DNA-seq analysis:
From raw reads to processed alignments

Elodie Girard
elodie.girard@curie.fr
U900 Institut Curie, INSERM, Mines ParisTech
Dataset

• Public data: exome sequenced by the International HapMap Project
• Single-end reads of 100bp, Illumina Genome Analyzer IIx
• Corresponding RNA-seq data available (Pickrell et al., Nature, 2010)

Objectives of this session:
1. Mapping the DNA-seq data to the reference genome
2. Process the alignments for the variant calling
Workflow

Raw reads (Fastq)

Mapping on the reference genome

Mapping Post-processing

Variant Calling Pre-processing

Variant Calling

Variant Filtering & Annotation
Session Workflow

Reads (Fastq) → Conversion to Galaxy Format → Groomer → Reference Genome (Fasta) → Mapping (BWA) → PCR duplicates Marking (MarkDup) → Target Intersection (Intersect Bam) → Preprocess GATK part 1 (Local realignment around indels) → Preprocess GATK part 2 (Base Quality Score Recalibration)

Aligned and preprocessed reads (BAM)
- Marked PCR duplicates
- Intersected on target regions
- Realigned around indels
- Recalibrated

Not for small targets / amplicons design!
Genome or Target DNA sequencing

**Whole Genome VS Exome Sequencing**

- Genome
- DNA fragmentation
- Hybridization
- Separation
- Elution

- Illumina technology (Hiseq2000/2500)

**Amplicon Sequencing**

- Sequencing of a dedicated panel of genes/hotspots
- PCR amplification
- Mostly IonTorrent technology (PGM/Proton)

Whole Genome VS Exome Sequencing

- Genomic DNA
- Exome DNA

- Hybridization
- Separation
- Elution

- Illumina technology (Hiseq2000/2500)

Amplicon Sequencing

- Sequencing of a dedicated panel of genes/hotspots
- PCR amplification
- Mostly IonTorrent technology (PGM/Proton)
Raw data

- **Raw data**: millions of short reads of the same size (Illumina) or different sizes (Ion PGM), single-end or paired-end

**Single-end**

- Read 1
- DNA fragment

**Paired-end**

- Read 1
- Read 2
- DNA fragment

ACTGATTAGTCTGAATTAGANNGATAGGAT

GATCGATGCATAGCGATCAGCATCGATACG

CGGCGCTCCGCTCTCGAAACTAGCACTGAC

AGCATCAGGATCTACGATCTAGCGAACTGAC

ACTAGCTACTATCGAGCTATCAGCGAGCATCTATC

CTGACTACTATCGAGCGAGCTACTAACTGAC

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN

ACTAGGCATCGGCATCAGCGACNNNNNNNNNN

ACTAGCTATCGAGCTATCAGCGAGCATCTATC

ACTAGCTACTATCGAGCCGGATCATCGAC

CTGACTACTATCGAGCGAGCTACTAACTGAC

NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
Sequence Encoding (FASTQ)

- Extension from traditional FASTA format

- Each block has 4 elements (in 4 lines):
  - Sequence name (read name, group, etc…)
  - Sequence
  - + (optional: sequence name again)
  - Associated quality scores (phred-scaled) : different encoding possible

- Example record:
  - @FCD19MJACXX:2:1101:1735:1993#GTTCGACA/1
  - NGAGGCTGAGGCAGGTCAGGAGATCGAGACCATC
  - +
  - BP\ccccc]ceeecheheeZbe_cZbd_dbbdd\aXab_`b
Sequence Encoding (FASTQ)

• The base calling (A, T, G or C) is performed based on Phred Scores.

<table>
<thead>
<tr>
<th>Phred Quality Score</th>
<th>Probability of Incorrect Base Call</th>
<th>Base Call Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1 in 10</td>
<td>90%</td>
</tr>
<tr>
<td>20</td>
<td>1 in 100</td>
<td>99%</td>
</tr>
<tr>
<td>30</td>
<td>1 in 1,000</td>
<td>99.9%</td>
</tr>
<tr>
<td>40</td>
<td>1 in 10,000</td>
<td>99.99%</td>
</tr>
<tr>
<td>50</td>
<td>1 in 100,000</td>
<td>99.999%</td>
</tr>
</tbody>
</table>

→ 1% error rate

• Phred scores provide log\(_{10}\)-transformed error probability values
  → If p is probability that the base call is wrong the Phred score is

\[
Q = -10 \log_{10} (P) \iff P = 10^{-Q/10}
\]
FASTQC: Quality control on raw data

- Quality control on specific metrics:
  - A] Read length distribution
  - B] Sequence content per base and % of GC
  - C] Quality score per base and D] over the reads
FASTQC: Quality control on raw data

- **Sequence Length Distribution**: sequencers produce same (Illumina) or different (PGM) read length. This metric helps identify abnormal read length

- **Sequence content**: % A/C/G/T ; %GC; %N at each position in the read
  
  → Proportion biased for targeted sequencing

- **Quality score**:  
  1. **Per base**: identify base calls with low quality (commonly towards the end of a read)  
  2. **Per sequence**: to see if a subset of your sequences have universally low quality values

- **K-mers content**: a k-mer is a motif of length k observed more than once in a sequence (repeats : ACACAC ; spaced occurrences : tccGAGGaaggGAGGaag)

- **Over-represented sequences**: highly duplicated sequence in your library (primer, adapter..)
Session Workflow

- **Reads (Fastq)**
- **Conversion to Galaxy Format**
  - Groomer
- **Reference Genome (Fasta)**
  - Mapping
  - BWA
- **PCR duplicates Marking**
- **Quality Control**
  - FastQC
- **Target regions (bed)**
  - Intersect Bam
- **Preprocess GATK part 1**
  - Local realignment around indels
- **Preprocess GATK part 2**
  - Base Quality Score Recalibration

- **Groomer**
- **Quality Control**
- **Mapping**
  - BWA
- **PCR duplicates Marking**
- **Taget Intersection**
  - Intersect Bam
- **Preprocess GATK part 1**
  - Local realignment around indels
- **Preprocess GATK part 2**
  - Base Quality Score Recalibration
Import Data

1. In the top menu, click on « **Shared Data** » then on « **Data Librairies** »

2. Click on « **TP Variant** » then select the 4 following files:
   → pickrell.fastq → chr12.fa → dbsnp[...].vcf → Mills[...].vcf

3. Click on « **Import to current history** » then « **Go** »
1. Rename your history to « **TP Variant** » by clicking on « **Unnamed history** » then « **Enter** »

2. In the left panel, click on « **FASTQ Groomer** » to convert the FASTQ into the FASTQ Sanger Format

3. Click on « **Execute** » to launch the conversion

→ **Sanger** is the current quality encoding for Illumina & IonTorrent
→ Check the quality encoding of older FASTQ
FASTQC : FASTQ Quality Control

1. In the left panel, click on « FASTQC: Read QC » (or use the search box)

2. Select the FASTQ in fastqsanger format and click on « Execute »

→ The result of FASTQC is an html page that you can view by clicking on the eye
Session Workflow

Reads (Fastq) → Conversion to Galaxy Format
   ⏫ Groomer

Reference Genome (Fasta) → Mapping
   ⏫ BWA

PCR duplicates Marking → MarkDup

Quality Control
   ⏫ FastQC

Target regions (bed) → Target Intersection
   ⏫ Intersect Bam

Mapping statistics
   ⏫ flagstat

Aligned and preprocessed reads (BAM)
   - Marked PCR duplicates
   - Intersected on target regions
   - Realigned around indels
   - Recalibrated

Preprocess GATK part 1
   ⏫ Local realignment around indels

Preprocess GATK part 2
   ⏫ Base Quality Score Recalibration

Not for small targets / amplicons design!
De Novo Reads Assembly

- **De Novo Reads Assembly**: used when there’s no reference genome; aims at reconstructing long scaffolds from single reads

The genome is assembled from overlapping, matching reads sequences.
Mapping on a reference Genome

• Reads are aligned $\geq 1$ times on the reference genome

• A mapping quality is associated to each alignment:
  - Quantify the probability that the alignment is correct
  - Decreases with the number of mismatches (wrong nucleotide) & gaps (small insertions/deletions) & the number of alignments
Reads Alignment - Vocabulary

**Mapping method:** seed & extend

1. Align the seed (small part of the read)
2. Extend the seed to align the whole read

**Mismatch:** Incoherence between two nucleotides

**Indels:** Insertion/Deletion into the reference genome

**Gap:** Bridge within the read alignment (i.e. small indels)

**Mappability:** Uniqueness of a region

- repeated region = low mappability
- unique region = good mappability
Multiple Alignments

• A read can align **multiple times** on the genome (repeated elements…)
• How to deal with these multiple alignments reads ?
• Three strategies:
  -1- Report only unique alignment
  -2- Report best alignments & randomly assign reads across equally good loci
  -3- Report all (best) alignments

- Mapping Quality: quantify the probability that a read is misplaced.
  → Low if a read has multiple alignments
Statistics used as Quality Control

- **Depth of coverage** = mean number of reads covering a base (X)
  Example: 30X for normal sample, 100X for tumor sample

- **Coverage** = part of the reference with at least one read
  Example: \( \geq 80\% \) of your exome target is covered by 20X
Alignment tools

- Multitude of alignment tools: BWA, Bowtie, Bowtie2, Bfast…
- How to choose the best tool?
  - Is my sequencing technology supported?
  - Do I have short or long reads? Reads of different sizes?
  - Do I want to allow gapped alignment? Multiple alignments?
  - Does it support single/paired-end reads?
  - On which alignment algorithm is it based?
  - Computational issues? Is it used by the community?

- A classical and performant tool for Illumina sequencing: BWA (Burrows-Wheeler Aligner)
Mapping with BWA

1: «Use one from the history»

2: Reference Genome = «chr12.fa»

3: «FASTQ Groomer»

4: «Full Parameter list»
Adding Read Group

Adding a **Read Group** is **mandatory** for some tools (GATK)

→ Informations on the sample, library used, platform…

→ Attribute a read to a sample (useful in multi-samples study)

---

At the bottom of the parameters list

Select « Yes »
Adding Read Group

- **ID** and **Platform** are the most important but fill every **required** box

Specify the read group for

Yes

Read group identifier (ID). alignment records. Must be

ID1

Required if RG specified. Required if RG specified. Rea

Library name (LB):

Library1

Required if RG specified. Use pool name where a pool is being

Platform/technology used to produce the reads (PL):

ILLUMINA

Required if RG specified. Valid values: CAPILLARY, LS454, ILL

Platform unit (PU):

PU1

Optional. Unique identifier (e.g. flowcell-barcode.lane for Ili

Sample (SM):

Pickrell

Required if RG specified. Use pool name where a pool is being

Suppress the header in the output SAM file:

BWA produces SAM with several lines of header information

Click on Execute
[Optional] Adding Read Group

- Use « **Add or Replace Groups** » from « NGS:Picard » to associate a sample ID and a sequencing technology to the reads

**Add or Replace Groups (version 1.56.0)**

- **SAM/BAM dataset to add or replace read groups in:**
  16: Intersection of MarkDups_Dupes Marked.bam and exome_regions.bed

- **Read group platform unit:**
  - PU1: run barcode, etc.

- **Read group ID (ID tag):**
  - ID1

- **Read group sample name (SM tag):**
  - Pickrell

- **Read group library (LB tag):**
  - Library1

- **Read group platform (PL tag):**
  - ILLUMINA: Illumina, so10d, 454, pacbio, helicos

**Specify additional (optional) arguments:**

- **Output bam instead of sam:**
  - Check box: Uncheck for sam output

**Execute**
SAM/BAM aligned format

- **SAM Format**: aligned format, human readable

  @SQ SN:chr12 LN:133851895
  @RG ID:Sample_ID LB:Sample_Library PL:ILLUMINA SM:Sample_Name PU:Platform_Unit

  Read name  Flag  Chr  5’ pos  MAPQ  Cigar  paired  5’ pos of the mate  Insert size
  ERR166338.1 99  chr12  82670685  23  101M = 82670850  266
  GCCCCTGGGGATTTTTGCACCAAGCCACTGTCTCCAGCTGG  sequence
  BBC@GIIHGCFCIEHEAIEIFFGEONDNJFINIONHNGJNNNNKNJN  Base quality
  RG:Z:Sample_ID

  XT:A:U NM:i:0 X0:i:1 X1:i:1 XM:i:0 XO:i:0 XG:i:0 MD:Z:100 XA:Z  tags

  Group affiliation

- **BAM Format**: Binary SAM Format (not human readable but compressed = smaller)
Choose the source for the reference list:

1: Choose History

Convert SAM file:

2: Mapping output « MAP with BWA »

Using reference file:

3: « Chr12 .fa »

4: Click on Execute
Removing Duplicates

- **Duplicates reads**: different reads having the same sequence caused by PCR amplification during sequencing library preparation.

- The removal of the duplicates depends on the application (not suitable for sequencing on small target).

- **Galaxy**: Use **“Mark Duplicates reads”** from **“NGS:Picard”** to **mark** duplicates (don’t remove them).

  → If duplicates are marked, samtools and GATK tools will ignore them.

- **Galaxy**: Run **“Flagstat”** on the output BAM to see the number of PCR duplicates.
Removing Duplicates

1: Choose the SAM-to-Bam output

2: Check this box to remove duplicates

3: Click on Execute
Mapping Statistics

**Flagstat**: a tool showing some mapping statistics (from the SAMtools software)

Choose the Sam-to-BAM output and execute

% of mapped reads

Properly paired reads:
- 0<= Insert size <= Max size
- Reads on same chromosome
- Reads facing each other
- Both reads are mapped
Target intersection

- Use « **Intersect BAM alignments with intervals** » from « **NGS:Bedtools** » to keep only the reads mapped on the targeted regions
  - Smaller BAM size
  - The targeted regions must be in BED format
    - 4 columns (tabulated): chr; start; end; name

![Intersect BAM alignments with intervals in another files (version 0.1.0)](image)
Session Workflow

Reference Genome (Fasta) -> Mapping (BWA) -> PCR duplicates Marking (MarkDup) -> Target Intersection (Intersect Bam) -> Aligned and preprocessed reads (BAM)

- Marked PCR duplicates
- Intersected on target regions
- Realigned around indels
- Recalibrated

Reads (Fastq) -> Conversion to Galaxy Format (Groomer) -> Quality Control (FastQC) -> Target regions (bed)

Preprocess GATK part 1 (Local realignment around indels) -> Preprocess GATK part 2 (Base Quality Score Recalibration)

Not for small targets / amplicons design!
Why realign around indels?

- **Small Insertion/deletion (Indels)** in reads (especially near the ends) can trick the mappers into mis-aligning with mismatches
  - The mapper has to find a trade-off between adding several mismatches and opening one gap
  - **Alignment scoring** – cheaper to introduce multiple **Single Nucleotide Variants (SNVs)** than an indel: induce a lot of false positive SNVs
  - The mapper align reads (or pairs) independently

**Realignment around indels helps improve the accuracy of the alignment by having a **global** view of the alignment at one specific position**
Local realignment around indels
Three types of realignment targets

- The idea is 1) to identify loci in need of local realignment then 2) apply realignment

- Types of realignment targets:

  - **Known sites**: Common polymorphisms (dbSNP, 1000Genomes)

    ➔ On Galaxy: add « dbsnp[...].vcf » & « Mills[...].vcf » as Binding Reference Ordered Data in GATK

  - **Indels** seen in original alignments (in CIGAR, indicated by I for Insertion or D for Deletion)

  - Sites where evidences suggest a hidden indel (lot of SNVs in a small loci)
Local realignment around indels

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

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

7bp “T” homopolymer run
Local realignment around indels
Session Workflow

**Reads (Fastq)**
- Conversion to Galaxy Format
  - Groomer
- Quality Control
  - FastQC

**Alignment**
- Reference Genome (Fasta)
  - Mapping
    - BWA
  - PCR duplicates Marking
    - Not for small targets / amplicons design!
  - MarkDup

**Target Intersection**
- Target regions (bed)
  - Intersect Bam

**Preprocess GATK part 1**
- Local realignment around indels

**Preprocess GATK part 2**
- Base Quality Score Recalibration

**Aligned and preprocessed reads (BAM)**
- Marked PCR duplicates
- Intersected on target regions
- Realigned around indels
- Recalibrated
1. Identify what regions need to be realigned

   ➔ RealignerTargetCreator
   + known sites

---

2. Perform the actual realignment (BAM output)

   ➔ IndelRealigner
GATK: Realigner Target Creator

Choose the source for the reference list:
- History

BAM file:
- 17: Intersection of MarkDups_Dupes Marked.bam and target <target>

Using reference file:
- 2: chr12.fa

Known Variants
Using data sets of known variants (<known>, --known <known>)

Known Variants 1
- Variant Type: dbSNP
- Variant file (VCF format):
  - 7: dbsnps137.hg19_chr12.vcf

Add new binding for reference-ordered data

Known Variants 2
- Variant Type: INDELS
- Variant file (VCF format): 4: Mills_and_1000G_gold_standard.indels.hg19_chr12.vcf

Remove Known Variants 2
- Add new Known Variants
- Basic or Advanced GATK options: Advanced

Operate on Genomic intervals
- L, --intervals <intervals>

Operate on Genomic intervals 1
- Genomic intervals:
  - 15: targeted_regions.bed
  - Execute

Choose advanced GATK options
Add new operate on Genomic Intervals
GATK: Indel Realigner

Add new binding for reference-ordered datas

Choose advanced GATK options

Add new operate on Genomic Intervals

Execute
GATK Preprocess: Base Quality Score Recalibration

- Analyze covariation among several features of a base, e.g:
  - Original quality score
  - Position within the read (machine cycle)
  - Preceding and current nucleotides (chemistry effect)
  - Sequencing technology…

- Adjust the quality score associated to each sequenced base to be more accurate

  → Remove systematic biases
BQSR tools

1. Calculate the covariates
   ➔ BaseRecalibrator
   + known sites

1. Apply the covariates to the alignments
   ➔ PrintReads
GATK: Base Recalibrator

- Add new binding for reference-ordered data (dbsnp.vcf & Mills.vcf)
- Choose advanced GATK options

Base Recalibrator (version 2.8.0)

Choose the source for the reference list:
- History

BAM file:
- 20: Indel Realigner on data 2, data 7, and others (BAM)

Using reference file:
- 2: chr12.fasta

Covariates to be used in the recalibration:
- Select All
- ContextCovariate
- CycleCovariate
- RepeatLengthCovariate
- RepeatUnitCovariate
- RepeatUnitAndLengthCovariate

Operate on Genomic intervals 1
- 15: targeted_regions.bed

Execute
GATK: Print Reads

Print Reads (version 2.8.0)

Covariates table recalibration file:

- 22: Base Recalibrator on data 2, data 20, and others (Covariate File)
  The input covariates table file which enables on-the-fly base quality score
  (intended for use with BaseRecalibrator files) (-BQSR, --BQSR)

Choose the source for the reference list:

- History

BAM file:

- 20: Indel Realigner on data 2, data 7, and others (BAM)
  -I, --input_file <input_file>

Using reference file:

- 2: chr12.fa
  -R, --reference_sequence <reference_sequence>

Basic or Advanced GATK options:

- Advanced

Operate on Genomic intervals

Genomic intervals:

- 15: targeted_regions.bed
  Remove Operate on Genomic intervals

Execute

Add new operate on Genomic Intervals
Session Workflow

1. **Reads (Fastq)**
   - Conversion to Galaxy Format
     - Groomer
   - Quality Control
     - FastQC

2. **Reference Genome (Fasta)**
   - Mapping
     - BWA

3. **PCR duplicates Marking**
   - MarkDup

4. **Target Intersection**
   - Intersect Bam

5. **Preprocess GATK part 1**
   - Local realignment around indels

6. **Preprocess GATK part 2**
   - Base Quality Score Recalibration

**Aligned and preprocessed reads (BAM)**
- Marked PCR duplicates
- Intersected on target regions
- Realigned around indels
- Recalibrated

*Not for small targets / amplicons design!*