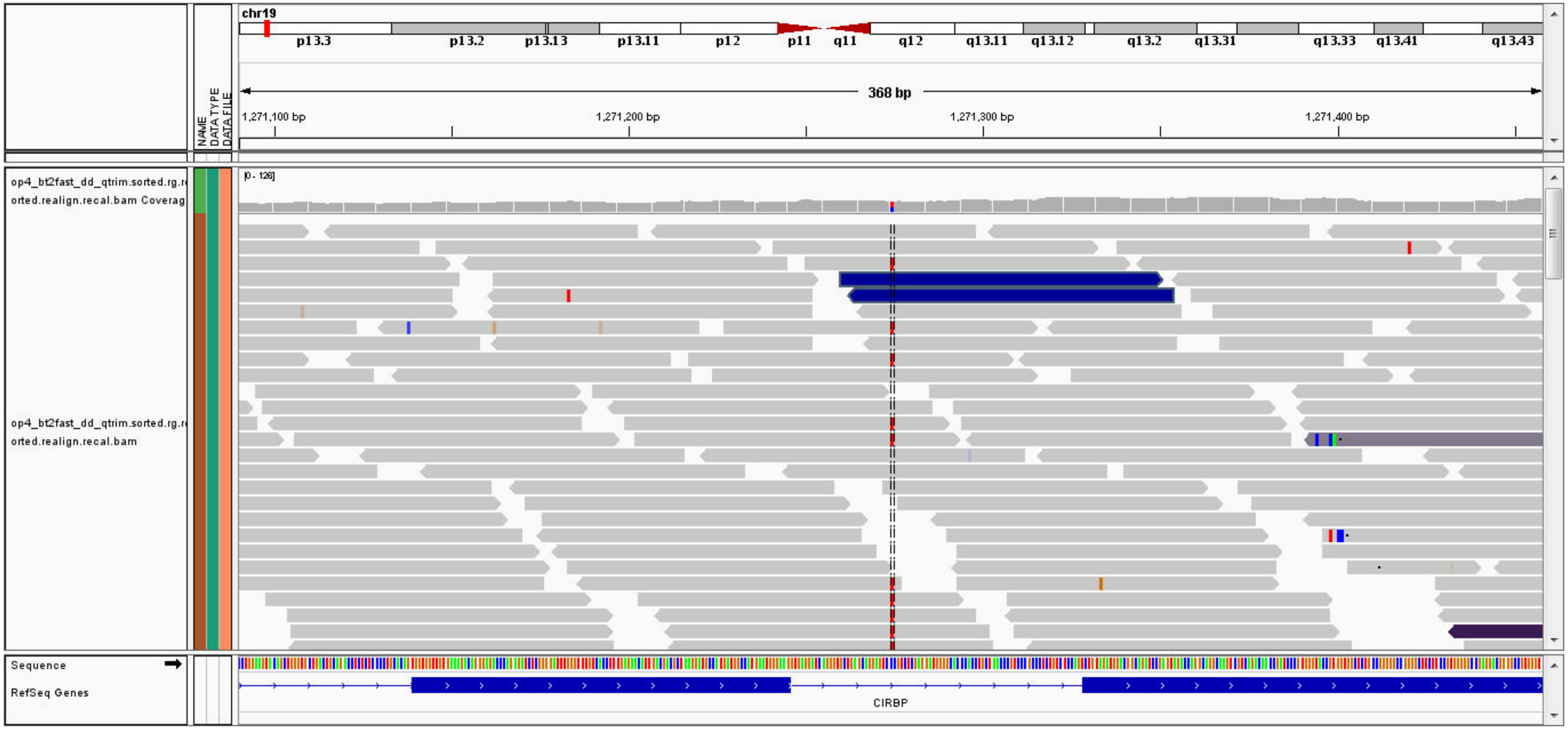
- Hash-based
  - Use hashing to find occurrence of seed, then extend
- Burrows-Wheeler Transform-based
  - cf. suffix arrays

BWT-based programs

- Widely used
- First, build (or download) index for the genome
- bowtie:
  - bowtie (1) (Langmead, Trapnell, Pop, Salzberg 2009)
    - short reads, no gaps
  - bowtie2 (Langmead, Salzberg 2012)
    - longer reads, gaps allowed
- bwa:
  - backtrack (Li, Durbin 2009)
    - reads up to 100 nt
  - bwasw (Li, Durbin 2010)
    - longer reads: 70 nt-1M nt, gaps allowed
  - mem (Li 2013) [arXiv:1303.3997v2]
    - reads of length 70 nt-1M nt
    - seems to be most popular now

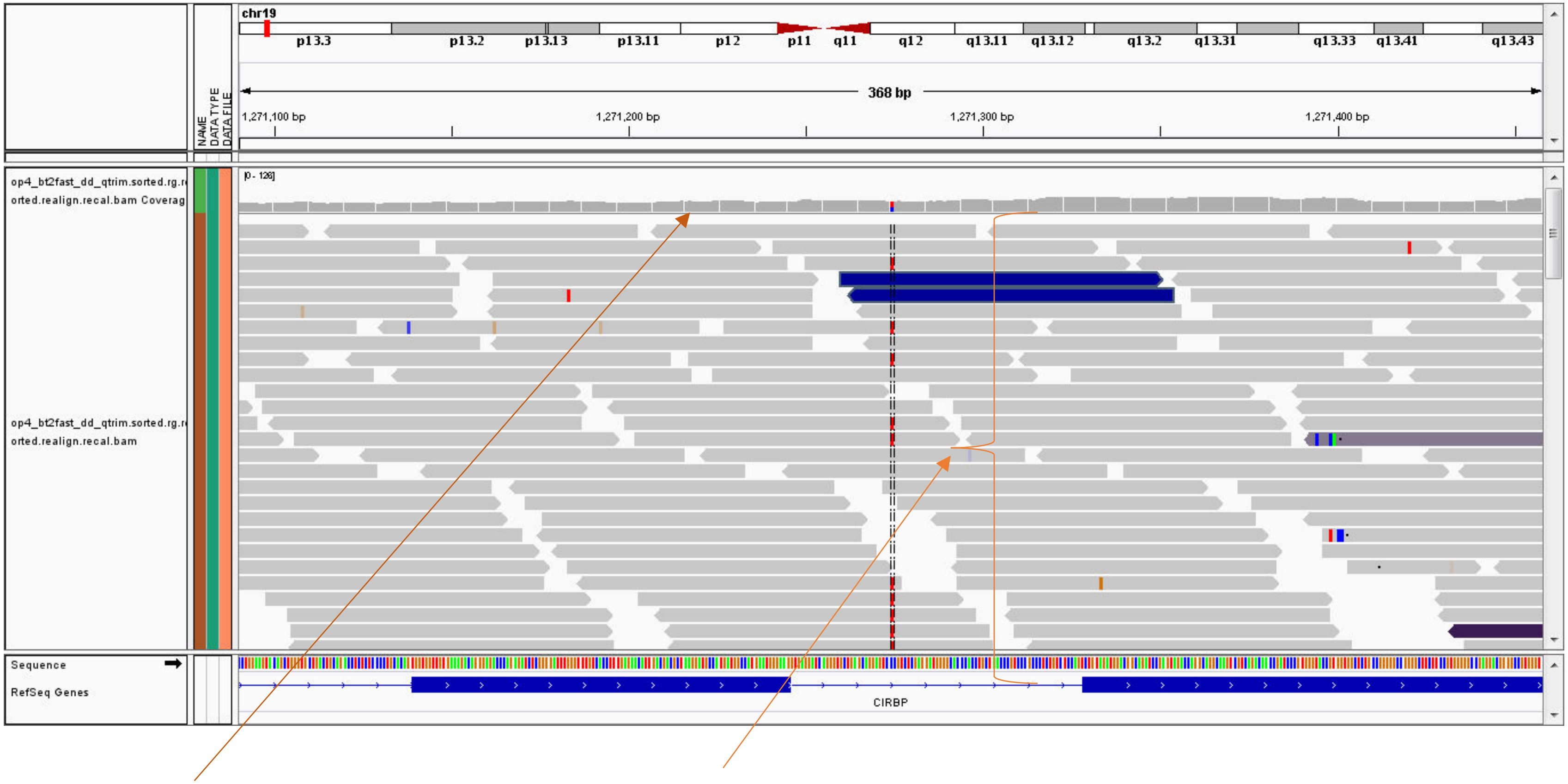
After mapping

BAM file, as viewed through the Integrated Genome Viewer (IGV)



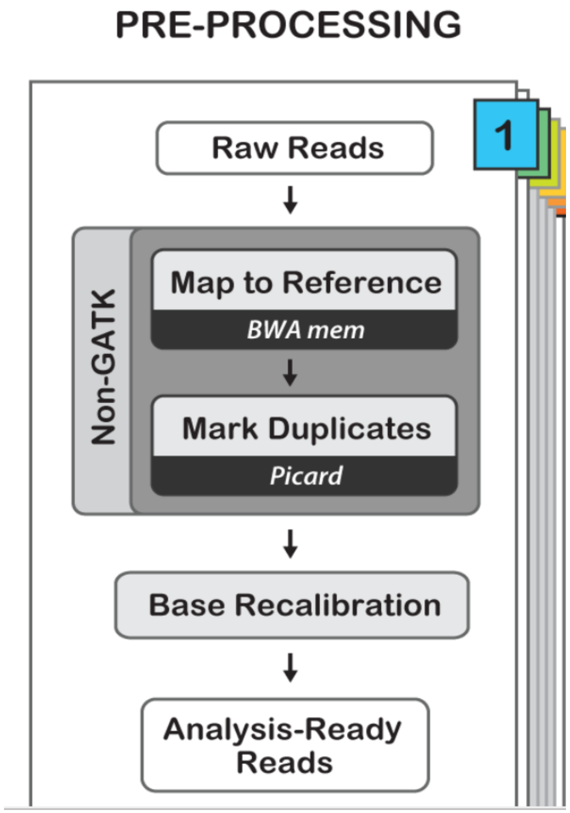
After mapping

BAM file, as viewed through the Integrated Genome Viewer (IGV)



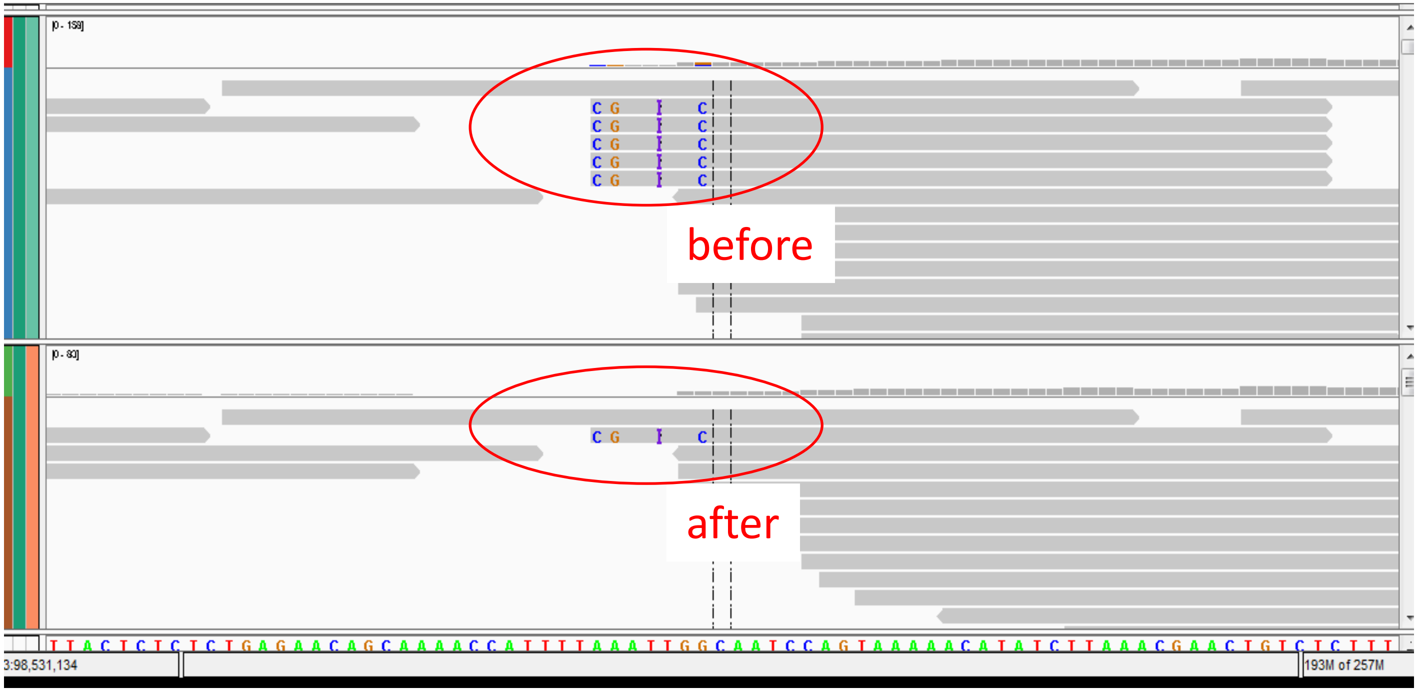
Depth of coverage: how high is the "pile" of reads?
What is the average coverage of the sample?

Post-mapping: GATK* Best Practices

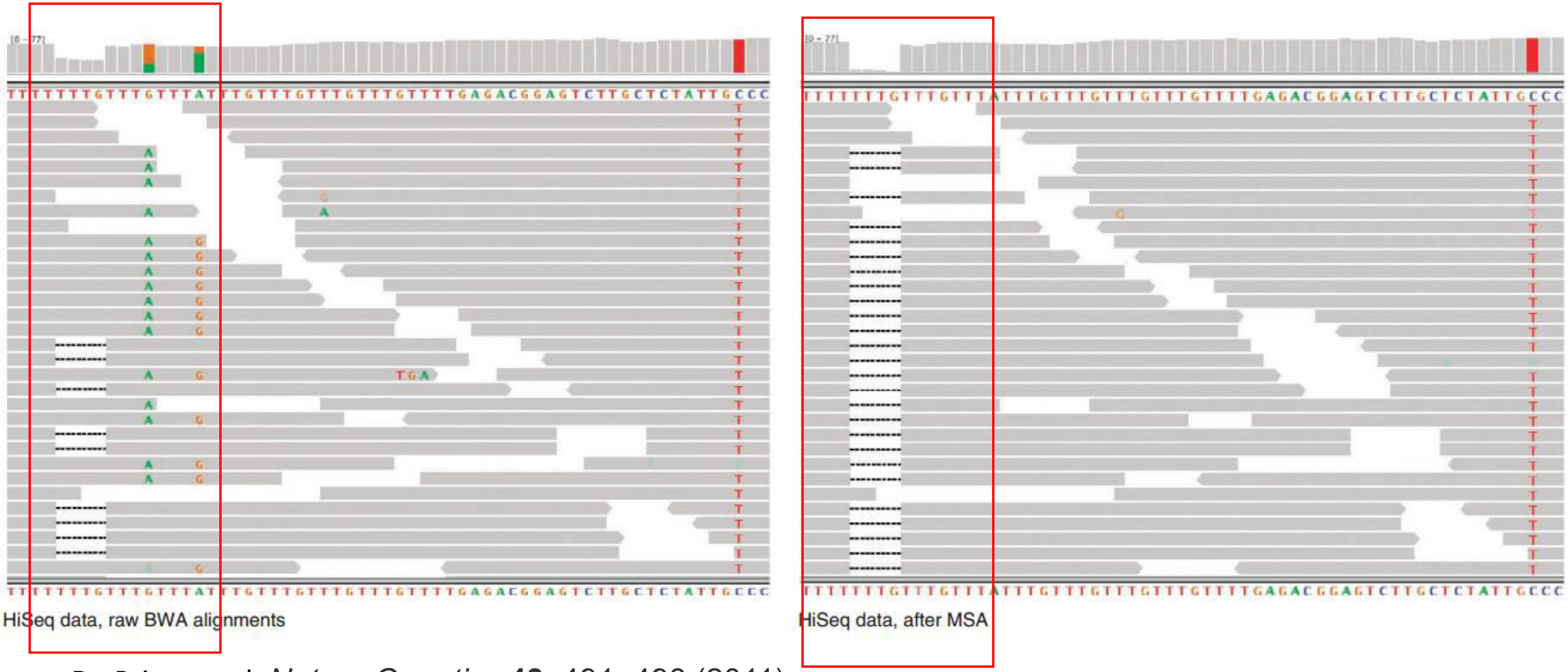


*Genome Analysis Toolkit, DePristo 2011 (Broad Institute)

Post-mapping: deduplication


[0 - 158]
C G I C
C G I C
C G I C
C G I C
C G I C
before
[0 - 80]
C G I C
after
T T A C T C T C T C T G A G A A C A G C A A A A C C A T T T T A A A T T G G C A A T C C A G T A A A A A C A T A T C T T A A A C G A A C T G T C T C T T T
3:98,531,134
193M of 257M

Post-mapping: indel realignment*

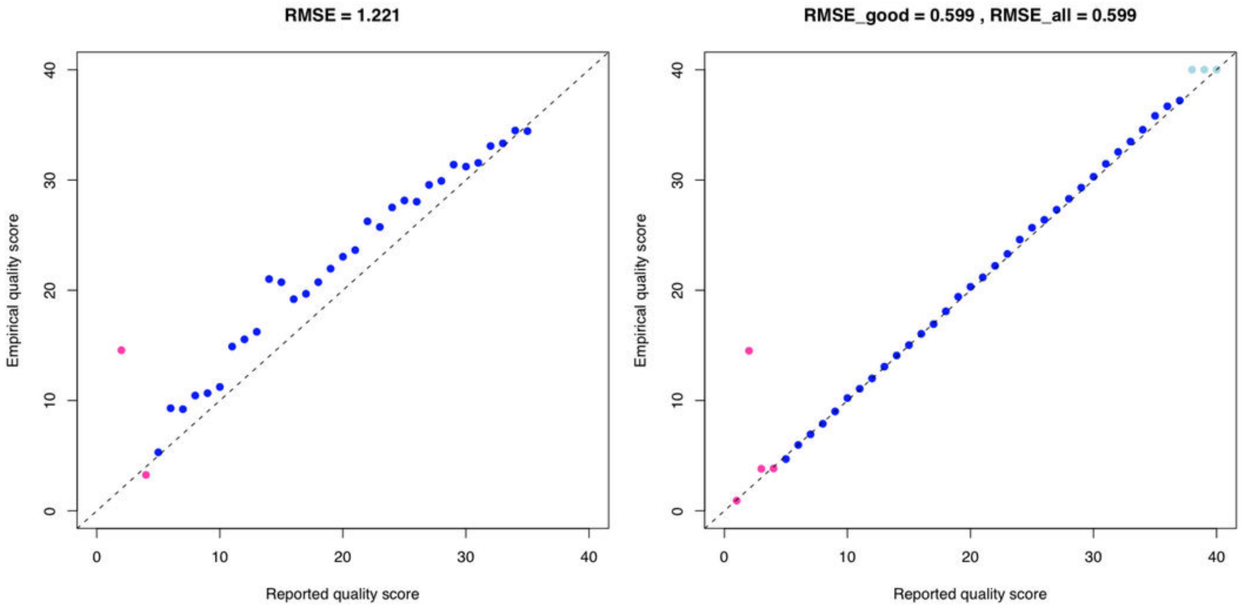


De Pristo et al. *Nature Genetics* **43**, 491–498 (2011)

*No longer recommended by GATK due to variant callers' local reassembly, but may still be useful

Post-mapping:
Base quality score recalibration

Reported Quality vs. Empirical Quality



Original Data

After GATK Recalibration

http://gatkforums.broadinstitute.org/discussion/44/base-quality-score-recalibration-bqsr

Variant calling

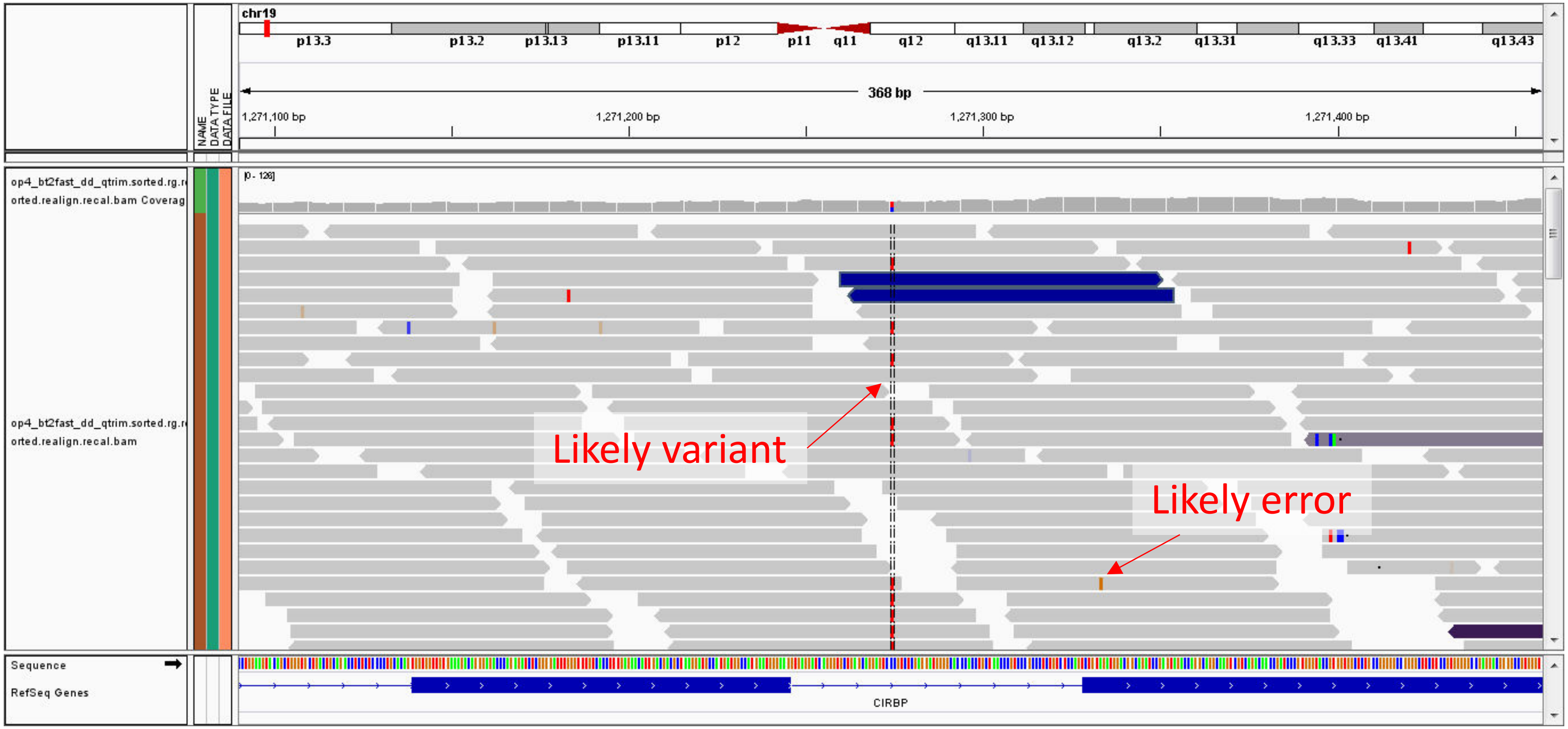
Variant calling

- Where does our individual differ from reference?
    - germline (inborn) variants
- Where does the cancer genome differ from normal tissue?
    - somatic variants

Computational problem: identify true variants

how do we tell a true variant from noise?

Differentiating variants from errors



Difficulties when

- low coverage
- problematic mapping
  - indels
- can‘t assume 50%-50% split between the variants
  - e.g., in cancer

Variant calling

- Identify differences from reference
- **Select true variants:**
- Probabilistic model/heuristic filters:
  - How many/what fraction of reads support this variant?
  - How confident was the sequencer of these reads?
  - Are there systematic biases to the variants?
    - indicative of sequencing artifacts
- How do we know what true variants look like?
  - a priori
  - **learn** the model from high-quality data
      - GATK Variant Quality Score Recalibration
    - Common variants are likely found in the particular sample too

Variant calling (continued)

- Call variants in multiple samples
  - Can “rescue” poor-coverage variant in individual samples
- “Reassemble” reads in interesting region
  - Avoid mapping problems

- Some tools:
  - GATK: Genome Analysis Toolkit (DePristo et al. 2011)
  - samtools (Li 2011)
  - FreeBayes (Garrison and Marth)

Variant calling in cancer

- Normal tissue vs Tumor tissue
- Find variants in tumor that were not in normal
  - Somatic mutations
- Cannot assume 50-50% ratio of variants

- Example tool:
  MuTect(2) (Cibulskis et al 2013)

Variant evaluation

Does a variant cause disease?

- 0.1% variation in European genomes
  - Which of these $3x10^5$ variants are damaging?
- How would we know?
  - Gold standard: experimental/clinical
    - expensive
    - done for relatively few variants, mostly in important genes
  - Computational predictions:
    - some “easy” cases based on annotation
    - population
    - evolution
    - structure
    - …

Mutations in DNA

- Point mutations (single nucleotide change)
  - Synonymous: no AA change
    - e.g., TCA -> TCC (Ser -> Ser)

LIKELY BENIGN

  - Nonsynonymous: AA change
    - e.g., TCA->CCA (Ser->Pro)
  - *Nonsense*: introduce new STOP (*) codon

    => premature termination of translation
    - e.g., TCA->TAA (Ser->*)

LIKELY DAMAGING

- Short insertions or deletions (indels)

  e.g. TCACCATCG -> TCACATCG
  - *in-frame*: multiple of 3, preserves reading frame
  - *frameshift:* not multiple of 3, disrupts reading frame

Mutations in DNA

- Point mutations (single nucleotide change)
  - Synonymous: no AA change
    - e.g., TCA -> TCC (Ser -> Ser)
  - Nonsynonymous: AA change
    - e.g., TCA->CCA (Ser->Pro)
  - *Nonsense*: introduce new STOP (*) codon

    => premature termination of translation
    - e.g., TCA->TAA (Ser->*)
- Short insertions or deletions (indels)

  e.g. TCACCATCG -> TCACATCG
  - *in-frame*: multiple of 3, preserves reading frame
  - *frameshift:* not multiple of 3, disrupts reading frame

WHAT ABOUT THESE?

Evaluating variants

- Population information
    - 1000s of individuals have been sequenced
    - Common alleles are unlikely to be very damaging
        - Otherwise they would have been eliminated by evolution
- Evolution
    - Observed in other species: likely OK
        - But: compensation by variants elsewhere
    - Never seen in evolution: probably for a reason
- Structure
    - Where is the variant in the protein?
        - core, binding site: more likely damaging
        - unstructured regions: less likely damaging
        - …
    - How similar is it to the reference variant?
        - similar/dissimilar amino acids

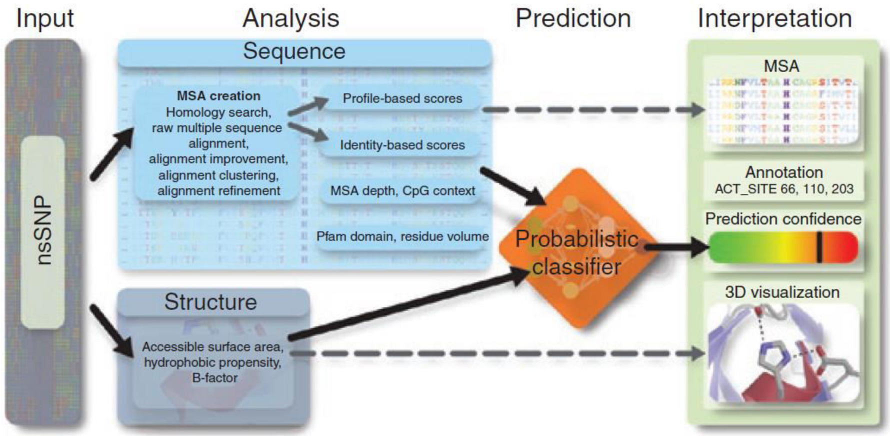
Evaluating variants

- How do we know which ones are important?
  - **Learn** from data and combine!
- Training sets:
  - Positive: known damaging variants
    - Experimentally verified
    - Simulated
  - Negative: likely benign
    - Common in population
    - Found in closely related species
      - Caveat: compensatory variants elsewhere

Example: PolyPhen2

Adzhubei et al. 2010

- Single-nucleotide variant effect prediction
- Naïve Bayes
- Features based on
  - Evolutionary conservation
  - Structure



Input

Analysis

Prediction

Interpretation

nsSNP

Sequence

MSA creation
Homology search,
raw multiple sequence alignment,
alignment improvement,
alignment clustering,
alignment refinement

Profile-based scores

Identity-based scores

MSA depth, CpG context

Pfam domain, residue volume

Structure

Accessible surface area,
hydrophobic propensity,
B-factor

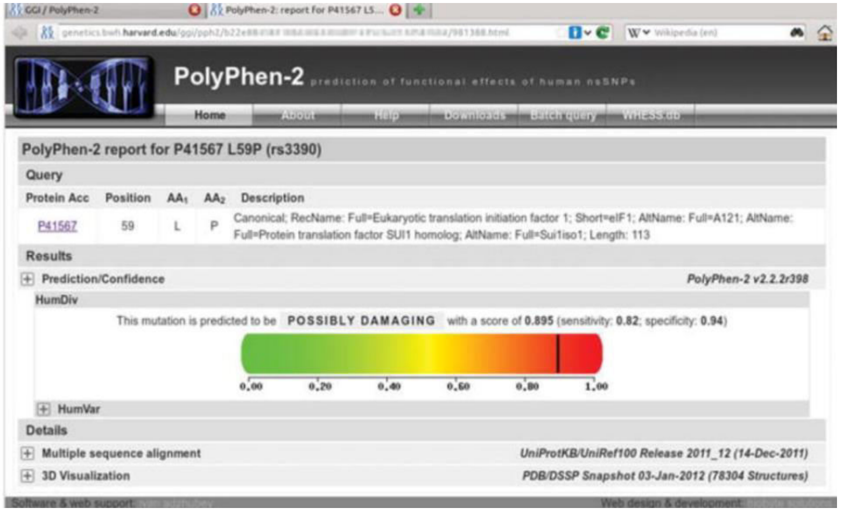
Probabilistic classifier

MSA

Annotation
ACT_SITE 66, 110, 203

Prediction confidence

3D visualization



GGI / PolyPhen-2
PolyPhen-2: report for P41567 L5...
genetics.bwh.harvard.edu/ggi/pph2/b22e.../981368.html
Wikipedia (en)
PolyPhen-2 prediction of functional effects of human nsSNPs
Home
About
Help
Downloads
Batch query
WHESS.db

PolyPhen-2 report for P41567 L59P (rs3390)

Query

Protein Acc	Position	AA_1	AA_2	Description
P41567	59	L	P	Canonical; RecName: Full=Eukaryotic translation initiation factor 1; Short=eIF1; AltName: Full=A121; AltName: Full=Protein translation factor SUI1 homolog; AltName: Full=Sui1iso1; Length: 113

Results

⊞ Prediction/Confidence *PolyPhen-2 v2.2.2r398*

HumDiv

This mutation is predicted to be **POSSIBLY DAMAGING** with a score of **0.895** (sensitivity: **0.82**; specificity: **0.94**)

⊞ HumVar

Details

⊞ Multiple sequence alignment *UniProtKB/UniRef100 Release 2011_12 (14-Dec-2011)*

⊞ 3D Visualization *PDB/DSSP Snapshot 03-Jan-2012 (78304 Structures)*

Software & web support:

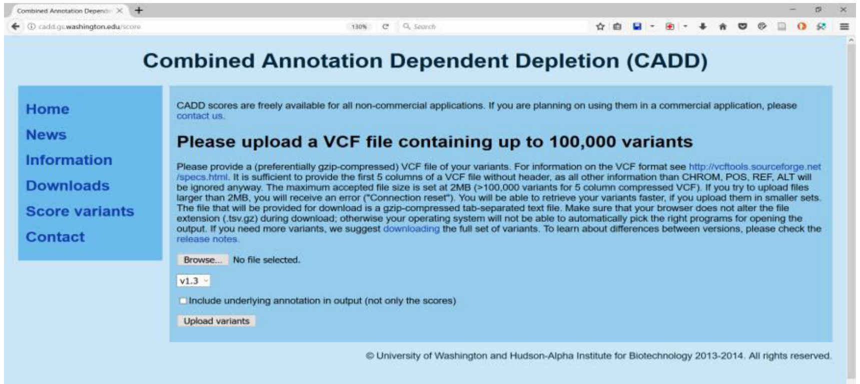
Web design & development:

PolyPhen2: training sets

- HumDiv:
  - Damaging: 3,155 alleles causing human Mendelian diseases and affecting protein stability or function
  - Neutral: 6,321 differences between human proteins and their closely related mammalian homologs
- HumVar:
  - Damaging: 13,032 human disease-causing mutations
  - Neutral: 8,946 nonsynonymous variants without annotated involvement in disease

Example: CADD

Kircher et al. 2014



Combined Annotation Dependent Depletion (CADD)

CADD scores are freely available for all non-commercial applications. If you are planning on using them in a commercial application, please contact us.

Please upload a VCF file containing up to 100,000 variants

Please provide a (preferentially gzip-compressed) VCF file of your variants. For information on the VCF format see http://vcftools.sourceforge.net/specs.html. It is sufficient to provide the first 5 columns of a VCF file without header, as all other information than CHROM, POS, REF, ALT will be ignored anyway. The maximum accepted file size is set at 2MB (>100,000 variants for 5 column compressed VCF). If you try to upload files larger than 2MB, you will receive an error ("Connection reset"). You will be able to retrieve your variants faster, if you upload them in smaller sets. The file that will be provided for download is a gzip-compressed tab-separated text file. Make sure that your browser does not alter the file extension (.tsv.gz) during download; otherwise your operating system will not be able to automatically pick the right programs for opening the output. If you need more variants, we suggest downloading the full set of variants. To learn about differences between versions, please check the release notes.

No file selected.

□ Include underlying annotation in output (not only the scores)

© University of Washington and Hudson-Alpha Institute for Biotechnology 2013-2014. All rights reserved.

- single nucleotide variants, insertions, deletions
- SVM on 63 features:
  - conservation scores (various tools)
  - prediction scores (PolyPhen2 and others)
  - sequence features
  - post-translational modifications
  - “a limited number of interaction terms”

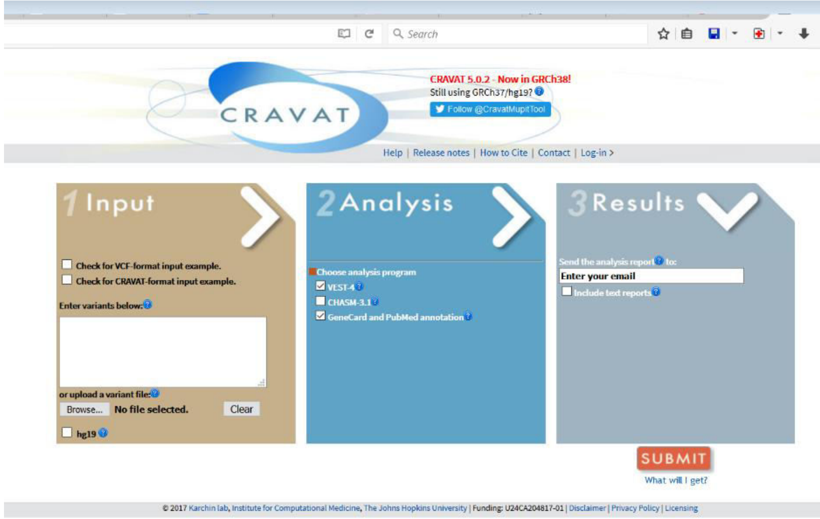
CADD: training sets

- CADD predicts *deleteriousness*

  ~ bad from evolutionary perspective
- Deleterious: simulated
- Non-deleterious: fixed in human population

Example: VEST-indel

Douville and Masica et al 2016



CRAVAT 5.0.2 - Now in GRCh38!
Still using GRCh37/hg19?

□ Check for VCF-format input example.
□ Check for CRAVAT-format input example.

Enter variants below:

or upload a variant file:
No file selected.

□ hg19

Choose analysis program
☑ VEST-4
□ CHASM-3.1
☑ GeneCard and PubMed annotation

Send the analysis report to:
Enter your email
□ Include text reports

© 2017 Karchin lab, Institute for Computational Medicine, The Johns Hopkins University | Funding: U24CA204817-01 | Disclaimer | Privacy Policy | Licensing

- Predict *pathogenicity* of indels
  - Let’s focus on in-frame
- Random forest
- Start with 49 candidate features
- Greedily select 23 features for classification
  - gene importance (according to literature) ← top feature
  - evolutionary and population features
  - structural features
  - sequence context

VEST-indel: training/testing sets

Training:

- Pathogenic: annotations from database
- Benign: occurs in >= 1% of population AND occurs in people with African ancestry

Testing:

- Pathogenic: annotations from different database
- Benign: found in other mammals

Meta-classifiers

VEST-indel paper:

- All Boolean combination of VEST-indel and three other tools
- find best-performing combination

Conclusions

- Advanced and developing technology
- Advanced and developing computational techniques
    - Room for algorithmic improvements, machine learning