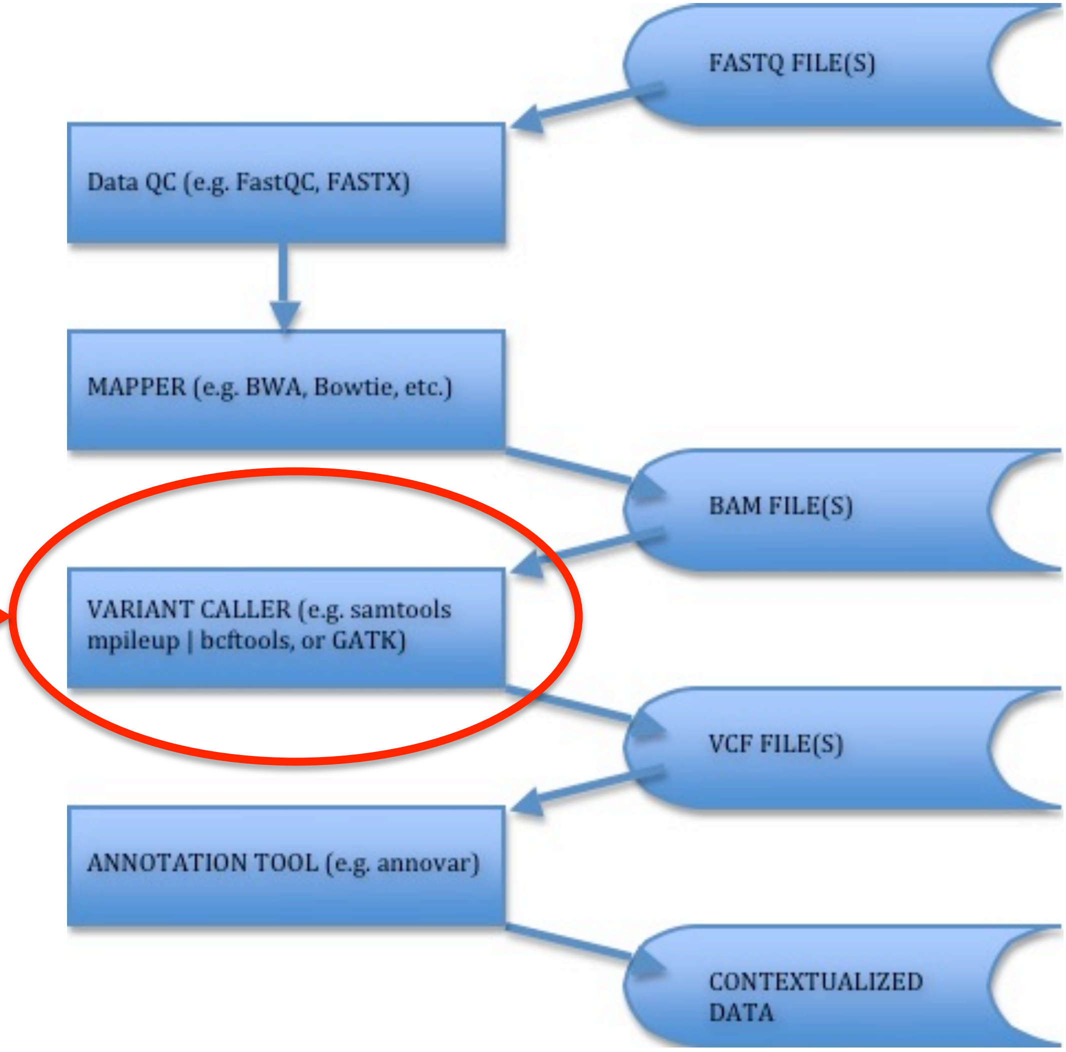
Advanced Variant Calling

You are here!



FASTQ FILE(S)

Data QC (e.g. FastQC, FASTX)

MAPPER (e.g. BWA, Bowtie, etc.)

BAM FILE(S)

VARIANT CALLER (e.g. samtools mpileup | bcftools, or GATK)

VCF FILE(S)

ANNOTATION TOOL (e.g. annovar)

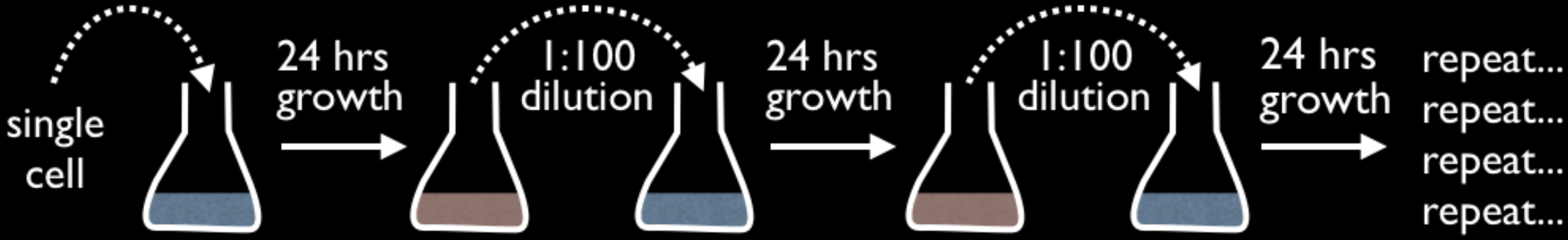
CONTEXTUALIZED DATA

Types of Genome Sequence Variants

1. Single Nucleotide Variants (SNVs) *
   – Single base changes, e.g., A→T.
2. Insertions-Deletions (Indels; DIPs) * ▶
   – Consisting of one or a few bases, e.g., +ATGA, ΔT.
3. Structural Variants (SVs) ▶
   – Everything else: large deletions, insertions, duplications, inversions, translocations, mobile element insertions, horizontal gene transfer

Different sequencing information and different algorithms are used to predict each kind of variant.

Long-term *E. coli* evolution experiment



- 12 independent populations evolved >20 yrs. Frozen "fossil record" has been archived.
- How many and what mutations?
- Compare rates of genomic change and fitness increase, monitor diversity in the population, understand molecular basis of adaptation.

Knowing what you don't know

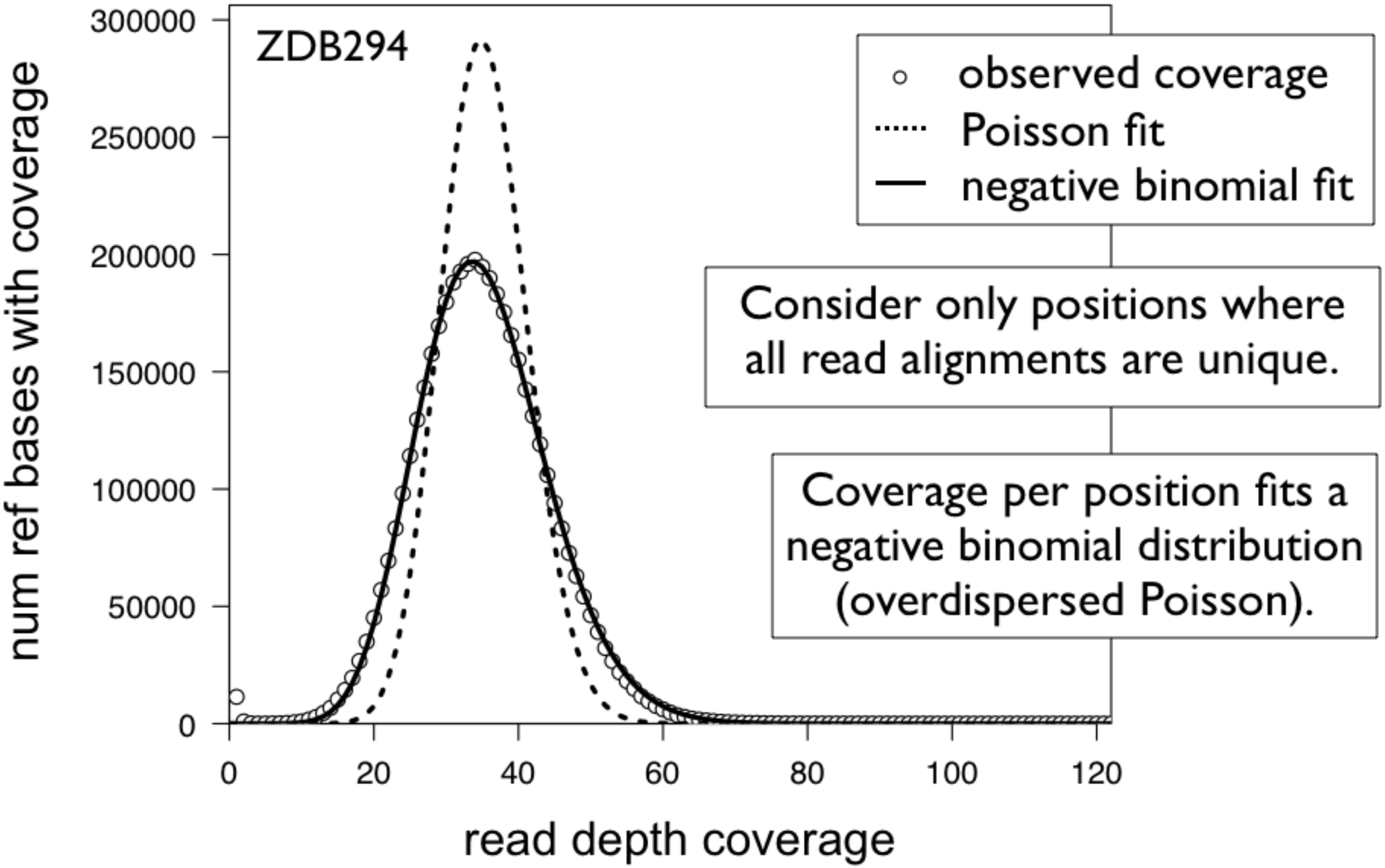
1. Theoretical limits: Read length and pair distance.
2. Practical limits: Base quality and coverage evenness.

	single-end	paired-end	mate-paired
IS insertions	*	*	*
duplications	*	*	*
inversions across IS	–	–	*
SNPs in repeats	–	–	*
insertion of new seq	–	–	–

IS = bacterial mobile elements 0.8-1.5 kb in length.

Need standardized metrics to describe completeness of re-sequencing data on a per-base per-genome basis.

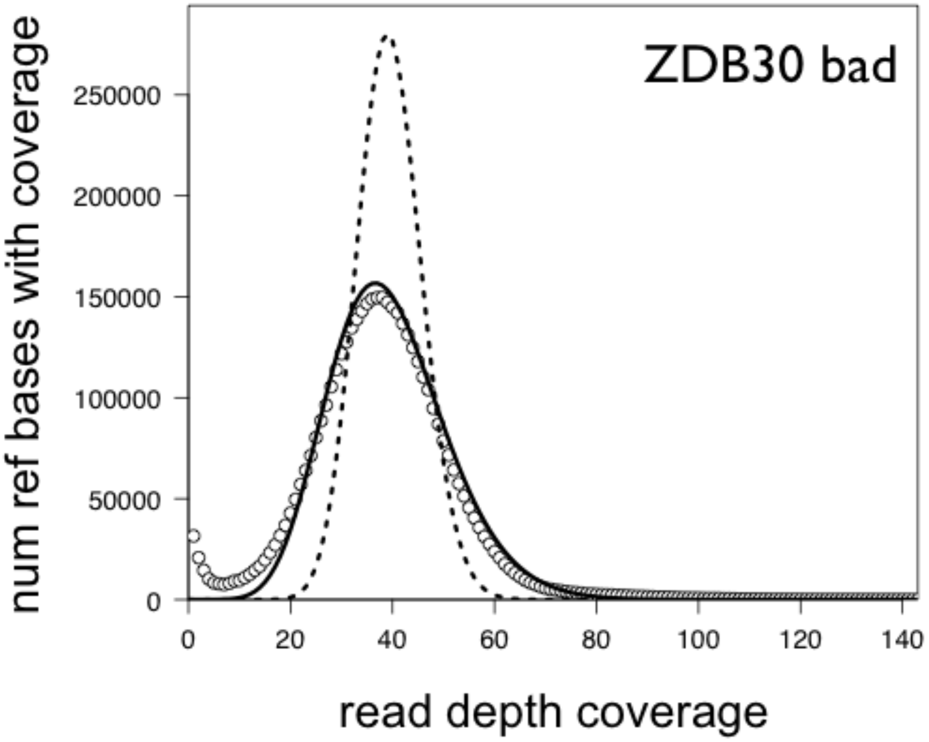
Typical Coverage Distribution

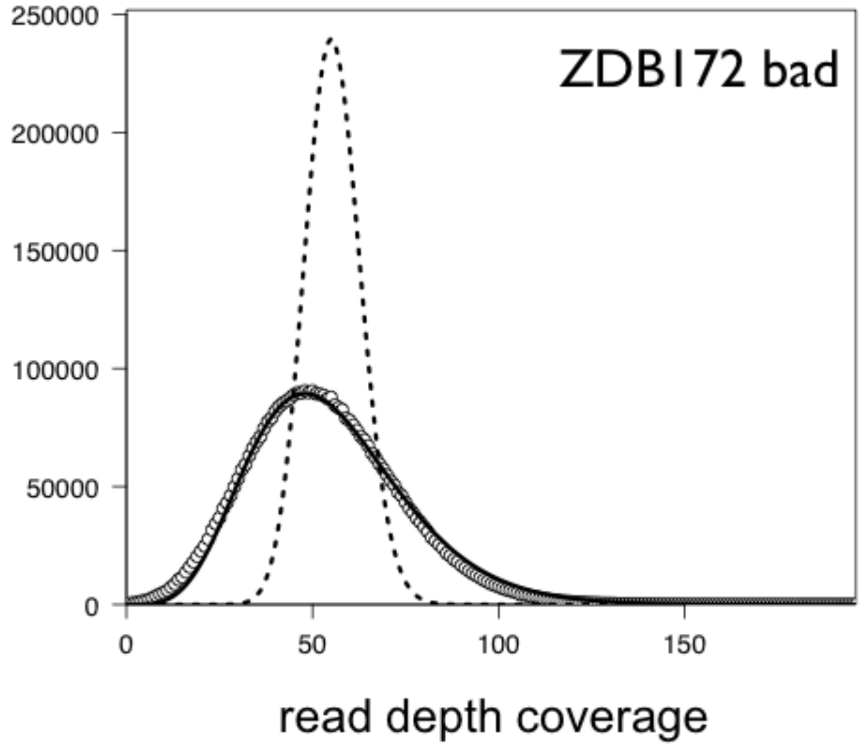


Consider only positions where all read alignments are unique.

Coverage per position fits a negative binomial distribution (overdispersed Poisson).

Problem Coverage Distributions





- Contamination with another sample?

- Large variance, missing coverage.

Both apparently from problems with library prep.

Identifying within-alignment indels

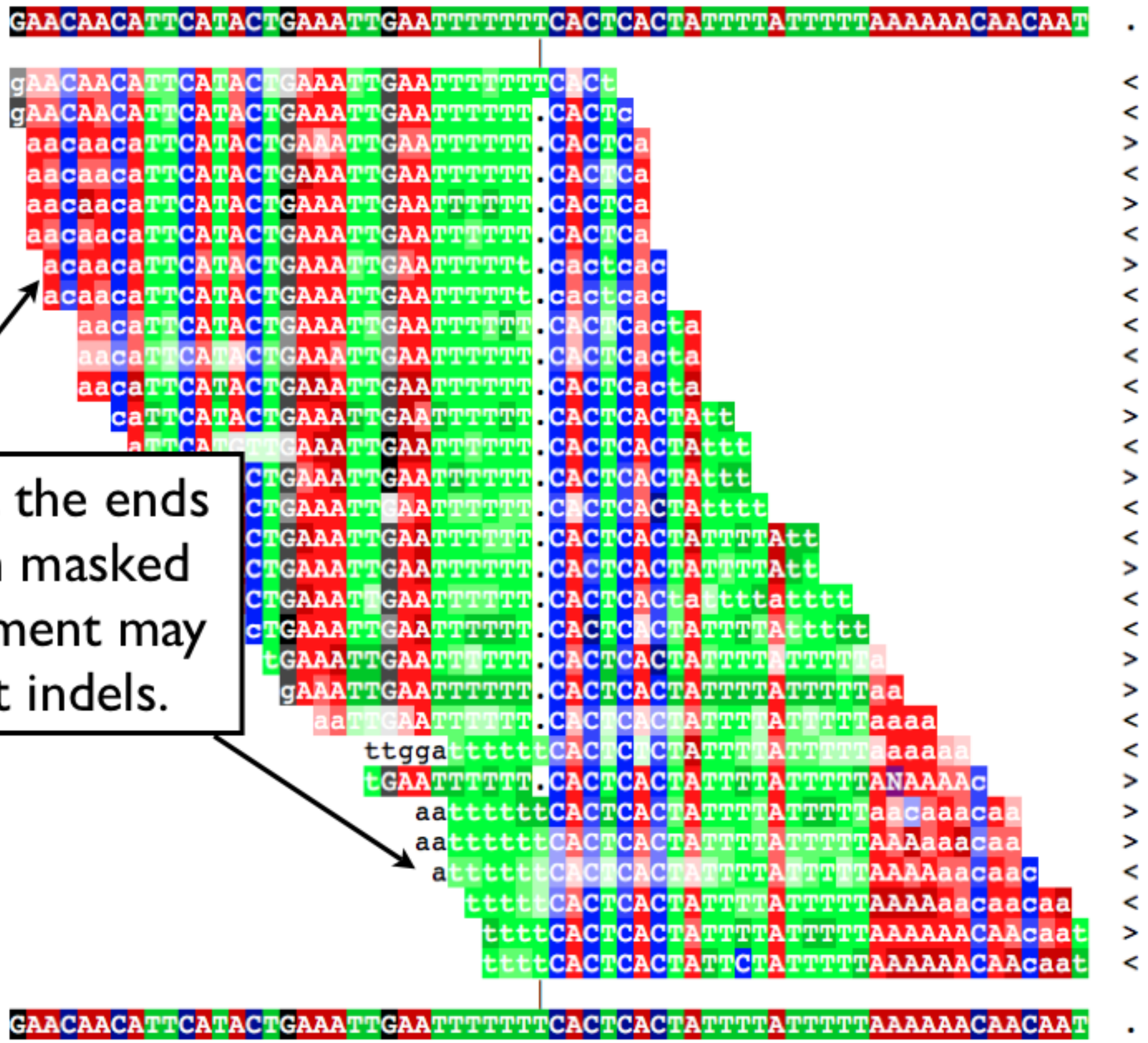
- Need to be careful in repetitive sequences and at the edges of short reads...

```
TATATTAATGCGCGCGCTAGGCTAGCT
TATATTAAT--GCGCGCTAGGCTAGCT <
TATATTAATGCGCGC--TAGGCTAGCT >
TATATTAATGCGCGC............ >
...........GCGCGCTAGGCTAGCT <
```

...where reads aligned from different directions can be ambiguously aligned.

...where reads from different directions that end in a simple sequence repeat may hide indels.

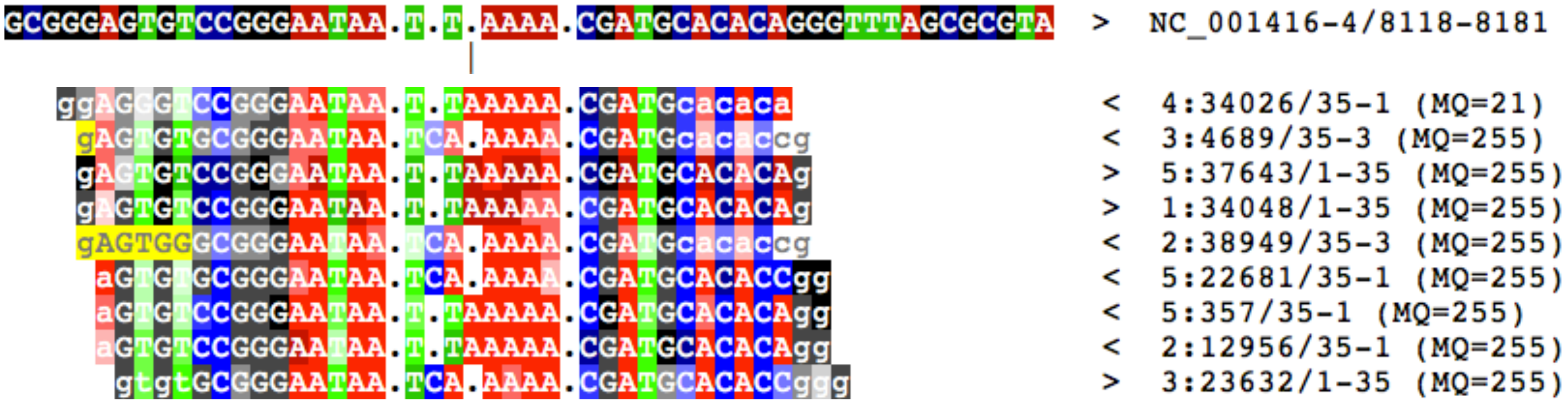
Example of a ***breseq*** prediction



```
GAACAACATTCATACTGAAATTGAATTTTTTTCACTCACTATTTTATTTTTAAAAAACAACAAT    .

gAACAACATTCATACTGAAATTGAATTTTTTTCACt                                <
gAACAACATTCATACTGAAATTGAATTTTTT.CACTc                               <
 aacaacaTTCATACTGAAATTGAATTTTTT.CACTCa                              >
 aacaacaTTCATACTGAAATTGAATTTTTT.CACTCa                              <
 aacaacaTTCATACTGAAATTGAATTTTTT.CACTCa                              >
 aacaacaTTCATACTGAAATTGAATTTTTT.CACTCa                              <
  acaacaTTCATACTGAAATTGAATTTTTt.cactcac                             >
  acaacaTTCATACTGAAATTGAATTTTTt.cactcac                             <
    aacaTTCATACTGAAATTGAATTTTTT.CACTCacta                           <
    aacaTTCATACTGAAATTGAATTTTTT.CACTCacta                           <
    aacaTTCATACTGAAATTGAATTTTTT.CACTCacta                           <
      caTTCATACTGAAATTGAATTTTTT.CACTCACTAtt                         >
       aTTCATGTTGAAATTGAATTTTTT.CACTCACTAttt                        <
              CTGAAATTGAATTTTTT.CACTCACTAttt                        >
              CTGAAATTGAATTTTTT.CACTCACTAtttt                       <
              CTGAAATTGAATTTTTT.CACTCACTATTTTAtt                    <
              CTGAAATTGAATTTTTT.CACTCACTATTTTAtt                    >
              CTGAAATTGAATTTTTT.CACTCACtattttatttt                  <
              cTGAAATTGAATTTTTT.CACTCACTATTTTAttttt                 <
               tGAAATTGAATTTTTT.CACTCACTATTTTATTTTTa                >
                gAAATTGAATTTTTT.CACTCACTATTTTATTTTTaa               >
                  aaTTGAATTTTTT.CACTCACTATTTTATTTTTaaaa             <
                     ttggattttttCACTCTCTATTTTATTTTTaaaaaa           <
                     tGAATTTTTT.CACTCACTATTTTATTTTTANAAAAc          >
                        aattttttCACTCACTATTTTATTTTTaacaaacaa        >
                        aattttttCACTCACTATTTTATTTTTAAAaaacaa        >
                         attttttCACTCACTATTTTATTTTTAAAAaacaac       <
                           tttttCACTCACTATTTTATTTTTAAAAaacaacaa     <
                            ttttCACTCACTATTTTATTTTTAAAAAACAAcaat    >
                            ttttCACTCACTATTCTATTTTTAAAAAACAAcaat    <

GAACAACATTCATACTGAAATTGAATTTTTTTCACTCACTATTTTATTTTTAAAAAACAACAAT    .
```

Lowercase bases at the ends of reads have been masked because their alignment may be ambiguous wrt indels.

Pitfalls of the column mindset

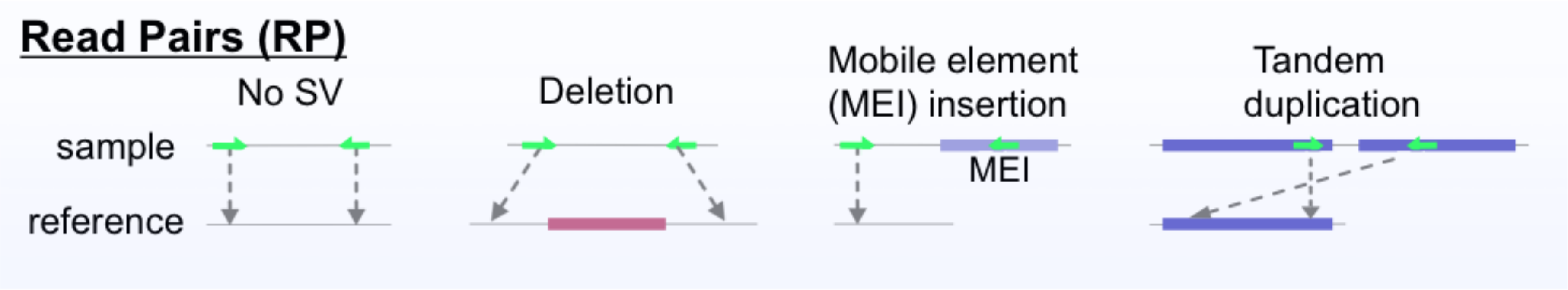


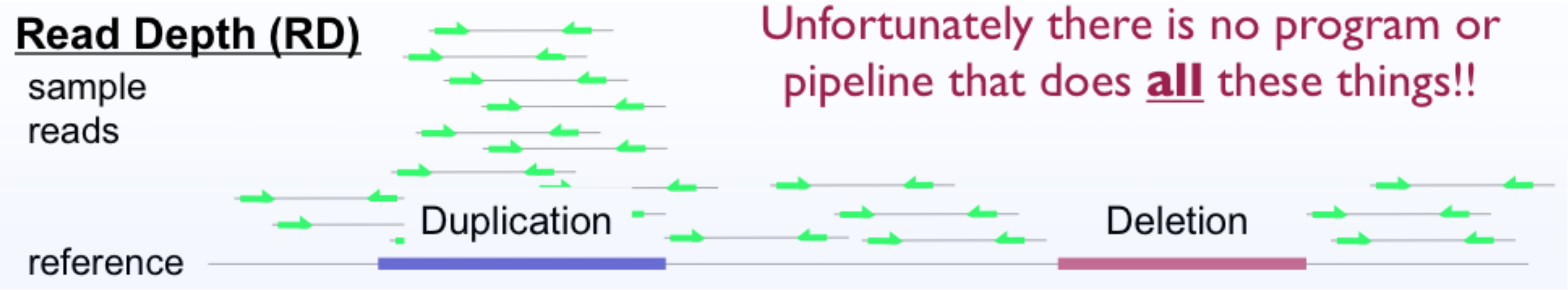
```
GCGGGAGTGTCCGGGAATAA.T.T.AAAA.CGATGCACACAGGGTTTAGCGCGTA  >  NC_001416-4/8118-8181
                        |
   ggAGGGTCCGGGAATAA.T.TAAAAA.CGATGcacaca                 <  4:34026/35-1 (MQ=21)
    gAGTGTGCGGGAATAA.TCA.AAAA.CGATGcacaccg                <  3:4689/35-3 (MQ=255)
    gAGTGTCCGGGAATAA.T.TAAAAA.CGATGCACACAg                >  5:37643/1-35 (MQ=255)
    gAGTGTCCGGGAATAA.T.TAAAAA.CGATGCACACAg                >  1:34048/1-35 (MQ=255)
    gAGTGGGCGGGAATAA.TCA.AAAA.CGATGcacaccg                <  2:38949/35-3 (MQ=255)
     aGTGTGCGGGAATAA.TCA.AAAA.CGATGCACACCgg               <  5:22681/35-1 (MQ=255)
     aGTGTCCGGGAATAA.T.TAAAAA.CGATGCACACAgg               <  5:357/35-1 (MQ=255)
     aGTGTCCGGGAATAA.T.TAAAAA.CGATGCACACAgg               <  2:12956/35-1 (MQ=255)
      gtgtGCGGGAATAA.TCA.AAAA.CGATGCACACCggg              >  3:23632/1-35 (MQ=255)
```

Requires local multiple sequence re-alignment to get it right!

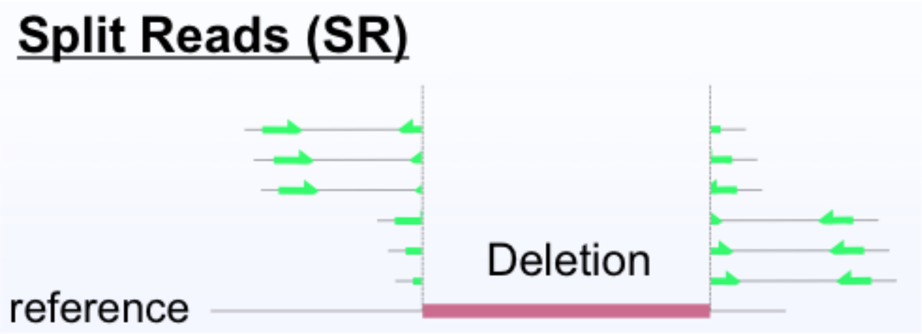
Implemented in samtools mpileup and the Genome Alignment Toolkit (GATK).

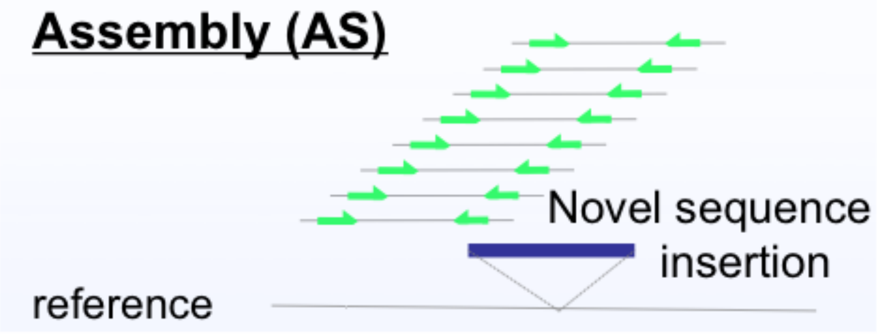
Predicting structural variants





Unfortunately there is no program or pipeline that does **all** these things!!





(http://www.genome.gov/Pages/Research/DER/1000GenomesProjectTutorials/StructuralVariants-JanKorbel.pdf)

Identifying large deletions

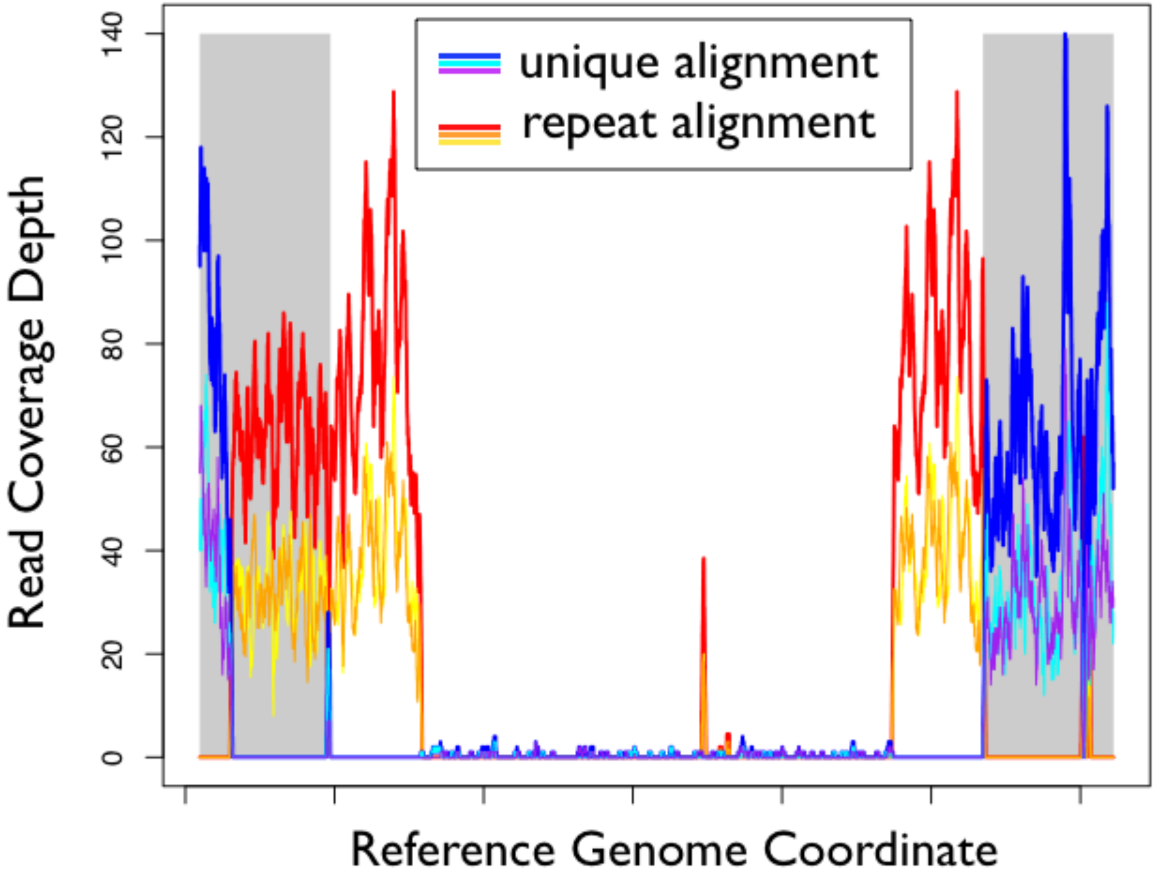
1. Seed deletions at positions with zero coverage.

2. Propagate boundaries outward until reaching a read-depth threshold based on the overall distribution.

3. Propagate through repeat regions, where a read aligns to multiple places in the genome.

Example of a ***breseq*** prediction

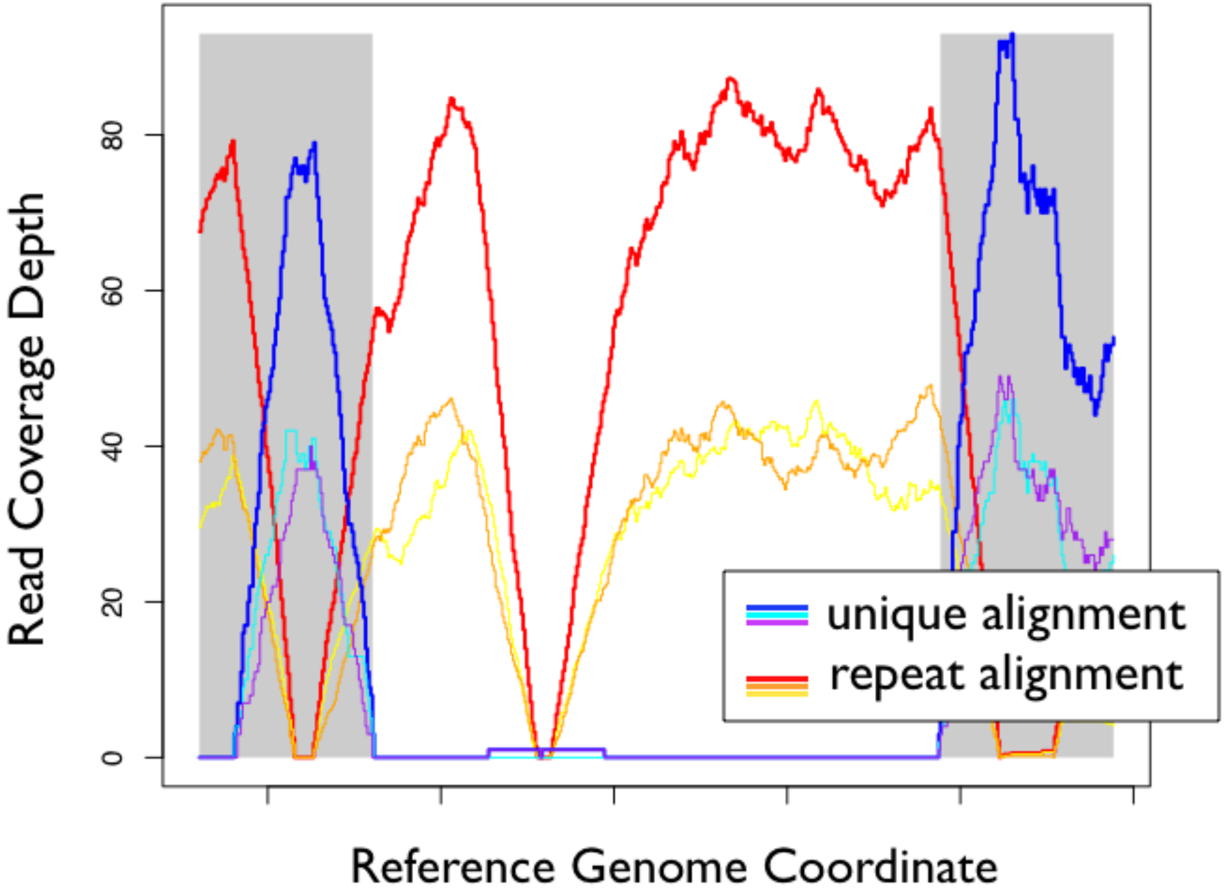
- Sometimes the molecular event is obvious...



- Recombination between nearby IS3 copies.

Example of a ***breseq*** prediction

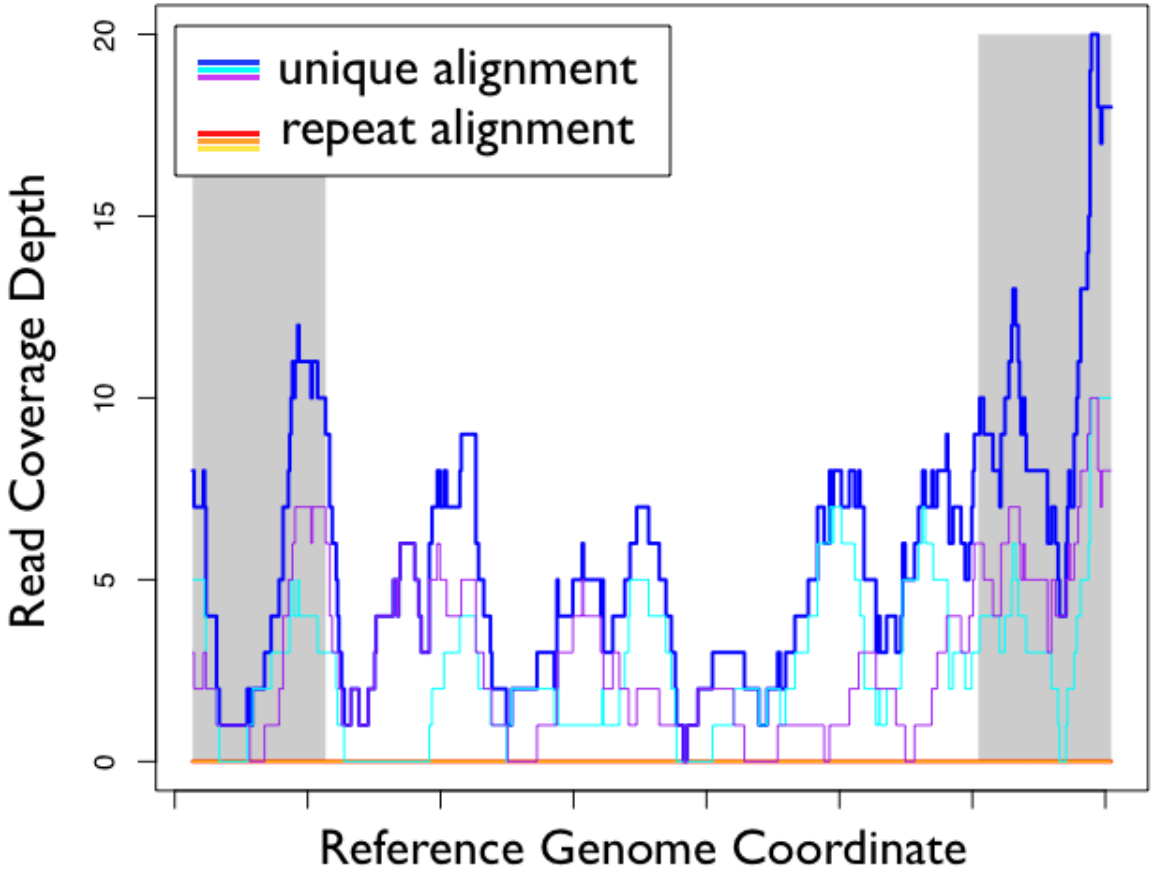
- Sometimes the mutation is not obvious...



- Gene conversion of 23S rRNA copy!!

Example of a ***breseq*** prediction

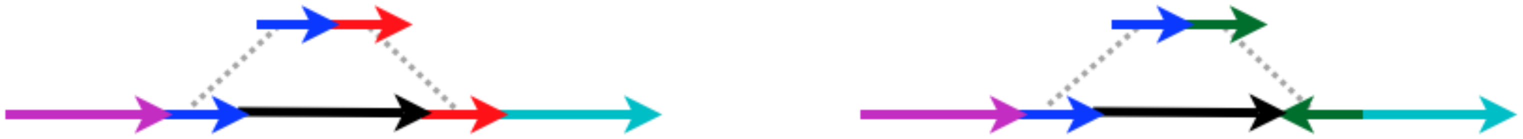
- Sometimes overall low or biased coverage leads to false predictions of deletions.



- Recognizable by sloped vs. steep edges.

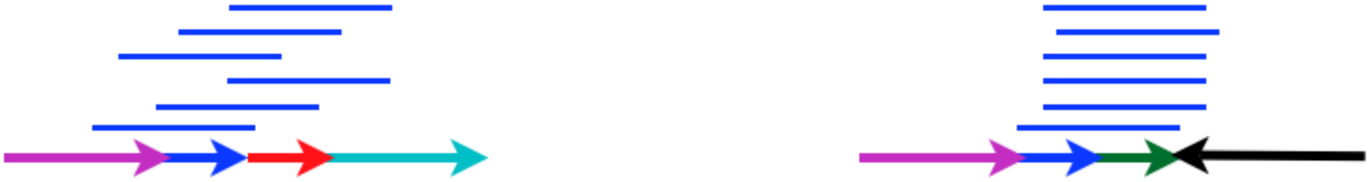
Identifying new junctions

1. Find “mosaic” reads that partially map to two locations in the genome (possibly with overlap).



2. Create consensus list of possible new junctions.

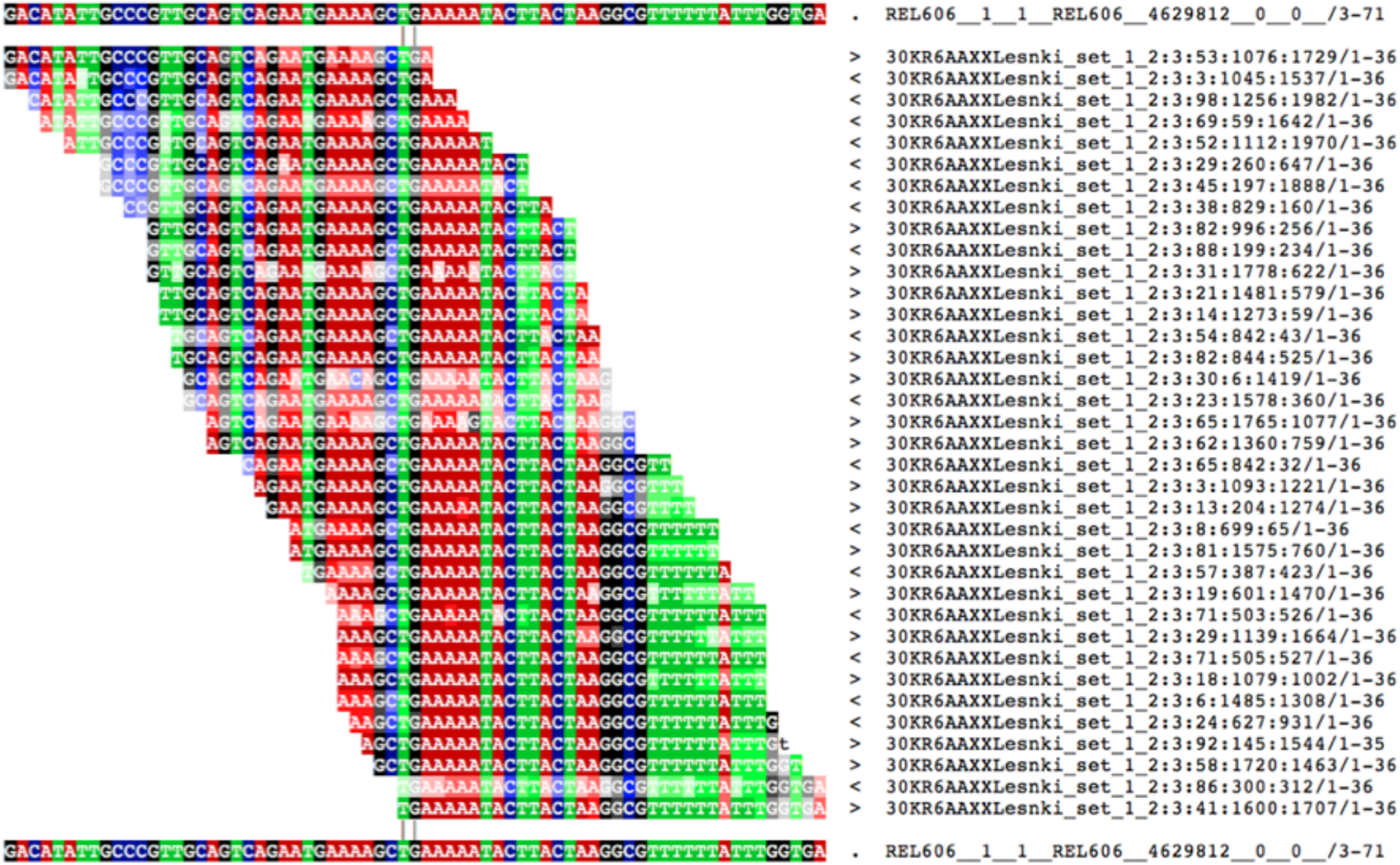
3. Re-align all reads to candidate junctions.



4. Predict a new junction if reads map better to it than to the reference across its whole length.

Example of a good junction

position	overlap	reads	gene	coords	product
1 =	0	36	*-/thrL*	/189	-/thr operon leader peptide
= 4629812			*lasT/-*	4629789/	predicted rRNA methyltransferase/-

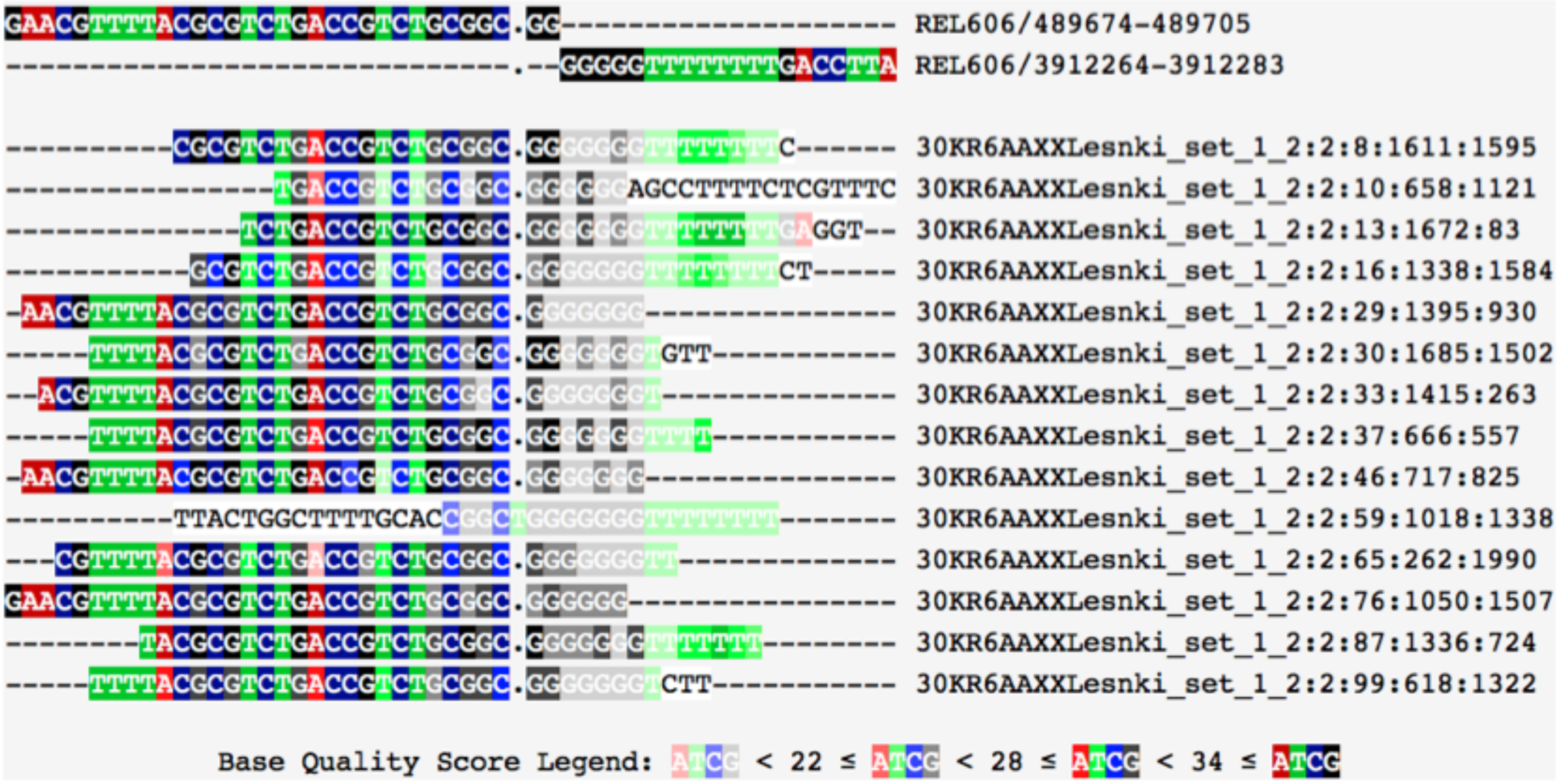


```
GACATATTGCCCGTTGCAGTCAGAATGAAAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTTGGTGA  . REL606__1__1__REL606__4629812__0__0__/3-71

GACATATTGCCCGTTGCAGTCAGAATGAAAAGCTGA                                   > 30KR6AAXXLesnki_set_1_2:3:53:1076:1729/1-36
GACATATTGCCCGTTGCAGTCAGAATGAAAAGCTGA                                   < 30KR6AAXXLesnki_set_1_2:3:3:1045:1537/1-36
  CATATTGCCCGTTGCAGTCAGAATGAAAAGCTGAAA                                 < 30KR6AAXXLesnki_set_1_2:3:98:1256:1982/1-36
   ATATTGCCCGTTGCAGTCAGAATGAAAAGCTGAAAA                                < 30KR6AAXXLesnki_set_1_2:3:69:59:1642/1-36
     ATTGCCCGTTGCAGTCAGAATGAAAAGCTGAAAAAT                              < 30KR6AAXXLesnki_set_1_2:3:52:1112:1970/1-36
        GCCCGTTGCAGTCAGAATGAAAAGCTGAAAAATACT                           < 30KR6AAXXLesnki_set_1_2:3:29:260:647/1-36
        GCCCGTTGCAGTCAGAATGAAAAGCTGAAAAATACT                           < 30KR6AAXXLesnki_set_1_2:3:45:197:1888/1-36
          CCGTTGCAGTCAGAATGAAAAGCTGAAAAATACTTA                         < 30KR6AAXXLesnki_set_1_2:3:38:829:160/1-36
            GTTGCAGTCAGAATGAAAAGCTGAAAAATACTTACT                       > 30KR6AAXXLesnki_set_1_2:3:82:996:256/1-36
            GTTGCAGTCAGAATGAAAAGCTGAAAAATACTTACT                       < 30KR6AAXXLesnki_set_1_2:3:88:199:234/1-36
            GTTGCAGTCAGAATGAAAAGCTGAAAAATACTTACT                       > 30KR6AAXXLesnki_set_1_2:3:31:1778:622/1-36
             TTGCAGTCAGAATGAAAAGCTGAAAAATACTTACTA                      > 30KR6AAXXLesnki_set_1_2:3:21:1481:579/1-36
             TTGCAGTCAGAATGAAAAGCTGAAAAATACTTACTA                      > 30KR6AAXXLesnki_set_1_2:3:14:1273:59/1-36
              TGCAGTCAGAATGAAAAGCTGAAAAATACTTACTAA                     < 30KR6AAXXLesnki_set_1_2:3:54:842:43/1-36
              TGCAGTCAGAATGAAAAGCTGAAAAATACTTACTAA                     > 30KR6AAXXLesnki_set_1_2:3:82:844:525/1-36
               GCAGTCAGAATGAACAGCTGAAAAATACTTACTAAG                    > 30KR6AAXXLesnki_set_1_2:3:30:6:1419/1-36
               GCAGTCAGAATGAAAAGCTGAAAAATACTTACTAAG                    < 30KR6AAXXLesnki_set_1_2:3:23:1578:360/1-36
                 AGTCAGAATGAAAAGCTGAAAAGTACTTACTAAGGC                  > 30KR6AAXXLesnki_set_1_2:3:65:1765:1077/1-36
                 AGTCAGAATGAAAAGCTGAAAAATACTTACTAAGGC                  > 30KR6AAXXLesnki_set_1_2:3:62:1360:759/1-36
                    CAGAATGAAAAGCTGAAAAATACTTACTAAGGCGTT               < 30KR6AAXXLesnki_set_1_2:3:65:842:32/1-36
                     AGAATGAAAAGCTGAAAAATACTTACTAAGGCGTTT              > 30KR6AAXXLesnki_set_1_2:3:3:1093:1221/1-36
                      GAATGAAAAGCTGAAAAATACTTACTAAGGCGTTTT             > 30KR6AAXXLesnki_set_1_2:3:13:204:1274/1-36
                        ATGAAAAGCTGAAAAATACTTACTAAGGCGTTTTTT           < 30KR6AAXXLesnki_set_1_2:3:8:699:65/1-36
                        ATGAAAAGCTGAAAAATACTTACTAAGGCGTTTTTT           > 30KR6AAXXLesnki_set_1_2:3:81:1575:760/1-36
                         TGAAAAGCTGAAAAATACTTACTAAGGCGTTTTTTA          < 30KR6AAXXLesnki_set_1_2:3:57:387:423/1-36
                           AAAAGCTGAAAAATACTTACTAAGGCGTTTTTTATT        > 30KR6AAXXLesnki_set_1_2:3:19:601:1470/1-36
                            AAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTT       < 30KR6AAXXLesnki_set_1_2:3:71:503:526/1-36
                            AAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTT       > 30KR6AAXXLesnki_set_1_2:3:29:1139:1664/1-36
                            AAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTT       < 30KR6AAXXLesnki_set_1_2:3:71:505:527/1-36
                            AAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTT       > 30KR6AAXXLesnki_set_1_2:3:18:1079:1002/1-36
                            AAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTT       < 30KR6AAXXLesnki_set_1_2:3:6:1485:1308/1-36
                             AAGCTGAAAAATACTTACTAAGGCGTTTTTTATTTG      < 30KR6AAXXLesnki_set_1_2:3:24:627:931/1-36
                              AGCTGAAAAATACTTACTAAGGCGTTTTTTATTTGt     > 30KR6AAXXLesnki_set_1_2:3:92:145:1544/1-35
                               GCTGAAAAATACTTACTAAGGCGTTTTTTATTTGGT    > 30KR6AAXXLesnki_set_1_2:3:58:1720:1463/1-36
                                 TGAAAAATACTTACTAAGGCGTTTTTTATTTGGTGA  < 30KR6AAXXLesnki_set_1_2:3:86:300:312/1-36
                                 TGAAAAATACTTACTAAGGCGTTTTTTATTTGGTGA  > 30KR6AAXXLesnki_set_1_2:3:41:1600:1707/1-36

GACATATTGCCCGTTGCAGTCAGAATGAAAAGCTGAAAAATACTTACTAAGGCGTTTTTTATTTGGTGA  . REL606__1__1__REL606__4629812__0__0__/3-71
```

Example of a bad junction

- Beware of reads ending in homopolymer runs!

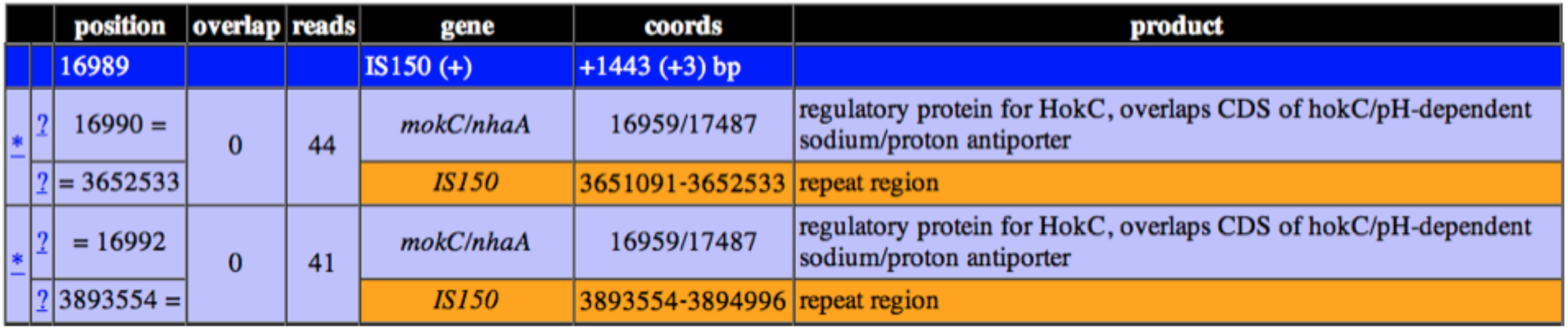
position	overlap	full / total reads	gene	coords	product
= 489705	0	7 / 14	*ybbN*	490447-489593	predicted thioredoxin domain-containing protein
3912264 =			*ilvL/ilvG*	3912221/3912359	ilvG operon leader peptide/acetolactate synthase II, valine insensitive, large subunit



```
GAACGTTTTACGCGTCTGACCGTCTGCGGC.GG-------------------- REL606/489674-489705
------------------------------.--GGGGGTTTTTTTTGACCTTA REL606/3912264-3912283

----------CGCGTCTGACCGTCTGCGGC.GGGGGGGTTTTTTTTC------ 30KR6AAXXLesnki_set_1_2:2:8:1611:1595
----------------TGACCGTCTGCGGC.GGGGGGAGCCTTTTCTCGTTTC 30KR6AAXXLesnki_set_1_2:2:10:658:1121
--------------TCTGACCGTCTGCGGC.GGGGGGGTTTTTTTTGAGGT-- 30KR6AAXXLesnki_set_1_2:2:13:1672:83
-----------GCGTCTGACCGTCTGCGGC.GGGGGGGTTTTTTTTCT----- 30KR6AAXXLesnki_set_1_2:2:16:1338:1584
-AACGTTTTACGCGTCTGACCGTCTGCGGC.GGGGGGG--------------- 30KR6AAXXLesnki_set_1_2:2:29:1395:930
-----TTTTACGCGTCTGACCGTCTGCGGC.GGGGGGGTGTT----------- 30KR6AAXXLesnki_set_1_2:2:30:1685:1502
--ACGTTTTACGCGTCTGACCGTCTGCGGC.GGGGGGGT-------------- 30KR6AAXXLesnki_set_1_2:2:33:1415:263
-----TTTTACGCGTCTGACCGTCTGCGGC.GGGGGGGTTTT----------- 30KR6AAXXLesnki_set_1_2:2:37:666:557
-AACGTTTTACGCGTCTGACCGTCTGCGGC.GGGGGGG--------------- 30KR6AAXXLesnki_set_1_2:2:46:717:825
----------TTACTGGCTTTTGCACCGGCTGGGGGGGTTTTTTTT------- 30KR6AAXXLesnki_set_1_2:2:59:1018:1338
---CGTTTTACGCGTCTGACCGTCTGCGGC.GGGGGGGTT------------- 30KR6AAXXLesnki_set_1_2:2:65:262:1990
GAACGTTTTACGCGTCTGACCGTCTGCGGC.GGGGGG---------------- 30KR6AAXXLesnki_set_1_2:2:76:1050:1507
--------TACGCGTCTGACCGTCTGCGGC.GGGGGGGTTTTTTT-------- 30KR6AAXXLesnki_set_1_2:2:87:1336:724
-----TTTTACGCGTCTGACCGTCTGCGGC.GGGGGGGTCTT----------- 30KR6AAXXLesnki_set_1_2:2:99:618:1322

        Base Quality Score Legend: ATCG < 22 ≤ ATCG < 28 ≤ ATCG < 34 ≤ ATCG
```

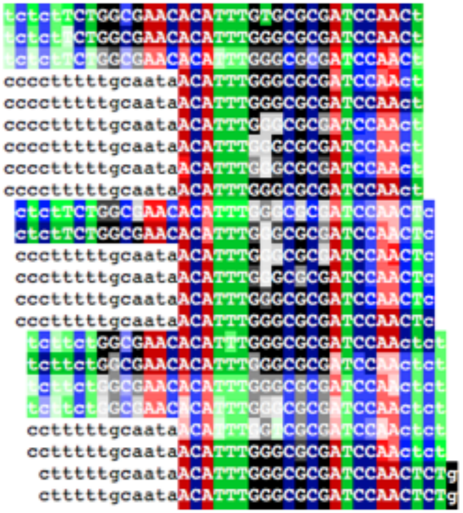
Example of a ***breseq*** prediction

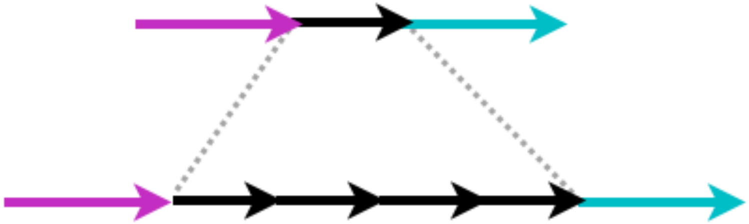
- IS insertions create two new junctions...



		position	overlap	reads	gene	coords	product
		16989			IS150 (+)	+1443 (+3) bp	
*	?	16990 =	0	44	*mokC/nhaA*	16959/17487	regulatory protein for HokC, overlaps CDS of hokC/pH-dependent sodium/proton antiporter
	?	= 3652533			*IS150*	3651091-3652533	repeat region
*	?	= 16992	0	41	*mokC/nhaA*	16959/17487	regulatory protein for HokC, overlaps CDS of hokC/pH-dependent sodium/proton antiporter
	?	3893554 =			*IS150*	3893554-3894996	repeat region

- Sometimes both new and old junctions exist...

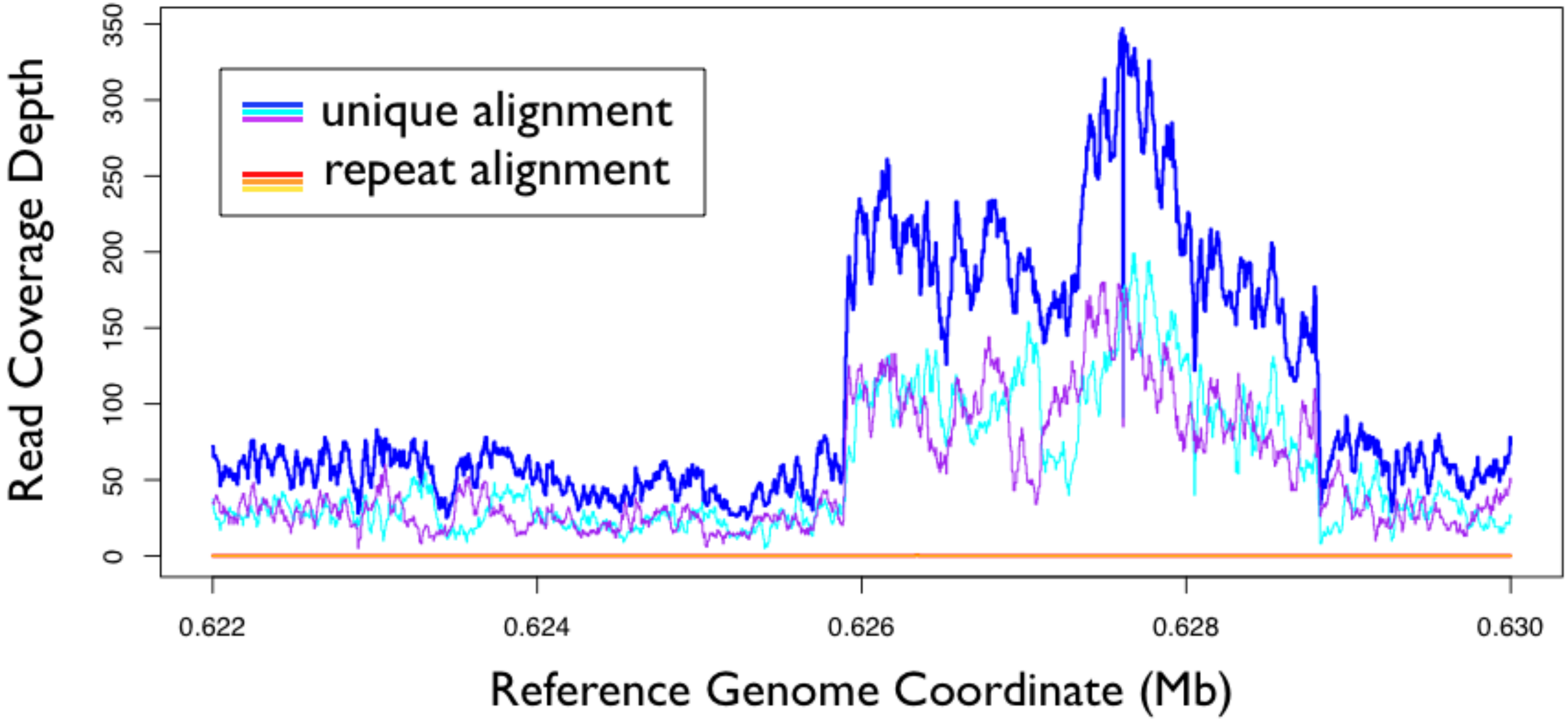




tandem head-to-tail duplications

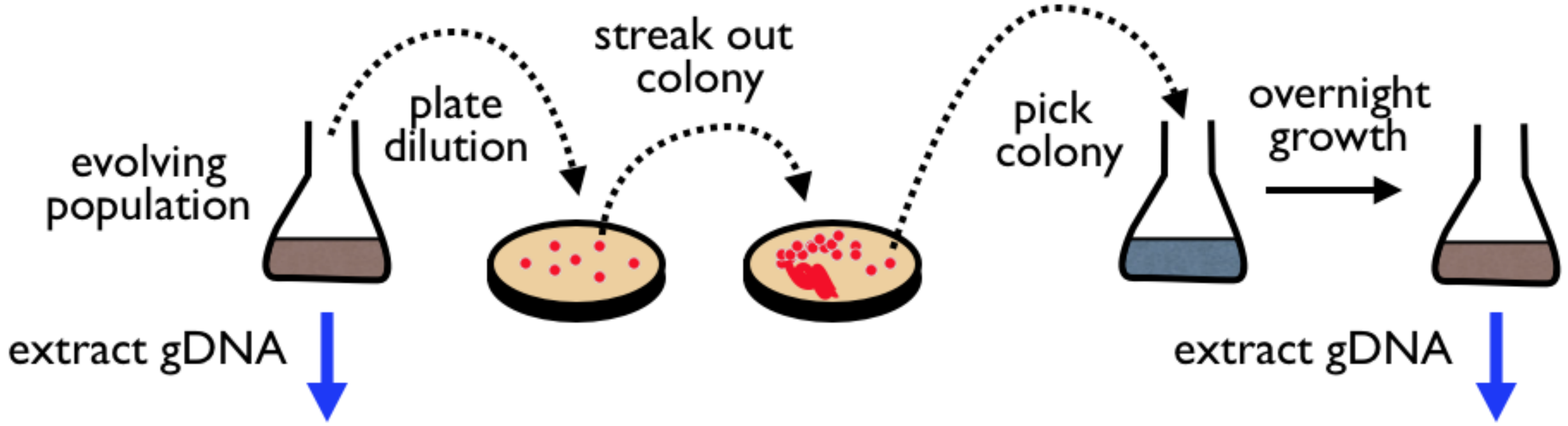
Identifying copy number variation

- Coverage is very noisy, but a fingerprint is (somewhat) consistent across runs.



- Tile into segments, train model on many genomes, look for deviation

Mixed population analysis



Every read could be from any individual.

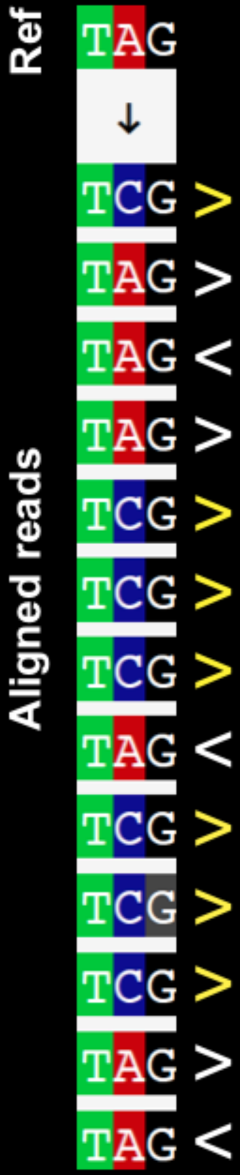
All reads are from a single clone.

Frequencies of mutations competing in population.

No linkage information.

Information about which mutations occur together.

Sequencing error or polymorphism?



- Map reads to ancestor genome.
  Only consider single-base substitutions.

- Log-likelihood test for polymorphism:

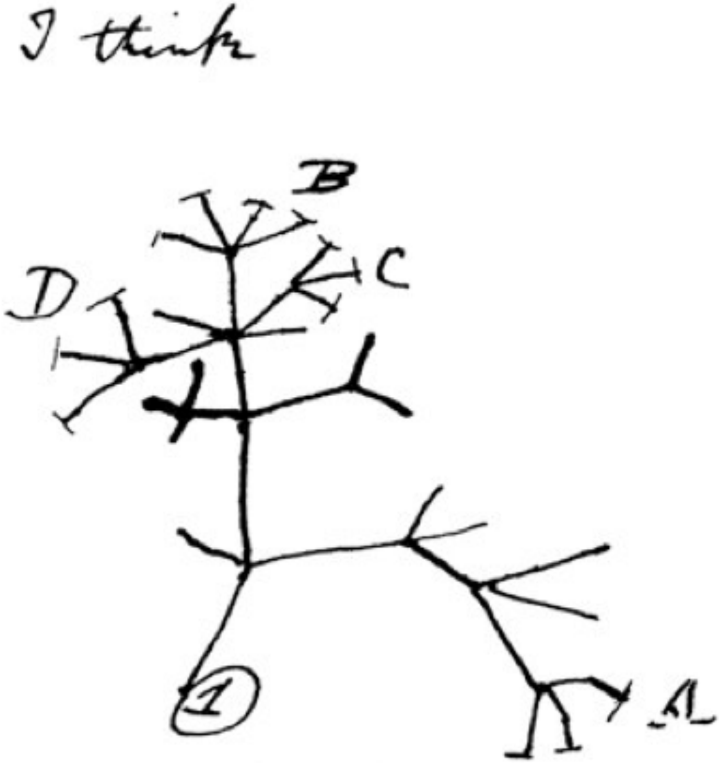
$$D = -2 \ln \frac{\Pr(\text{obs} \mid \text{no polymorphism, i.e. all error})}{\Pr(\text{obs} \mid \text{ML fraction new allele})}$$

- Clone sequence data serves as a negative control (all errors, no polymorphisms).

- Filter out predictions with other biases:
  strand bias, systematically low quality scores

From evidence to mutations

- Genome sequencing data gives us *evidence* of changes: read alignments, missing coverage, new junctions, ...
- But we really want a list of biological *mutations* to study evolutionary history and molecular mechanisms.
- Complication: Later events may sometimes hide earlier events (e.g. SNV in region that is later deleted)



Genome Diffs

- To submit a changed genome sequence to GenBank you must currently re-submit the entire genome – *even if it has only a one base difference.*

- Mutational events are essentially *genome differences*. (In a Comp Sci sense of applying "patches" to files)

- Supplementary tables are not a sustainable, standardized, or re-usable way to report this data.

- An ideal genome analysis also reports what is not known, frequency information for mixed population samples, quality metrics, ...