GHIF Practical guidelines for quality control of WGS results in population-scale initiatives

Metric definitions

In this document, the GHIF WGS QC workgroup intends to identify a set of key QC metrics and spell out their detailed definitions. Whilst doing so, we expect to encounter recurrent information fields that apply to many metrics. Those can then be used as the basis for standardised guidelines for reporting QC metrics.

#### General notes

* The document is just a first draft to capture conversations from our regular meetings. It provides some templated sections as a guide, but those may not be exhaustive. As such, feel free to make as many changes as needed; we can always recover previous versions of the document using the file history.
* In terms of scope, the workgroup has agreed to focus on *germline WGS QC* first. While all of the workgroup participants are working with *short-read data* at the moment, we wish to make the definitions general enough to be applicable to other technologies as well. The workgroup also acknowledges that there are multiple stages in the analysis pipeline at which one may want to perform QC (e.g. post-FASTQ generation, post-alignment, post-variant calling). For the first iteration of the guidelines, the workgroup has agreed to focus on metrics that can be obtained from a *BAM/CRAM file*. Thus, metrics such as contamination or variant counts remain out of scope at the moment.

#### Controlled vocabulary

This section lists several example metrics in an attempt to capture which fields would be required to accurately describe how each metric has been calculated. When defining each metric, we attempt to align to the following general template:

* Id (mandatory): metric\_id
* Description (mandatory): Metric description.
* Source (mandatory): Tool and version used to calculate the metric.
* Implementation details (optional, depending on metric): Standardised insights into the metric implementation, where possible.

##### Example metric: Bases ≥ Q30

* Id: yield\_bp\_q30
* Description: The number of bases with BQ ≥ 30.
* Source: DRAGEN 3.7.6
* Implementation details:

| **Field** | **Description** | **Format** | **Value in example implementation** |
| --- | --- | --- | --- |
| MIN\_BQ | Minimum base quality | Integer | 30 |
| MIN\_MQ | Minimum mapping quality | Integer | 0 |
| DUP | Are reads marked as duplicates included? | Boolean | FALSE |
| CLP | Are clipped bases (hard and soft clipped) included? | Boolean | FALSE |
| OLP | Are overlapping bases included? | Boolean | FALSE |
| UMI | Are Unique Molecular Identifiers used to collapse reads? | Boolean | FALSE |
| SEC | Are secondary alignments included? | Boolean | FALSE |

##### Example metric: Mean autosome coverage

* Id: mean\_autosome\_coverage
* Description: The mean coverage in autosomes.
* Source: in-house tool based on mosdepth v0.3.2
* Implementation details:

| **Field** | **Description** | **Format** | **Value in example implementation** |
| --- | --- | --- | --- |
| REF | Genomic reference build | String | GRCh38 |
| BED | Genomic filtering regions | String | Homo\_sapiens\_assembly38.autosomes.bed |
| MIN\_BQ | Minimum base quality | Integer | 0 |
| MIN\_MQ | Minimum mapping quality | Integer | 20 |
| DUP | Are duplicates included? | Boolean | FALSE |
| CLP | Are clipped bases (hard and soft clipped) included? | Boolean | FALSE |
| OLP | Are overlapping bases included? | Boolean | FALSE |
| UMI | Are Unique Molecular Identifiers used to collapse reads? | Boolean | FALSE |
| SEC | Are secondary alignments included? | Boolean | FALSE |

##### Example metric: Percent autosomes covered ≥ 15 X

* Id: pct\_autosomes\_15x
* Description: The percentage of bases that attained at least 15X sequence coverage in autosomes.
* Source: in-house tool based on mosdepth v0.3.2
* Implementation details: same as mean\_autosome\_coverage.

##### Example metric: Genome coverage uniformity

* Id: autosome\_coverage\_uniformity
* Description: The percentage of bases with more *or less* than 25% coverage difference from the mean autosome coverage
* Source: in-house tool based on mosdepth v0.3.2 calculating (PCT < 0.75 \* mean\_autosome\_coverage) + (PCT > 1.25 \* mean\_autosome\_coverage)
* Implementation details: same as mean\_autosome\_coverage.

Example metric: Read mapping quality

* Id (mandatory): read\_mapping\_quality
* Description (mandatory): The percentage of reads mappable to the REF sequence with MAPQ>0
* Source (mandatory): Via samtools (reads\_mapped\_percent)
* Implementation details: same as mean\_autosome\_coverage.

Example metric: Discordant read pairs

* Id (mandatory): discordant\_read\_pairs
* Description (mandatory): The percentage of properly paired reads after alignment
* Source (mandatory): Via samtools (reads\_properly\_paired\_percent)
* Implementation details: same as mean\_autosome\_coverage.

Example metric: Mean insert size

* Id (mandatory): mean\_insert\_size
* Description (mandatory): A tuple of mean insert size for paired and mapped reads followed by the insert size standard deviation for the average template length distribution
* Source (mandatory): Via samtools (Insert\_size\_average and insert\_size\_standard\_deviation)
* Implementation details: same as mean\_autosome\_coverage.

#### Example implementations

##### calculate\_coverage.py

A mosdepth wrapper to calculate average coverage and genome completeness metrics.

Documentation: <https://bit.ly/3EcgZXN>