

Short-read sequencing tools for diagnostics

Marc Sturm^{1*}, Christopher Schroeder¹, Jakob Matthes¹, Stephan Ossowski¹

¹ Institute of Medical Genetics and Applied Genomics, University of Tuebingen, Germany.

*marc.sturm@med.uni-tuebingen.de

Abstract

Over the last few years, next-generation-sequencing (NGS) continuously became more affordable and the data analysis tools for NGS became more and more mature. Thus, many molecular biology technologies traditionally used in medical genetics (e.g. Sanger sequencing and genome arrays) are widely replaced by NGS. There are well-established standard tools for the main analysis steps, e.g. BWA for mapping and GATK for variant calling of short-read DNA sequencing, that are readily usable in diagnostics. However, many of the auxiliary tools needed for a diagnostics NGS pipeline, e.g. those for quality control, are written rather for research than for diagnostics and are not as mature as they should be for diagnostics. They often lack in speed, documentation and/or ease-of-use. Additionally, they are distributed over many different projects and are based on different programming languages, which makes installation and regular updates very time-consuming.

To fill this gap, we developed ngs-bits, a bundle of short-read sequencing tools that focus on the not so prominent steps in a diagnostic short-read NGS pipeline. ngs-bits offers tools for adapter trimming, UPD/ROH calling, quality control and many more auxiliary tools. All ngs-bits tools are written in C++ using the Qt framework and the htlib library. They are platform-independent and the source code is freely available under the MIT license on GitHub (<https://github.com/imgag/ngs-bits>). Binaries for Linux/Mac are available through Bioconda.

Quality control

Technical quality control is a crucial step when using NGS in clinical diagnostics. ngs-bits can perform QC for individual samples using all three layers of information: Unprocessed raw data (*ReadQC*), mapped reads (*MappingQC*) and variant list (*VariantQC*) are used to compute a comprehensive list of quality metrics and plots.

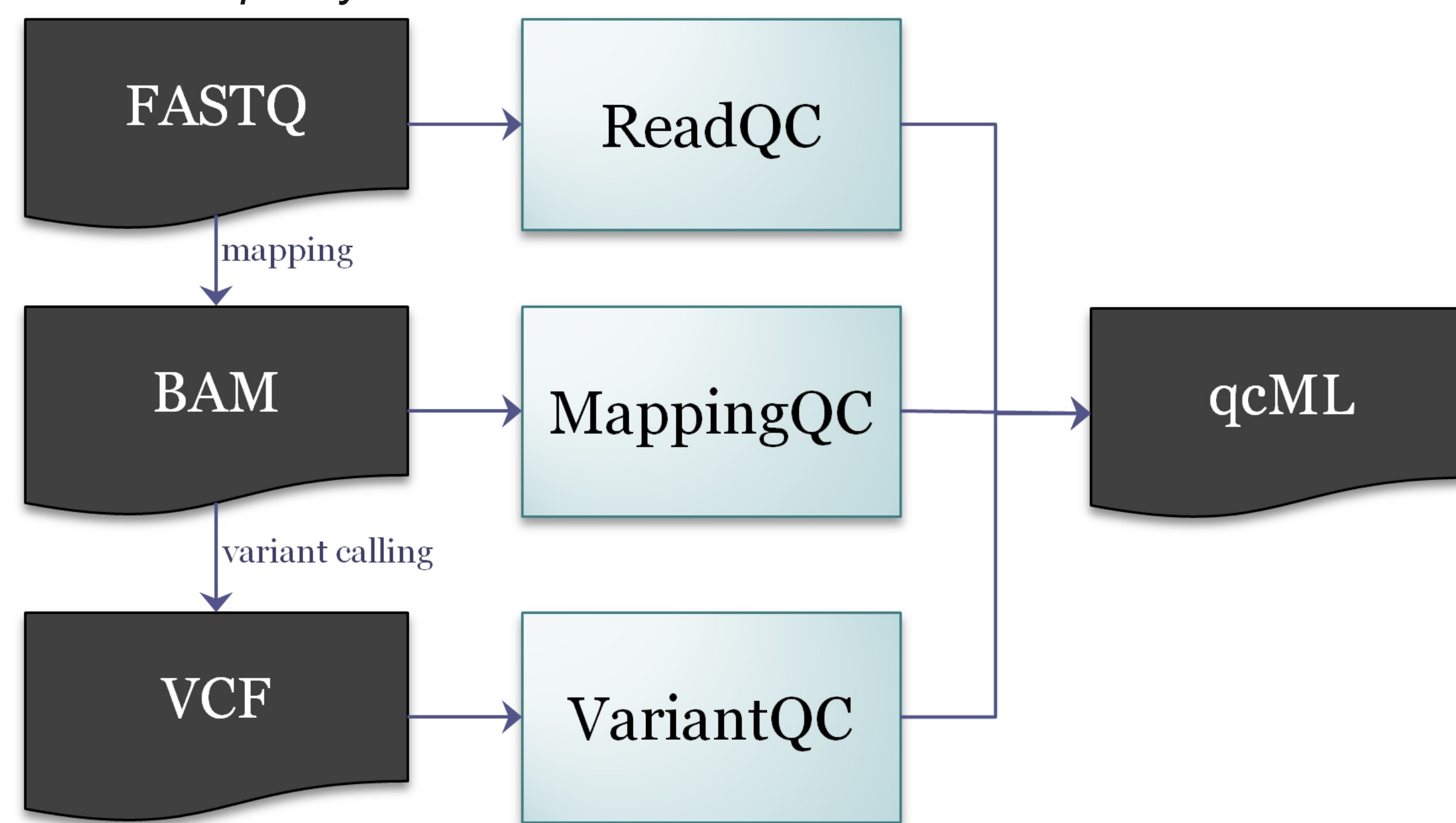
Our adapter trimmer (*SeqPurge*) can also perform raw read QC, which speeds up the overall analysis by avoiding a second processing of the complete raw data.

Quality control of tumor-normal pairs is supported using the *SomaticQC* tool.

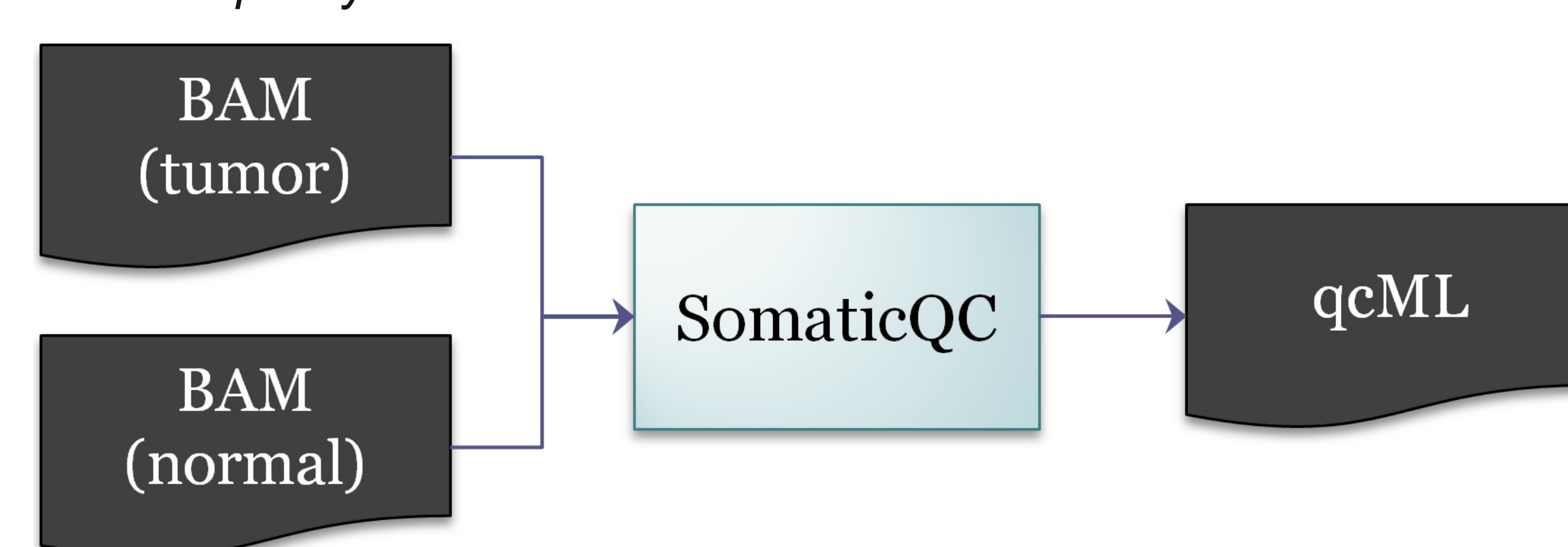
qcML format:

The quality metrics computed by ngs-bits tools are stored in qcML format, a standard QC format developed for high-throughput proteomics. We added quality terms for NGS to the qcML ontology, which makes the format readily usable for genomics. qcML files are easy to process computationally (XML) and can also be opened in a web browser for visual inspection (via an embedded style sheet). We are currently implementing MultiQC support.

Germline quality control:



Somatic quality control:



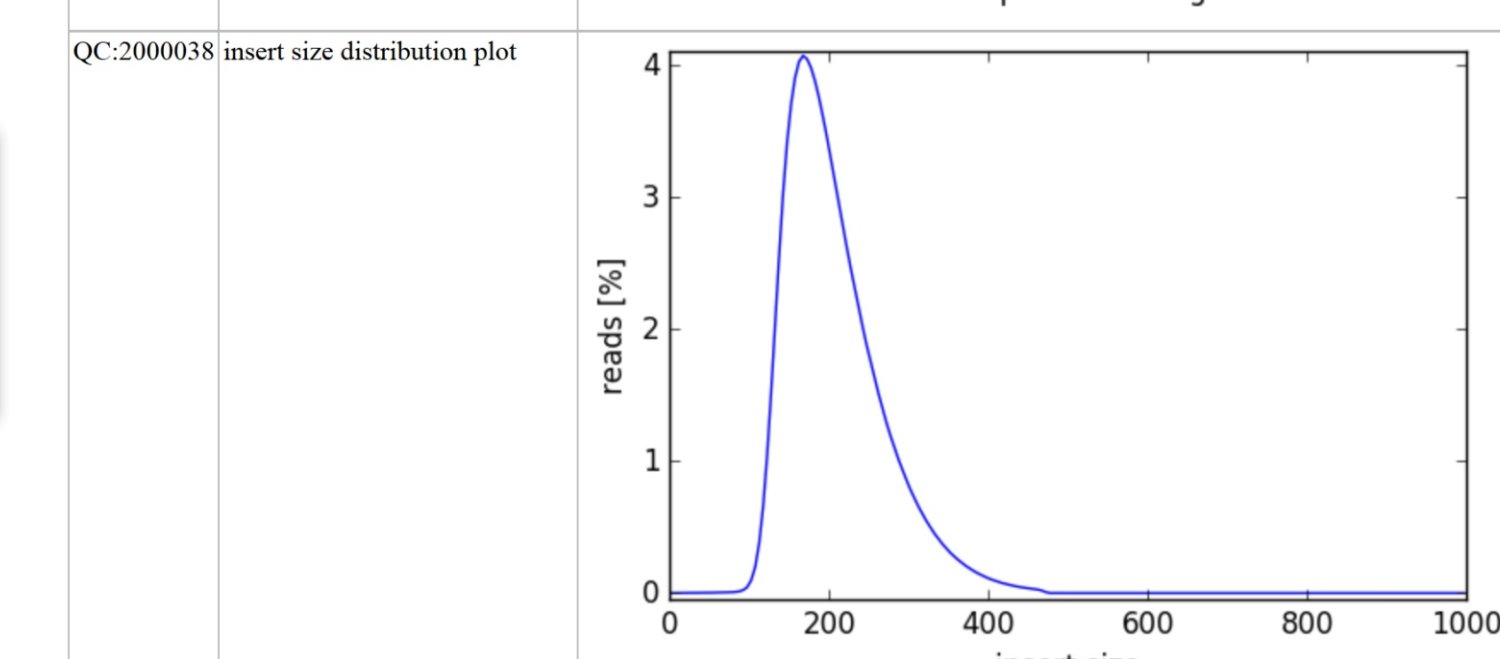
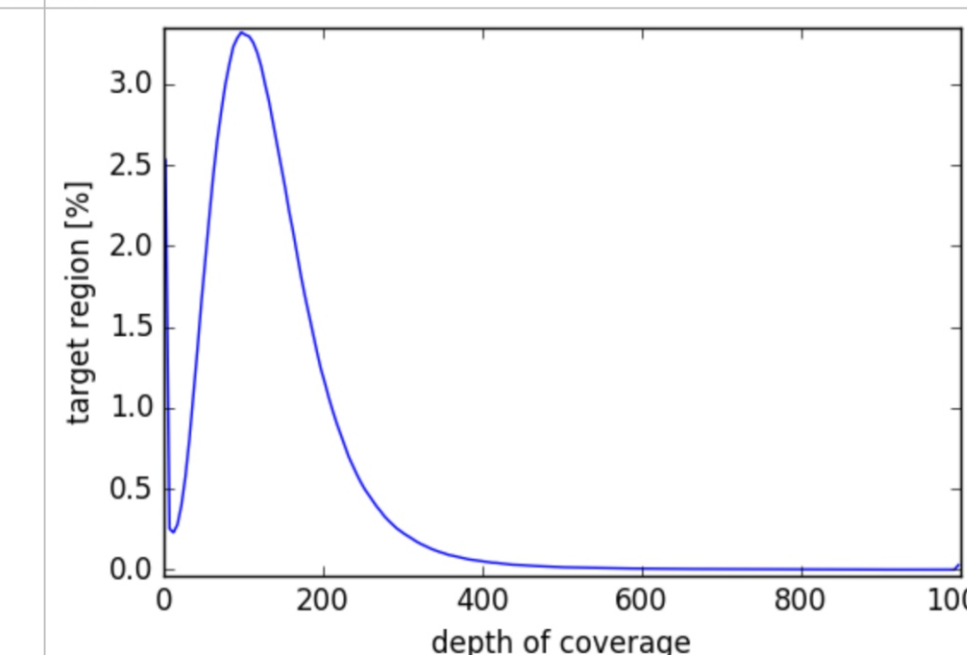
Example qcML (rendered in browser):

Meta data:

Accession	Name	Value
QC:1000002	creation software	MappingQC 2018_06-30-g7704dc
QC:1000003	creation software parameters	-vsi -s4EAE-7 2018_06_06.bed
QC:1000004	creation date	2018-08-31T03:41:19
QC:1000005	source file	DX183751_01.bam

Quality parameters:

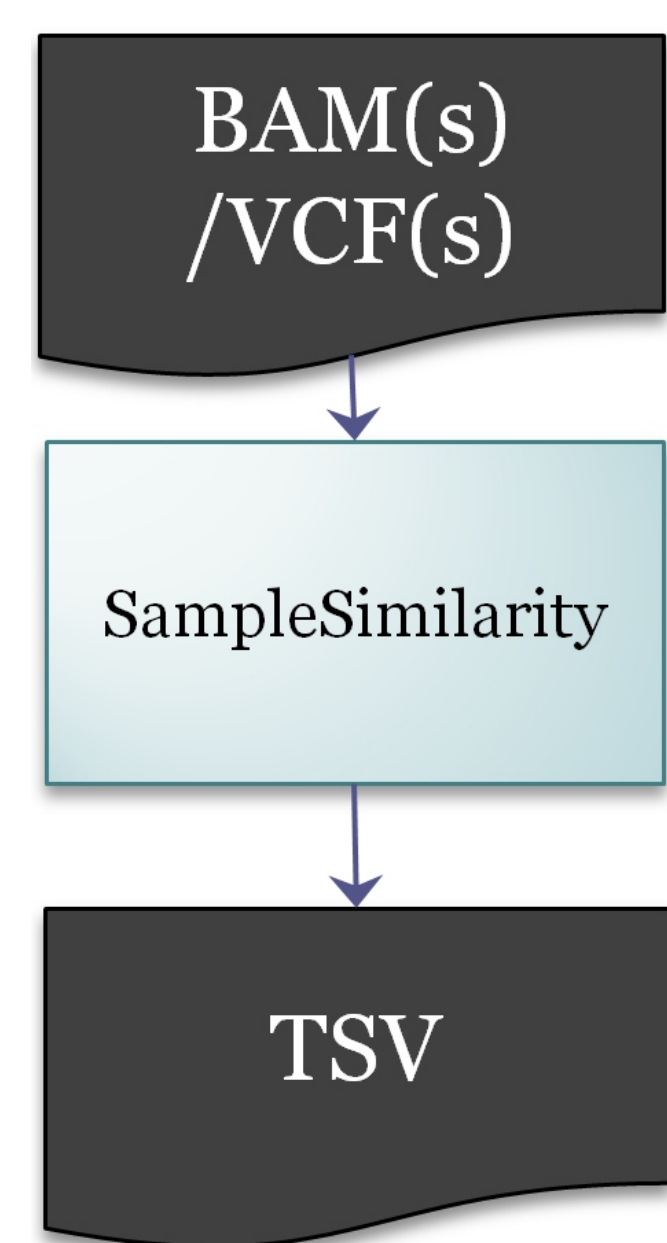
Accession	Name	Value
QC:2000019	trimmed base percentage	0.11
QC:2000052	clipped base percentage	0.42
QC:2000020	mapped read percentage	99.51
QC:2000021	on-target read percentage	68.20
QC:2000022	properly-paired read percentage	95.55
QC:2000023	insert size	206.57
QC:2000024	duplicate read percentage	27.46
QC:2000050	bases usable (MFI)	5899.02
QC:2000025	target region read depth	132.77
QC:2000026	target region 10x percentage	97.21
QC:2000027	target region 20x percentage	96.70
QC:2000028	target region 30x percentage	95.73
QC:2000029	target region 50x percentage	90.83
QC:2000030	target region 100x percentage	62.84
QC:2000031	target region 200x percentage	15.15
QC:2000032	target region 500x percentage	0.49
QC:2000051	SNV allele frequency deviation	1.42
QC:2000037	depth distribution plot	



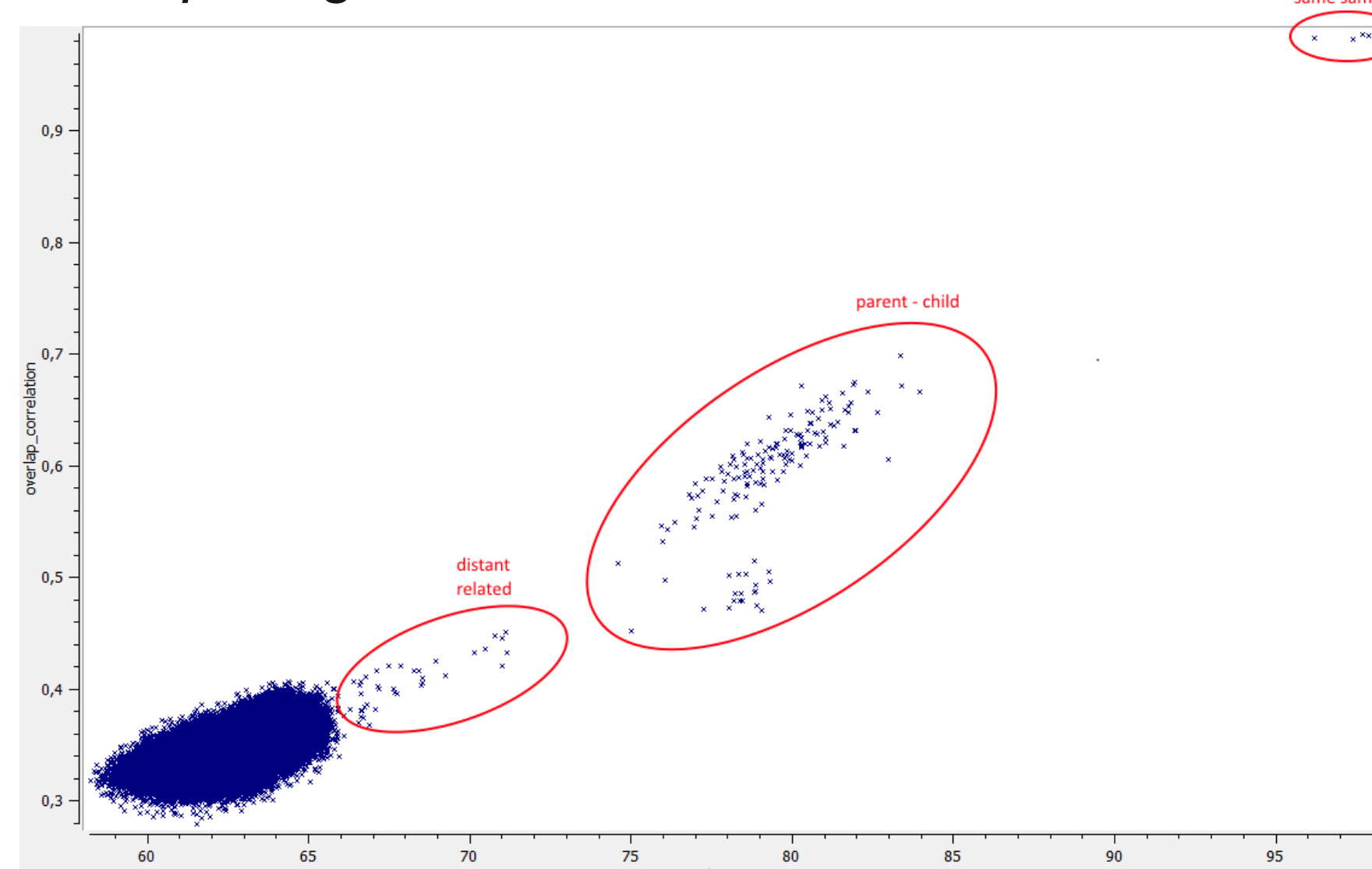
Sample similarity

Sample similarity is used to check data consistency when performing multi-sample analyses (trio, family, tumor-normal). *SampleSimilarity* calculates several metrics:

- overlap: Percentage of variants that occur in both samples
- correlation: Correlation of variant genotypes
- ibs0: Percentage of variants with zero IBS, e.g. AA and CC
- ibs2: Percentage of variants with complete IBS, e.g. AA and AA



Example Agilent V6 exomes:



Sample gender

Checking gender alone can already identify 50% of sample swaps. The *SampleGender* tool implements several methods:

- coverage of the SRY gene (genome/exome/panel)
- percentage of heterozygous SNPs on chrX (exome/panel without SRY gene)
- ratio of reads mapped to chrX / chrY (shallow genome)



Sample ancestry

Sample ancestry can be important as consistency check and for upstream interpretation of data (e.g. Africans have more variants). The *SampleAncestry* tool assigns one of the four main populations (AFR, EUR, SAS, EAS).



Other tools

The most notable ngs-bits tools not shown in detail on this poster are:

SeqPurge	A highly-sensitive adapter trimmer for paired-end short-read data.
UpdHunter	Uniparental disomy (UPD) detection. Input is a multi-sample VCF file of a trio.
RohHunter	Runs of homozygosity (ROH) detection. Input is a VCF file annotated with allele frequency values.

A full list of the more than 50 ngs-bits tools can be found on our website. Auxiliary operations on many NGS standard formats are supported:

BED	merging, intersecting, statistics, coverage calculations, annotation.
FASTQ	checks, conversions, trimming, unique molecular barcode handling.
BAM	quality filtering, overlap clipping, downsampling, conversions.
VCF	sorting, normalizing, filtering by regions, annotation.

Availability

ngs-bits is implemented using C++/htlib/Qt and is freely available under the MIT license from GitHub at: <https://github.com/imgag/ngs-bits>

Platform-independent

ngs-bits can be built from sources under Linux, macOS and Windows. Binaries for Linux and macOS and a Docker container are available via Bioconda:

BIOCONDA

\$ conda install ngs-bits

Genomes

ngs-bits supports both hg19/GRCh37 and hg38/GRCh38.



Scan me