Indel-based Realignment

Improving the original alignments of the reads based on multiple sequence (re-)alignment
You are here in the GATK Best Practices workflow for *germline* variant discovery.

### Data Pre-processing

**Non-GATK**
- Raw Reads
  - Map to Reference
    - BWA mem
  - Mark Duplicates & Sort (Picard)
  - Indel Realignment
  - Base Recalibration
  - Analysis-Ready Reads

### Variant Discovery

**Analysis-Ready Reads**
- Var. Calling
  - *HC in ERC mode*
- Genotype Likelihoods
  - Joint Genotyping
  - Raw Variants
    - SNPs
    - Indels
  - Variant Recalibration
    - separately per variant type
- Analysis-Ready Variants
  - SNPs
  - Indels

### Callset Refinement

- Analysis-Ready Variants
  - SNPs & Indels
- Variant Evaluation
  - look good?
  - troubleshoot
  - use in project
InDels = insertion/deletion

**Ref seq**

A G C T A G G G G T C

**Sample seq**

AGCTAGGGTC

**Insertion**

A G C T A G G G G T C

**Deletion**

AGCGGTC
The problem we want to fix

Alignment by BWA

Several consecutive “SNPs” only found on reads ending on the right of the homopolymer

7bp “T” homopolymer run

After realignment

Adding a 1-bp insertion brings sanity to the entire alignment
Why does this happen?

- Mappers cannot “see” indels near ends of reads
  - Because mismatches are “cheaper” than a gap in this context

![DNA alignment diagram]

- Local realignment around indels -> most parsimonious alignment

- Improves accuracy of several downstream processing steps

```plaintext
Ref       T A C C C A T T T T T T C T A A A A G C T
BWA       C C A T T T T T T C T A A A A A C T
IR        C C A - T T T T T C T A A A A A C T
```

Mismatch = -1
Open gap   = -3
How do we identify where realignment is needed?

- Known sites (e.g. dbSNP, 1000 Genomes)
- Indels seen in original alignments (in CIGARs)
- Sites where evidence suggests a hidden indel
  - Entropy calculation identifies “messy areas”
How does the realignment algorithm work?

1. Find the best alternate consensus sequence that, together with the reference, best fits the reads in a pile (maximum of 1 indel)

2. Score for alternate consensus = total sum of quality scores of mismatching bases

3. If best alternate consensus is sufficiently better than the original alignments (using LOD score threshold) -> accept proposed realignment
Indel Realignment steps/tools

- Identify what regions need to be realigned
  → RealignerTargetCreator

- Perform the actual realignment
  → IndelRealigner
RealignerTargetCreator

- Pre-processing step to find intervals that may need realignment
- Input BAM file not necessary if processing only at known indels
- Using a list of known indels will both speed up processing and improve accuracy, but is not required

```
java -jar GenomeAnalysisTK.jar \
-T RealignerTargetCreator \
-R human.fasta \
-I original.bam \
-known indels.vcf \
-o realigner.intervals
```
IndelRealigner

- Attempts realignment at RealignerTargetCreator target intervals
- Must use same input file(s) used in RealignerTargetCreator step
- Processing options
  - Only at known indels: much faster, accurate for ~90-95% of indels
  - At indels seen in the original BAM alignments: the recommended mode
  - Using full Smith-Waterman realignment: most accurate, but heavy computational cost and not really necessary with the new techs

```
java –jar GenomeAnalysisTK.jar \
  –T IndelRealigner \
  –R human.fasta \
  –I original.bam \
  –known indels.vcf \
  –targetIntervals realigner.intervals \
  –o realigned.bam
```
This is what a realigned BAM looks like

Old data
(lower quality)

New data
(higher quality)

Can I see the effects of realignment?

- Indel Realigner changes the CIGAR string of realigned reads but maintains the original CIGAR (with OC tag)

-> Can grep for realigned regions and view in genome browser (IGV)

```
20GAVAAXX100126:1:67:10041:180738  99  20  10011431  70  87M1D14M = 10011720  390
TTAAATGTGTATTATCTATTGTTACACTGATTATAAAAATCAAAGATTATTTTCATGAAACTCAGTACCCCTTCAGGGAAAAAAAA
AAAAT
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
HHHHHHHHHHHHHGGGGGGGGG
NM:i:1  SM:i:37  XM:i:1  XO:i:0
BQ:Z:@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
`a`^\[Y MQ:i:60  XT:A:
```
Is realignment still necessary with latest software?

• Variant callers with reassembly step (HaplotypeCaller, MuTect 2, Platypus) do not require indel realignment

• BUT potential improvement for Base Quality Score Recalibration when run on realigned BAM files (artifactual SNPs are replaced with real indels).

• Also still useful for legacy tools
  – Unified Genotyper
  – MuTect 1
You are here in the GATK Best Practices workflow for germline variant discovery.
Further reading

http://www.broadinstitute.org/gatk/guide/best-practices

http://www.broadinstitute.org/gatk/guide/article?id=38

https://www.broadinstitute.org/gatk/gatkdocs/org_broadinstitute_gatk_tools_walkers_indels_IndelRealigner.php

https://www.broadinstitute.org/gatk/gatkdocs/org_broadinstitute_gatk_tools_walkers_indels_RealignerTargetCreator.php