



LEIDEN UNIVERSITY MEDICAL CENTER

# Combining tools into a pipeline

**Jeroen F. J. Laros**

**Leiden Genome Technology Center**

**Department of Human Genetics**

**Center for Human and Clinical Genetics**



## *Pipelines*



Figure 1 : A real-life pipeline.

## *Pipelines*



Figure 2 : Scene from “Modern times”.

## *Pipelines*

Combining tools:

- The output of one tool can serve as the input for another.
- Not necessarily linear.
- ...

## *Pipelines*

Combining tools:

- The output of one tool can serve as the input for another.
- Not necessarily linear.
- ...

Running various different tools:

- Two or three different aligners.
- A couple of variant callers.
- ...

## *Running example: Exome sequencing*

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

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

## *Running example: Exome sequencing*

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.

## *Running example: Exome sequencing*

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*.



## *Sequencers: HiSeq*



Figure 3 : HiSeq 2000.

### Characteristics:

- High throughput.
- Paired end.
- High accuracy.
- Read length  $2 \times 150\text{bp}$ .
- Relatively long run time.
- Relatively expensive.

## *Sequencers: Ion Torrent*



Figure 4 : Ion torrent.

### Characteristics:

- Moderate throughput.
- Single end (for now).
- High accuracy.
- Read length  $\pm 200\text{bp}$ .
- Short run time.
- Cheap runs.

## *Data analysis*

Resequencing pipelines can roughly be divided in five steps.

## *Data analysis*

Resequencing pipelines can roughly be divided in five steps.

1. Pre-alignment.

- Quality control.
- Data cleaning.

## *Data analysis*

Resequencing pipelines can roughly be divided in five steps.

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

## *Data analysis*

Resequencing pipelines can roughly be divided in five steps.

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

## *Data analysis*

Resequencing 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.

*Data analysis*

Resequencing 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.



## Trimming

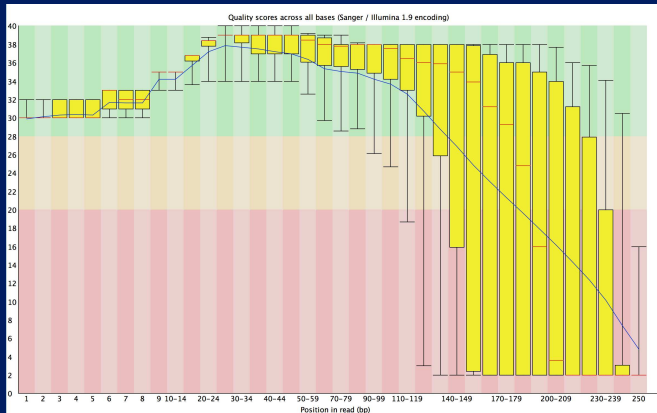


Figure 5 : Quality score per position.

## Clipping

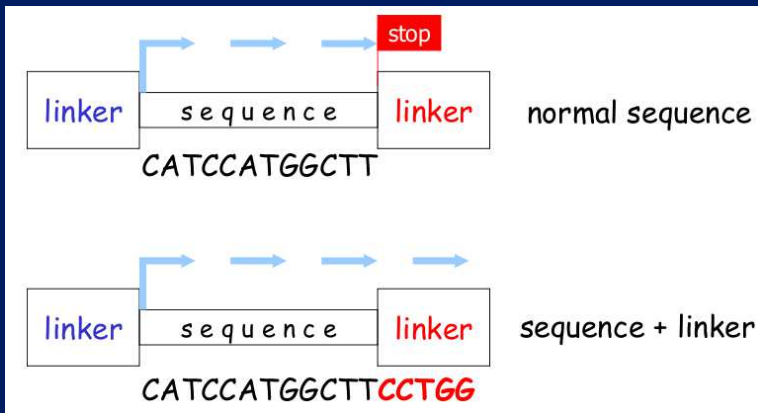


Figure 6 : Sequencing linkers.

## *Data cleaning and QC*

Depending on the sequencing platform, parts of the reads need to be removed.

- Remove linker sequences (*Cutadapt*, *FASTX toolkit*).
- Clip low quality reads at the end of the read (*Sickle*, *Trimmomatic*, *FASTX toolkit*).
- Length filtering (*Fastools*).

## *Data cleaning and QC*

Depending on the sequencing platform, parts of the reads need to be removed.

- Remove linker sequences (*Cutadapt*, *FASTX toolkit*).
- Clip low quality reads at the end of the read (*Sickle*, *Trimmomatic*, *FASTX toolkit*).
- Length filtering (*Fastools*).

The *FastQC toolkit* can be used for quality control (both before and after the data cleaning step).

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

# Pre-alignment

## Example QC output

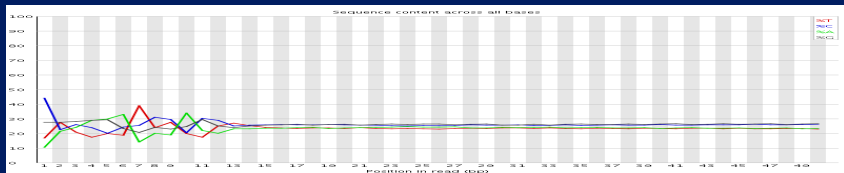


Figure 7 : Per base sequence content.

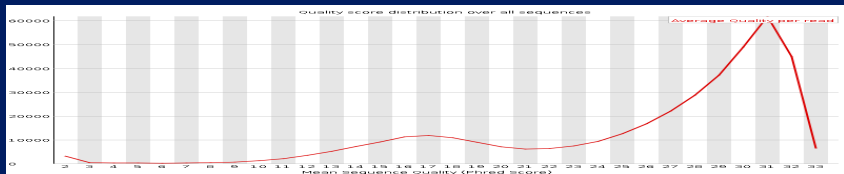


Figure 8 : Per sequence quality.

*Choose an aligner*

Alignment needs to be fault-tolerant.

## *Choose an aligner*

Alignment needs to be fault-tolerant.

Not all aligners can deal with indels.

- Only a couple of years ago, only SNPs were considered.
  - *Bowtie*.

## *Choose an aligner*

Alignment needs to be fault-tolerant.

Not all aligners can deal with indels.

- Only a couple of years ago, only SNPs were considered.
  - *Bowtie*.

Few aligners can work with large deletions.

- Spliced RNA.
  - *GMAP* / *GSNAP*.
  - *Tophat*.
- *BWA-MEM*.



## *Choose an aligner*

Alignment needs to be fault-tolerant.

Not all aligners can deal with indels.

- Only a couple of years ago, only SNPs were considered.
  - *Bowtie*.

Few aligners can work with large deletions.

- Spliced RNA.
  - *GMAP* / *GSNAP*.
  - *Tophat*.
- *BWA-MEM*.

The choice of aligner may be restricted by the sequencer.

- For the Ion Torrent: *Tmap*.
- For the PacBio: *BLASR*.

# Variant calling

## *Pileup*

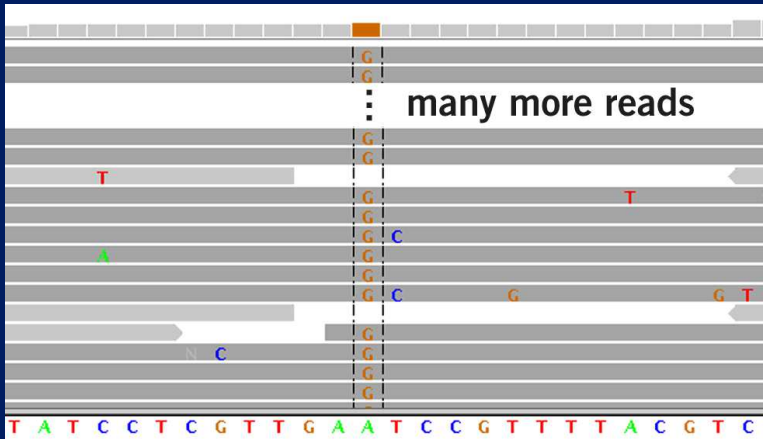


Figure 9 : Result of an alignment.

## *Some considerations*

Things a variant caller might take into account:

- Strand balance.
- Base quality.
- Mapping quality.
  - Distribution within the reads.
- Ploidity of the organism in question.

## *Some considerations*

Things a variant caller might take into account:

- Strand balance.
- Base quality.
- Mapping quality.
  - Distribution within the reads.
- Ploidy of the organism in question.

Complicating factors:

- Pooled samples.

## *Some considerations*

Things a variant caller might take into account:

- Strand balance.
- Base quality.
- Mapping quality.
  - Distribution within the reads.
- Ploidity of the organism in question.

Complicating factors:

- Pooled samples.
- RNA.
  - Allele specific expression.
  - RNA editing.

## *Some considerations*

Things a variant caller might take into account:

- Strand balance.
- Base quality.
- Mapping quality.
  - Distribution within the reads.
- Ploidity of the organism in question.

Complicating factors:

- Pooled samples.
- RNA.
  - Allele specific expression.
  - RNA editing.
- Strand specific sampleprep.

## *Choice of variant caller*

Rules of thumb:

- Well known organism and experiment: Statistical model.
- Use a simpler variant caller otherwise.

## *Choice of variant caller*

Rules of thumb:

- Well known organism and experiment: Statistical model.
- Use a simpler variant caller otherwise.

Popular variant callers:

- *Samtools*.
- *GATK*.
- *VarScan*.



## *Filtering on coverage*

We can set some thresholds:

- Minimum.
- Maximum.

## *Filtering on coverage*

We can set some thresholds:

- Minimum.
- Maximum.

We filter for a maximum coverage because of copy number variation.

## *Filtering on coverage*

We can set some thresholds:

- Minimum.
- Maximum.

We filter for a maximum coverage because of copy number variation.

A good way to calculate the maximum:

- Calculate the mean coverage.
  - Only of the covered (targeted) regions.
- Multiply this number with a reasonable factor e.g., 2.5.

## *What is already known about a variant*

A selection of SeattleSeq annotation:

- Is the variant known?
- Does it hit a gene?

## *What is already known about a variant*

A selection of SeattleSeq annotation:

- Is the variant known?
- Does it hit a gene?
  - Is it in an intron?
    - Does it hit a splice site?

## *What is already known about a variant*

A selection of SeattleSeq annotation:

- Is the variant known?
- Does it hit a gene?
  - Is it in an intron?
    - Does it hit a splice site?
- Is it in the coding region?
  - Is there a gain/loss of a stop codon?
  - Does the variant result in a frameshift?
  - ...

## *What is already known about a variant*

A selection of SeattleSeq annotation:

- Is the variant known?
- Does it hit a gene?
  - Is it in an intron?
    - Does it hit a splice site?
  - Is it in the coding region?
    - Is there a gain/loss of a stop codon?
    - Does the variant result in a frameshift?
    - ...
- Is it in the 5'/3' UTR of a gene?
- ...

## *What is already known about a variant*

A selection of SeattleSeq annotation:

- Is the variant known?
- Does it hit a gene?
  - Is it in an intron?
    - Does it hit a splice site?
  - Is it in the coding region?
    - Is there a gain/loss of a stop codon?
    - Does the variant result in a frameshift?
    - ...
  - Is it in the 5'/3' UTR of a gene?
  - ...
- Is it in a regulatory region?
- ...



## *Combining tools*

```
1  bwa aln -t 8 $reference $i > $i.sai
2  bwa samse $reference $i.sai $i > $i.sam
3  samtools view -bt $reference -o $i.bam $i.sam
```

Listing 1 : Shell script

## *Combining tools*

```

1  bwa aln -t 8 $reference $i > $i.sai
2  bwa samse $reference $i.sai $i > $i.sam
3  samtools view -bt $reference -o $i.bam $i.sam

```

Listing 1 : Shell script

```

1  %.sai: %.fq
2      $(BWA) aln -t $(THREADS) $(call MKREF, $@) $< > $@
3
4  %.sam: %.sai %.fq
5      $(BWA) samse $(call MKREF, $@) $^ > $@
6
7  %.bam: %.sam
8      $(SAMTOOLS) view -bt $(call MKREF, $@) -o $@ $<

```

Listing 2 : Makefile

## *Galaxy*

Galaxy: a graphical user interface:

- Wrapper for command line utilities.
- User friendly.
- Point and click.

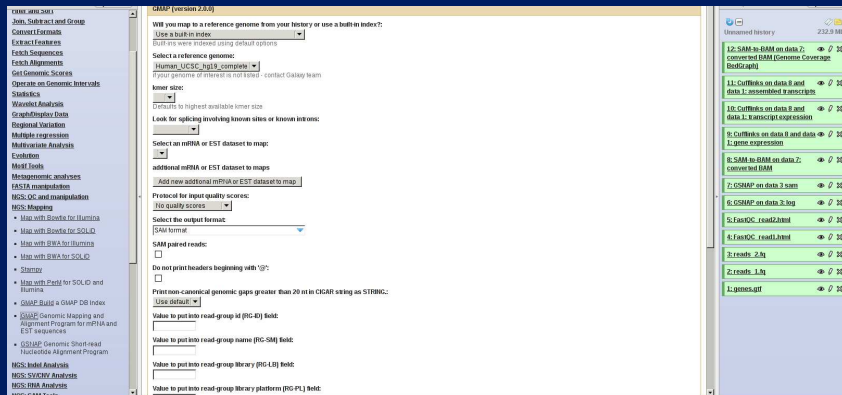
<http://galaxy.psu.edu/>

## *Galaxy*

Galaxy: a graphical user interface:

- Wrapper for command line utilities.
- User friendly.
- Point and click.
- Workflows.
  - Save all the steps you did in your analysis.
  - Rerun the entire analysis on a new dataset.
  - Share your workflow with other people.
  - ...

<http://galaxy.psu.edu/>



The screenshot displays the Galaxy web interface. On the left is a navigation sidebar with categories like 'Tools, Subtools and Groups', 'Workflows', 'Regional Variation', 'Multiple Sequences', 'Evolution', 'Metagenomic analyses', 'FASTA manipulation', 'NGS: QC and manipulation', 'NGS: Mapping', 'NGS: Index Analysis', 'NGS: SV/CRV Analysis', 'NGS: RNA Analysis', and 'NGS: BAM Tools'. The main panel shows the 'GMAP (version 2.8.0)' tool configuration page. It includes sections for 'Will you map to a reference genome from your history or use a built-in index?', 'Select a reference genome:' (set to 'Human\_UCSC\_hg19\_complete'), 'kmer size:', 'Look for splicing involving known sites or known introns:', 'Select an mRNA or EST dataset to map:', 'additional mRNA or EST dataset to maps', 'Protocol for input quality scores:', 'Select the output format:', 'SAM paired reads:', 'Do not print headers beginning with @:', 'Print non-canonical genomic gaps greater than 20 nt in CIGAR string as STRING:', 'Value to put into read-group id (RG-ID) field:', 'Value to put into read-group name (RG-SM) field:', 'Value to put into read-group library (RG-LB) field:', and 'Value to put into read-group library platform (RG-PL) field:'. The right sidebar shows an 'Unmapped history' with a list of 12 items, including '12: SAM-to-BAM on data 2; converted BAM (Genome Coverage bedGraph)', '11: Cufflinks on data 8 and data 1: assembled transcripts', '10: Cufflinks on data 8 and data 1: transcript expression', '9: Cufflinks on data 8 and data 1: gene expression', '8: SAM to BAM on data 2; converted BAM', '7: OSNAP on data 2: sam', '6: OSNAP on data 2: log', '5: FastQC\_read2.html', '4: FastQC\_read1.html', '3: reads\_2.bq', '2: reads\_1.bq', and '1: genes.gtf'.

Figure 10 : Galaxy main user interface

MPileup

**Compute genotype likelihoods:**

True ▼

Compute genotype likelihoods and output them in the binary call format (BCF).

**Output uncompressed BCF:**

True ▼

Similar to the Genotype parameter, except that the output is uncompressed BCF, which is preferred for piping.

**Input :**

▼

Execute

Generate BCF or pileup for one or multiple BAM files. Alignment records are grouped by sample identifiers in @RG header lines. If sample identifiers are absent, each input file is regarded as one sample.

**Generated By:**

LUMC Interface Generator (0.1)

2011-09-03T14:29:36.793452Z

**Based On:**

RDF Definition of "MPileup"

2011-09-02T16:17:29.010890Z

Figure 11 : User friendly interface with Galaxy

# Graphical interfaces

## *Workflow of a parallel pipeline*

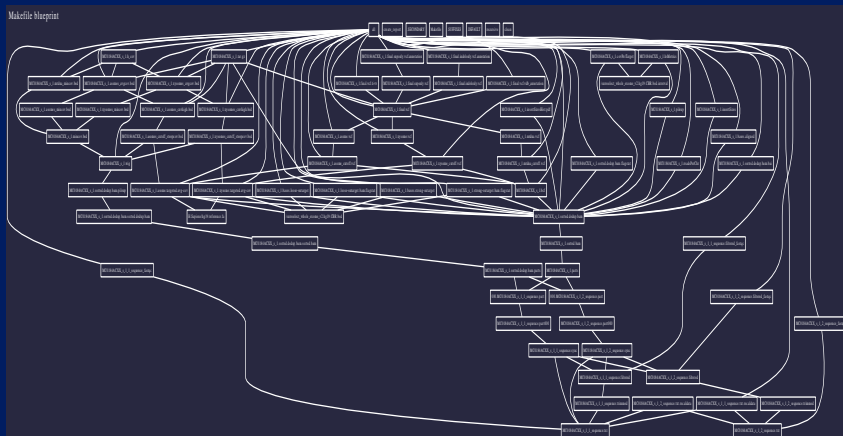


Figure 12 : Dependency diagram.

## Graphical interfaces

### *Workflow of a parallel pipeline*

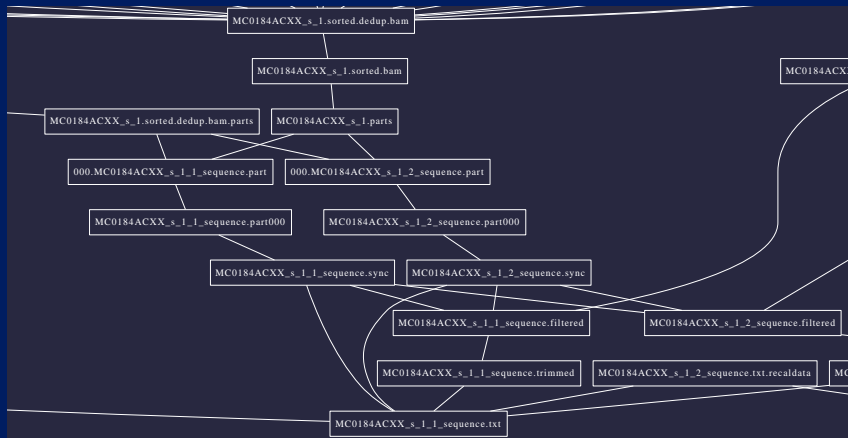


Figure 13 : Zoomed in.





Michiel van Galen  
Jeroen Laros

<https://humgenprojects.lumc.nl/trac/humgenprojects/wiki/NGS-intro>