



LEIDEN UNIVERSITY MEDICAL CENTER

# **GAPSS3 Exome Sequencing Pipeline**

**Jeroen F. J. Laros**

**Leiden Genome Technology Center**

**Department of Human Genetics**

**Center for Human and Clinical Genetics**



In *exome sequencing*, we select genomic regions of interest using a *target-enrichment strategy*.

- PCR.
- On array capture.
- **In-solution capture.**

In *exome sequencing*, we select genomic regions of interest using a *target-enrichment strategy*.

- PCR.
- On array capture.
- **In-solution capture.**

Overview of an in-solution capture.

- Fragmentation.
- Size selection.
- Linker ligation.
- Capture.

In *exome sequencing*, we select genomic regions of interest using a *target-enrichment strategy*.

- PCR.
- On array capture.
- **In-solution capture.**

Overview of an in-solution capture.

- Fragmentation.
- Size selection.
- Linker ligation.
- Capture.

These regions are then *sequenced*.

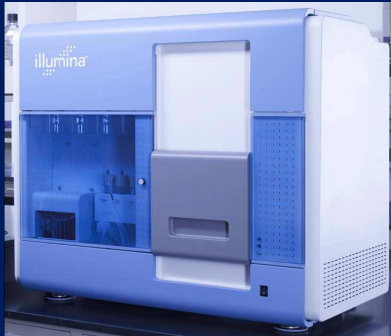
- Illumina Genome Analyser II (GAII).
- Illumina HiSeq 2000.

# Introduction

## The Illumina Genome Analyser II.



## The Illumina Genome Analyser II.



- Manufacturer: Illumina, Inc.
- Commercially available since 2005.
- Per cycle, one base is read.
- Reads up to  $100 \times 2$  base pairs.
- Takes about 8 days.
- Produces about 40 Giga bases per run.

## The Illumina Genome Analyser II.



- Manufacturer: Illumina, Inc.
- Commercially available since 2005.
- Per cycle, one base is read.
- Reads up to  $100 \times 2$  base pairs.
- Takes about 8 days.
- Produces about 40 Giga bases per run.

### Pros:

- Does paired end sequencing.
- Cheap.

## The Illumina HiSeq 2000.





## The Illumina HiSeq 2000.



- Manufacturer: Illumina, Inc.
- Commercially available since 2010.
- Per cycle, one base is read.
- Reads up to  $150 \times 2$  base pairs.
- Takes about 8 days.
- Produces about 150 Giga bases per run.

## Introduction

### The Illumina HiSeq 2000.



- Manufacturer: Illumina, Inc.
- Commercially available since 2010.
- Per cycle, one base is read.
- Reads up to  $150 \times 2$  base pairs.
- Takes about 8 days.
- Produces about 150 Giga bases per run.

### Pros:

- Even higher throughput.

Exome sequencing pipelines can roughly be divided in five steps.

## 1. Pre-alignment.

- Quality control.
- Data cleaning.

Exome sequencing pipelines can roughly be divided in five steps.

1. Pre-alignment.
  - Quality control.
  - Data cleaning.
2. Alignment.
  - Post-alignment quality control.

Exome sequencing pipelines can roughly be divided in five steps.

1. Pre-alignment.
  - Quality control.
  - Data cleaning.
2. Alignment.
  - Post-alignment quality control.
3. Variant calling.

Exome sequencing pipelines can roughly be divided in five steps.

1. Pre-alignment.
  - Quality control.
  - Data cleaning.
2. Alignment.
  - Post-alignment quality control.
3. Variant calling.
4. Filtering.
  - Post-variant calling quality control.

Exome sequencing pipelines can roughly be divided in five steps.

1. Pre-alignment.
  - Quality control.
  - Data cleaning.
2. Alignment.
  - Post-alignment quality control.
3. Variant calling.
4. Filtering.
  - Post-variant calling quality control.
5. Annotation.

## Pre-alignment

We use the FASTX toolkit for data cleaning.

- Remove linker sequences.
- Clip low quality reads at the end of the read.
- Judge the read that is left over.



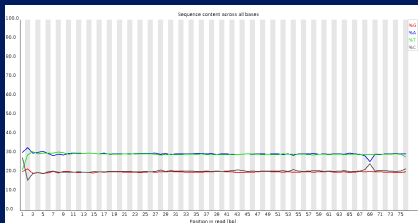
We use the FASTX toolkit for data cleaning.

- Remove linker sequences.
- Clip low quality reads at the end of the read.
- Judge the read that is left over.

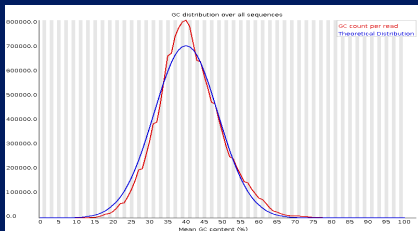
The FASTQC toolkit is used for quality control (both before and after the data cleaning step).

- GC content.
- GC distribution.
- Quality scores distribution.
- ...

# Pre-alignment

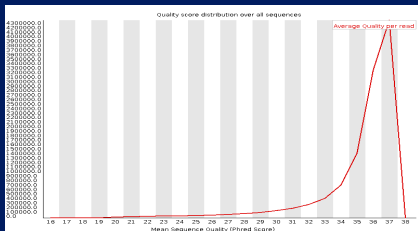


Per base sequence content.

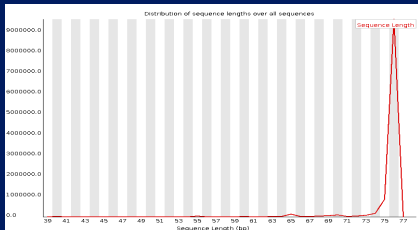


Per sequence GC content.

# Pre-alignment



Per sequence quality.



Sequence length distribution.

Stampy: A statistical algorithm for sensitive and fast mapping of Illumina sequence reads.

Stampy: A statistical algorithm for sensitive and fast mapping of Illumina sequence reads.

Some features:

- Base quality recalibration.
  - First map 1% of the input.
  - Recalibrate the Fastq quality scores.
  - Redo the alignment with the recalibrated scores.

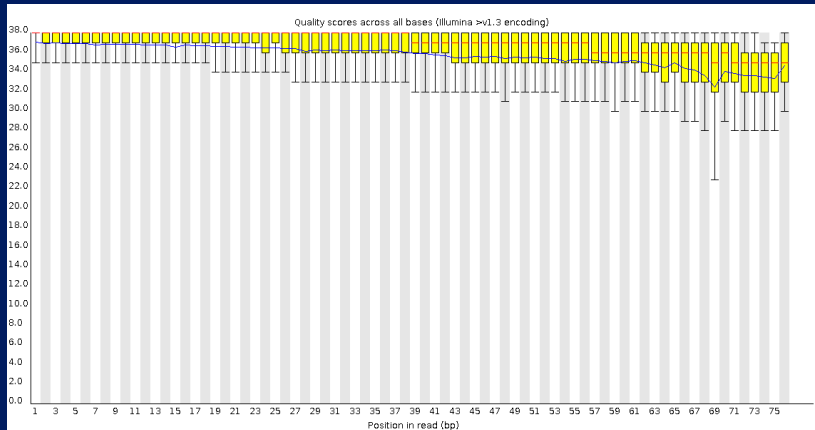
Stampy: A statistical algorithm for sensitive and fast mapping of Illumina sequence reads.

Some features:

- Base quality recalibration.
  - First map 1% of the input.
  - Recalibrate the Fastq quality scores.
  - Redo the alignment with the recalibrated scores.
- Uses BWA for the hard work.
  - Switches to its accurate built in aligner when BWA fails.

*Burrows-Wheeler Aligner* (BWA) is a short read aligner that allows small insertions and deletions.

## Base quality recalibration.



## Variant calling

Variant calling is done by Samtools, BCFtools / VCFutils.

The output of most modern aligners is in *Sequence Alignment / Map* (SAM) format.



## Variant calling

Variant calling is done by Samtools, BCFtools / VCFutils.

The output of most modern aligners is in *Sequence Alignment / Map* (SAM) format.

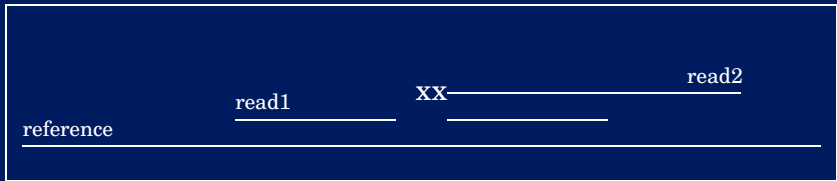
Mainly file format conversions.

- **SAM** → BAM.
- BAM → BAM.sorted.
- BAM.sorted → BAM.sorted.index.
- BAM.sorted → mpileup (**BAQ realignment**).
- BAM.sorted → BCF.
- BCF → **VCF**.

We end up with a list in *Variant Call Format* (VCF).

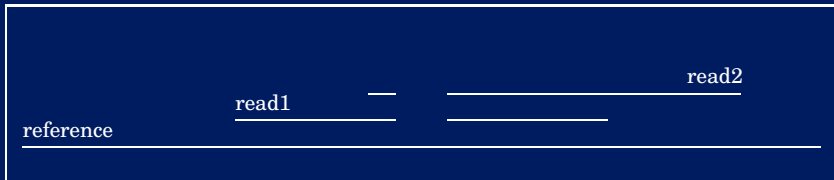
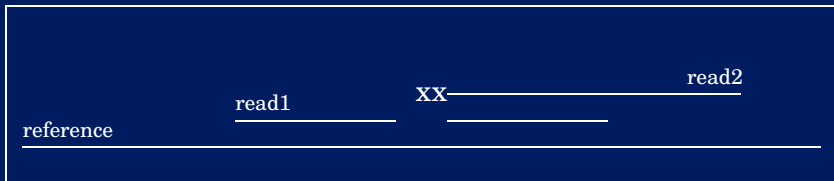
## Variant calling

*Base Alignment Quality (BAQ) realignment:*  
 Remove SNPs around indels.



## Variant calling

*Base Alignment Quality (BAQ) realignment:*  
 Remove SNPs around indels.



Samtools varfilter.

- Minimum coverage threshold.
- Strand bias.
- Quality scores.
- **Maximum coverage threshold.**
  - Copy number variation.
  - Alignment artefacts.

Samtools varfilter.

- Minimum coverage threshold.
- Strand bias.
- Quality scores.
- **Maximum coverage threshold.**
  - Copy number variation.
  - Alignment artefacts.

Still working on:

- Maximum coverage per region.
  - Probe affinity can vary greatly.

We use five annotation sources.

- Seattle Seq.
- Ensembl.
- Mutalyzer / SVEP.
- LOVD.
- In house database.

We use five annotation sources.

- Seattle Seq.
- Ensembl.
- Mutalyzer / SVEP.
- LOVD.
- In house database.
  - HGMD data.
  - 1000 genomes project.
  - Genome of the Netherlands (250 triplets).
  - All variants called by this pipeline.

We use five annotation sources.

- Seattle Seq.
- Ensembl.
- Mutalyzer / SVEP.
- LOVD.
- In house database.
  - HGMD data.
  - 1000 genomes project.
  - Genome of the Netherlands (250 triplets).
  - All variants called by this pipeline.
    - Coverage per variant.
    - Number of reads supporting the variant.
    - Horizontal coverage per sample.



Some implementation details.

- Framework in *bash*.
  - Stand alone scripts written in other languages (Perl, Python, ...).

Some implementation details.

- Framework in *bash*.
  - Stand alone scripts written in other languages (Perl, Python, ...).
- *Sun grid engine* to submit jobs to our local cluster.

Some implementation details.

- Framework in *bash*.
  - Stand alone scripts written in other languages (Perl, Python, ...).
- *Sun grid engine* to submit jobs to our local cluster.
- Database to keep track of the versions of all used tools and custom scripts.
  - If one or more tools are upgraded, the new versions are stored.
  - The version number of the pipeline is incremented.
  - The versions of all tools of all pipeline versions can be retrieved from this database.

Some implementation details.

- Framework in *bash*.
  - Stand alone scripts written in other languages (Perl, Python, ...).
- *Sun grid engine* to submit jobs to our local cluster.
- Database to keep track of the versions of all used tools and custom scripts.
  - If one or more tools are upgraded, the new versions are stored.
  - The version number of the pipeline is incremented.
  - The versions of all tools of all pipeline versions can be retrieved from this database.
- $\text{\LaTeX}$  documentation is automatically generated.
  - Compiled to pdf that can be handed over to the customer.

Some implementation details.

- Framework in *bash*.
  - Stand alone scripts written in other languages (Perl, Python, ...).
- *Sun grid engine* to submit jobs to our local cluster.
- Database to keep track of the versions of all used tools and custom scripts.
  - If one or more tools are upgraded, the new versions are stored.
  - The version number of the pipeline is incremented.
  - The versions of all tools of all pipeline versions can be retrieved from this database.
- $\text{\LaTeX}$  documentation is automatically generated.
  - Compiled to pdf that can be handed over to the customer.
- All individual commands are logged.

## Acknowledgements

Michiel van Galen  
Yu-Ching Lai  
Martijn Vermaat  
Bradley ten Broeke  
Jaap van der Heijden  
Michel Villerius  
Matthew Hestand  
Johan den Dunnen

<http://www.lgtc.nl>