华大智造
MGI

MegaBOLT 生信分析加速器

用户手册

地　址：中国深圳市盐田区北山工业区综合楼及11栋2楼
电　话：4000-966-988
邮　箱：MGI-service@mgi-tech.com
网　址：www.mgi-tech.com

仅供科研使用

深圳华大智造科技股份有限公司

版本 5.0

关于本手册

本手册适用于 MegaBOLT 生信分析加速器（MegaBOLT_scheduler），手册版本 5.0，软件版本 V2.4.0.0。

制造商信息

生产厂家	深圳华大智造科技股份有限公司
生产地址	中国深圳市盐田区北山工业区综合楼及 11 栋 2 楼
技术支持厂家	深圳华大智造科技股份有限公司
技术支持联系电话	4000-966-988
技术支持联系邮箱	MGI-service@mgi-tech.com

版本记录

	日期	版本
修订	2022 年 6 月 1 日	5.0
	2021 年 7 月 1 日	4.0
	2020 年 12 月 30 日	3.0
	2020 年 7 月 17 日	2.0
编制	2019 年 11 月 30 日	1.0

目录

1 简介

MegaBOLT 是一套高性能的重测序分析系统，包含胚系突变（Germline）与体细胞突变（Somatic）的全基因组（WGS）、全外显子组（WES）及 Panel 靶向测序数据分析，完成从测序序列文件 *fq.gz* 输入至变异检测结果 *vcf.gz* 输出的计算，包含了前处理（QC）、以及后处理（BAM 文件和 VCF 文件统计）。通过 FPGA（即现场可编程门阵列）硬件加速卡及多任务调度系统进行计算加速，与 CPU 常规流程相比（以 GATK Best Practice 为例），可加速 10~20 倍。

MegaBOLT 机架服务器模式为搭载 MegaBOLT 分析系统的机架式服务器，适用于集群环境的大规模数据分析场景。支持多种分析流程，操作灵活，适用于有一定生物信息分析背景的用户。

MegaBOLT 工作站模式为搭载 MegaBOLT 分析系统的小型工作站，适用于中小型数据分析场景。能够提供从测序仪测序、数据下机到 WGS/WES 分析的一站式分析服务。提供网页交互式操作界面和分析报告，操作简单，适用于广大非生物信息分析背景的用户。

--- 此页有意留白 ---

2 系统描述

两种运行模式

运行模式	说明
WGS	全基因组数据分析
WES	全外显子数据分析

多种流程选择

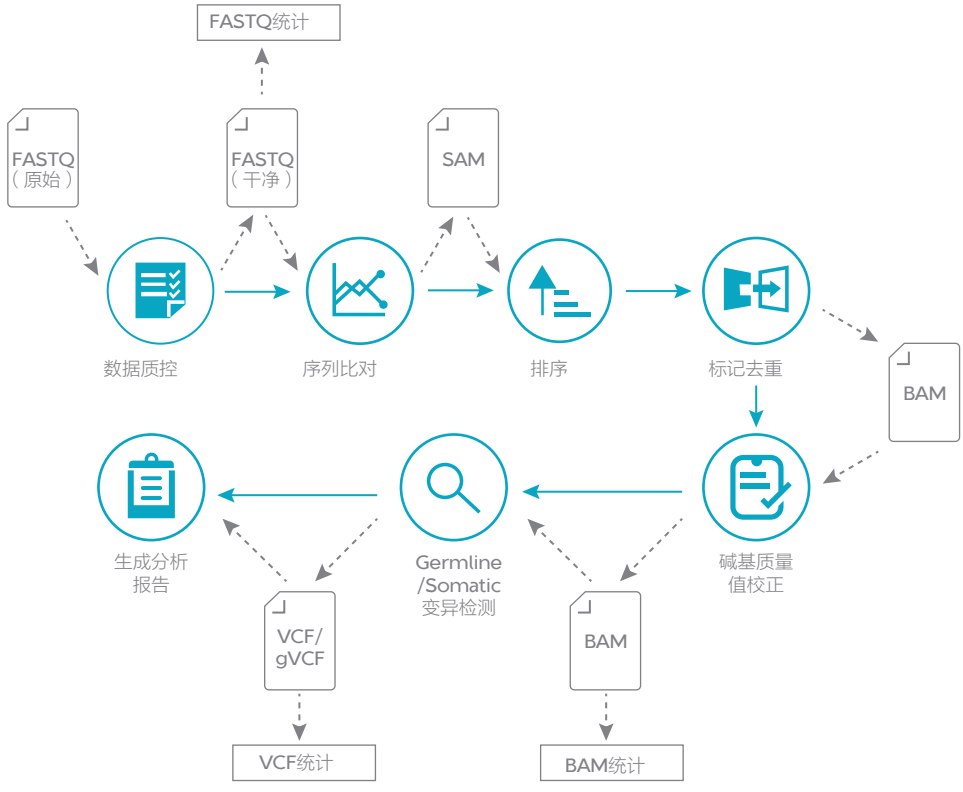

图 1 流程图

注意 带 * 标注的为可选流程或输入输出。

主要流程		说明
basic（默认流程）	胚系（Germline）变异检测基础流程	依次完成序列比对、排序、标记去重 *、碱基质量值再校准 * 和 Germline 变异检测流程。 输入数据为干净 FASTQ 文件。 输出结果为碱基质量值再校准后的 BAM 文件 * 和变异检测结果 VCF/gVCF 文件。
full	胚系（Germline）变异检测全流程	依次完成数据质控、序列比对、排序、标记去重 *、碱基质量值再校准 *、Germline 变异检测、BAM/VCF 统计和生成分析报告流程。 输入数据为原始 FASTQ 文件。 输出结果为干净 FASTQ 文件 *、碱基质量值再校准后的 BAM 文件 *、变异检测结果 VCF/gVCF 文件、FASTQ/BAM/VCF 统计结果和分析报告。
somatic	体细胞（Somatic）变异检测流程	依次完成数据质控 *、序列比对、排序、标记去重 *、碱基质量值再校准 *、Germline 变异检测 *、Somatic 变异检测、BAM/VCF 统计 * 和生成分析报告 * 流程。 输入数据为肿瘤、对照样本 * 的原始 / 干净 FASTQ 文件。 输出结果为干净 FASTQ 文件 *、碱基质量值再校准后的 BAM 文件 *、Germline 变异检测结果 VCF/gVCF 文件 *、Somatic 变异检测结果 VCF 文件、FASTQ/BAM/VCF 统计结果 * 和 Germline 变异分析报告 *。

MegaBOLT 还支持更多、更灵活的流程选择，详细说明参见*第 13 页“流程简介（--type）”*。

配置要求

项目	最低配置	推荐配置
CPU	2 × Intel Xeon E5-26XX Series	2 × Intel Xeon Gold 62XX Series
内存	96 GB	128 GB
硬盘	1 TB HDD 硬盘	>2 TB SSD 硬盘
系统	CentOS 7.3-7.5	CentOS 7.3-7.5
网卡	千兆网卡	万兆网卡

--- 此页有意留白 ---

3 快速开始

脚本示例

编写运行脚本 *run.sh* 如下：

WGS Germline 基础流程分析：

```
MegaBOLT --type basic --runtype WGS --list sample.list
```

WGS Germline 全流程分析：

```
MegaBOLT --type full --runtype WGS --list sample.list
```

WGS Somatic 流程分析：

```
MegaBOLT --type somatic --runtype WGS --list sample.tumor.list
--list2 sample.normal.list
```

WES Germline 基础流程分析：

```
MegaBOLT --type basic --runtype WES --list sample.list --bed
BV4
```

WES Germline 全流程分析：

```
MegaBOLT --type full --runtype WES --list sample.list --bed
BV4
```

WES Somatic 流程分析：

```
MegaBOLT --type somatic --runtype WES --list sample.tumor.list
--list2 sample.normal.list --bed BV4
```

构建 Index:

```
MegaBOLT --type buildindex --ref ref.fa --vcf dbsnp.vcf.gz
--knownSites indels1.vcf.gz --knownSites indels2.vcf.gz
```

注意
- MegaBOLT 支持的其他流程参见*第 13 页“流程简介（--type）”*。
- 更多使用示例参见*第 1 页“简介”*。

示例说明

下面以“WGS 基础流程分析”为例进行说明：

```
MegaBOLT --type basic --runtype WGS --list sample.list
```

其中“MegaBOLT”为运行程序名，“--type basic”指明运行流程为basic流程，“--runtype WGS”指明运行模式为 WGS，“--list sample.list”指明分析样本列表（list 文件）。

PE 数据 list 文件基本格式如下：

```
SampleName Read1 Read2 Adaptor1 Adaptor2 RGID RGSM RGLB RGPL
```

关于 list 文件格式的详细说明，请参见*第 9 页“List 文件”*。

此脚本隐含若干默认参数，例如脚本未指定参考序列（通过 --ref 参数），因此将使用系统默认的hg19.fa进行分析。“basic”和“WGS”也分别为“--type”和“--runtype”参数的默认值，因此上脚本可简化为：

```
MegaBOLT --list sample.list
```

详细参数说明请参见*第 21 页“参数说明”*，或通过命令“MegaBOLT -h”查看。

4 List 文件

List 文件是输入 FASTQ 文件的组织形式，支持 PE 数据和 SE 数据。

PE 数据 list 文件格式

格式 1

```
SampleName    Read1    Read2
```

说明：

- “SampleName”是样本名，“Read1”为 PE 数据 Read1 的 FASTQ 文件路径，“Read2”为 PE 数据 Read2 的 FASTQ 文件路径，字段与字段之间用空格符或制表符分隔。

  示例：

  ```
  sample    /data/example/read1.fq.gz    /data/example/
  read2.fq.gz
  ```

- 支持单个 list 文件包含多个样本，每个样本占一行，允许“#”注释行。

  示例：

  ```
  sample1    /data/example/read1_1.fq.gz    /data/
  example/read2_1.fq.gz
  sample2    /data/example/read1_2.fq.gz    /data/
  example/read2_2.fq.gz
  ```

- 支持单个样本包含多对 FASTQ 文件，多个 FASTQ 文件以“,”分隔，注意不要有空格。

  示例：

  ```
  sample    read1_1,read1_2,…,read1_N
  read2_1,read2_2,…,read2_N
  ```

格式 2

```
SampleName    Read1    Read2    Adaptor1    Adaptor2
```

说明：

- “Adaptor1”是 Read1 的测序接头序列，“Adaptor2”是 Read2 的测序接头序列。
- “Adaptor1”和“Adaptor2”应当成对出现，并且必须是仅包含“ATGC”的字符序列。
- 一个样本仅能包含一组接头序列。

示例：

```
sample    read1_1,read1_2    read2_1,read2_2    AAGTCGGA
AAGTCGGATC
```

格式 3

SampleName Read1 Read2 RGID RGSM RGLB RGPL

说明：

- “RGID”、“RGSM”、“RGLB”和“RGPL”分别为“Read 分组 ID 编号”、“Read 分组样本名”、“Read 分组文库名”和“Read 分组测序平台”。
- “RGID”、“RGSM”、“RGLB”和“RGPL”应当成组出现，并且不能改变顺序。
- “RGSM”应当与“SampleName”保持一致。
- “RGPL”仅支持如下字段：

  MGISEQ、BGISEQ、ILLUMINA、SLX、SOLEXA、SOLID、454、LS454、COMPLETE、PACBIO、IONTORRENT

- 当单个样本包含多对 FASTQ 文件时，允许对每个 FASTQ 文件的“RGID”、“RGSM”、“RGLB”和“RGPL”进行设置，以“,”分隔，但应保证列数与 Read 对数目一致。

示例：

```
sample   read1_1,read1_2   read2_1,read2_2   id1,id2  sample
lb  COMPLETE
```

格式 4

SampleName Read1 Read2 Adaptor1 Adaptor2 RGID RGSM RGLB RGPL

说明：

- “Adaptor1 Adaptor2”和“RGID RGSM RGLB RGPL”不可以交换顺序。
- 其他约束同格式 1-3。

示例：

```
sample  read1  read2  AAGTCGGA  AAGTCGGATC  id  sample  lb
COMPLETE
```

默认值说明

List 文件中未指定的列将使用默认值。

格式	说明
Adaptor1	AAGTCGGAGGCCAAGCGGTCTTAGGAAGACAA
Adaptor2	AAGTCGGATCGTAGCCATGTCGTTCTGTGAGCCAAGGAGTTG
RGID	rg
RGSM	list 文件中的“SampleName”
RGLB	lb
RGPL	COMPLETE

SE 数据 list 文件格式

说明：SE 数据 list 文件格式约束条件与 PE 数据相同。

格式 1

SampleName Read

格式 2

SampleName Read Adaptor

格式 3

SampleName Read RGID RGSM RGLB RGPL

格式 4

```
SampleName      Read      Adaptor      RGID      RGSM      RGLB
RGPL
```

5 流程简介（--type）

MegaBOLT 支持运行多种流程，包括单模块流程和组合流程，通过“--type”参数进行选择。

组合流程

MegaBOLT 提供了生物信息分析中经常用到的组合流程，并采用软硬件结合的方式进行优化。与单模块顺序执行相比，在保证结果的一致的前提下，能以更快的速度完成任务分析。因此我们推荐用户使用组合流程。

支持的组合流程如下：

注意 带 * 标注的为可选流程或输入输出。

组合流程		说明
basic（默认流程）	胚系（Germline）变异检测基础流程	依次完成序列比对、排序、标记去重 *、碱基质量值再校准 * 和 Germline 变异检测流程。 输入数据为干净 FASTQ 文件。 输出结果为碱基质量值再校准后的 BAM 文件 * 和变异检测结果 VCF/gVCF 文件。
full	胚系（Germline）变异检测全流程	依次完成数据质控、序列比对、排序、标记去重 *、碱基质量值再校准 *、Germline 变异检测、BAM/VCF 统计和生成分析报告流程。 输入数据为原始 FASTQ 文件。 输出结果为干净 FASTQ 文件 *、碱基质量值再校准后的 BAM 文件 *、变异检测结果 VCF/gVCF 文件、FASTQ/BAM/VCF 统计结果和分析报告。

组合流程		说明
somatic	体细胞（Somatic）变异检测流程	依次完成数据质控 *、序列比对、排序、标记去重 *、碱基质量值再校准 *、Germline 变异检测 *、Somatic 变异检测、BAM/VCF 统计 * 和生成 Germline 变异分析报告 * 流程。 输入数据为肿瘤、对照样本 * 的原始 / 干净 FASTQ 文件。 输出结果为干净 FASTQ 文件 *、碱基质量值再校准后的 BAM 文件 *、Germline 变异检测结果 VCF/gVCF 文件 *、Somatic 变异检测结果 VCF 文件、FASTQ/BAM/VCF 统计结果 * 和分析报告 *。 注意 ● 支持肿瘤 / 对照模式和单肿瘤模式； ● 可通过参数设置是否运行数据质控、标记去重、碱基质量值再校准、Germline 变异检测、BAM/VCF 统计和生成分析报告流程；（运行数据质控、Germline 变异检测和 BAM/VCF 统计才会生成 Germline 变异分析报告）； ● 肿瘤样本通过参数“--list”设置，对照样本通过参数“--list2”设置。
alignmentsortmarkdup	序列比对、排序和标记去重	依次完成序列比对、排序、标记去重 * 流程。 输入数据为干净 FASTQ 文件。 输出结果为排序、标记去重后的 BAM 文件。

组合流程		说明
alignmentsortmarkdupbqsr	序列比对、排序、标记去重和碱基质量值再校准	依次完成序列比对、排序、标记去重 * 和碱基质量值再校准流程。 输入数据为干净 FASTQ 文件。 输出结果为碱基质量值再校准后的 BAM 文件。
extract	胚系（Germline）变异检测提取流程	依次完成数据质控、序列比对、排序、标记去重 *、碱基质量值再校准 *、提取乘数、标记去重 *、Germline 变异检测、BAM/VCF 统计和生成分析报告流程。 输入数据为原始 FASTQ 文件。 输出结果为干净 FASTQ 文件 *、提取乘数后的碱基质量值再校准后的 BAM 文件 *、变异检测结果 VCF/gVCF 文件、FASTQ/BAM/VCF 统计结果和分析报告。
buildindex	构建 MegaBOLT 流程所需的各种索引文件	构建 MegaBOLT 流程所需的与参考基因组序列（ref）相关的各种索引文件。 输入数据为 ref 文件（*.fa）和已知 SNP/INDEL 数据库文件（*.vcf.gz）。 输出数据为 ref 的 dict 索引文件，fai 索引文件，effective bed 文件，BQSR 索引文件（*.vcfi）和 BWA 索引文件。 注意 索引文件会输出到 ref 文件所在目录，需要确认该目录可写，并且不存在该 ref 的 dict 索引文件，fai 索引文件，effective bed 文件，BQSR 索引文件（*.vcfi）和 BWA 索引文件。

组合流程支持通过参数设置流程，以满足用户个性化需求：

- 可通过参数选择比对软件：Minimap2（默认）或 BWA；

- 可通过参数选择变异检测软件：HaplotypeCaller 3.8（默认）、HaplotypeCaller 4.0 或 DeepVariant；
- 可通过参数设置是否运行标记去重和碱基质量值再校准流程；
- 可通过参数设置是否输出结果文件。

详细参数说明请参见*第 21 页“参数说明”*。

单模块流程

单模块流程		说明
qc	原始FASTQ文件质量控制	完成对原始 FASTQ 文件的过滤和统计。 输入数据为原始 FASTQ 文件。 输出结果为干净 FASTQ 文件和统计结果文件。
alignment	序列比对	将测序的序列比对定位到参考基因组上。 输入数据为干净 FASTQ 文件。 输出结果为干净 FASTQ 文件和统计结果文件。 注意 支持选择比对软件：Minimap2（默认）或 BWA。 BWA 软件使用加速版本，会在参考序列目录下构建一个较大的索引文件。
sortmarkdup	排序、标记去重	将比对好的序列按基因组上的位置进行排序，将比对到相同位置的序列标记为重复。 输入数据为比对后 SAM/BAM 文件。 输出数据为排序、标记去重后 BAM 文件。 注意 可通过参数选择是否运行标记去重。
bqsr	碱基质量值再校准	对比对结果数据中测序质量值进行数据内部的校准。 输入数据为 BAM 文件。 输出数据为碱基质量值再校准后的 BAM 文件。

单模块流程		说明
haplotypecaller	胚系（Germline）变异检测	利用已经比对到参考基因组上的基因序列及相关信息识别检测变异。 输入数据为 BAM 文件。 输出数据为变异检测结果 VCF/gVCF 文件。 注意 支持选择变异检测软件：GATK HaplotypeCaller 3.8（默认）、GATK HaplotypeCaller 4.0或DeepVariant。
mutect2	体细胞（Somatic）变异检测	利用已经比对到参考基因组上的基因序列及相关信息识别检测体细胞变异。 输入数据为 BAM 文件。 输出数据为变异检测结果 VCF 文件。
genotypegvcfs	联合基因分型	对变异检测生成的 gVCF 文件进行联合基因分型。 输入数据为 gVCF 文件。 输出数据为联合基因分型后的 VCF 文件。
vqsr	变异质量值再校准	对变异检测结果中变异质量值进行数据内部再校准。 输入数据为 VCF 文件。 输出数据为变异质量值再校准后的VCF文件。
filtration	变异过滤	根据 INFO 或 FORMAT 注释对变异检测数据进行过滤。 输入数据为 VCF 文件。 输出数据为过滤后的 VCF 文件。
rtgtools	变异检测结果评估	基于变异标准集对 VCF 文件中的 SNP/INDEL 变异位点的假阳性、假阴性、准确度和灵敏度进行评价。 输入数据为 VCF 文件。 输出数据为 VCF 文件中 SNP/INDEL 变异位点的假阳性、假阴性、准确度和灵敏度评价值。
builddict	构建dict索引	构建参考基因组序列（ref）的 dict 索引文件。 输入数据为 ref 文件（*.fa）。 输出数据为 ref 的 dict 索引文件。 注意 索引文件会输出到ref文件所在目录，需要确认该目录可写，并且不存在该 ref 的 dict 索引文件。

单模块流程		说明
buildfai	构建 fai 索引	构建参考基因组序列（ref）的 fai 索引文件。 输入数据为 ref 文件（*.fa）。 输出数据为 ref 的 fai 索引文件。 注意 索引文件会输出到ref文件所在目录，需要确认该目录可写，并且不存在该 ref 的 fai 索引文件。
buildbed	构建 effective bed 文件	构建参考基因组序列（ref）的 effective bed 文件。 输入数据为 ref 文件（*.fa）。 输出数据为 ref 的 effective bed 文件。 注意 文件会输出到 ref 文件所在目录，需要确认该目录可写，并且不存在该 ref 的 effective bed 文件。
bwaindex	构建 bwa 索引	构建参考基因组序列（ref）的 bwa 索引文件。 输入数据为 ref 文件（*.fa）。 输出数据为 ref 的 bwa 索引文件。 注意 索引文件会输出到ref文件所在目录，需要确认该目录可写，并且不存在该 ref 的 bwa 索引文件。
bqsrindex	构建碱基质量值再校准索引	构建进行碱基质量值再校准（BQSR）时需要的索引文件。 输入数据为参考基因组序列文件（*.fa）和已知 SNP/INDEL 数据库文件（*.vcf.gz）。 输出数据为 BQSR 的索引文件（*.vcfi）。 注意 ● 索引文件默认会输出到 ref 文件所在目录，如果该目录不可写，则会输出到用户输出目录； ● 相同 ref 文件和数据库文件只需构建一次。
bamstats	BAM 统计	BAM 文件信息统计。 输入数据为 BAM 文件。 输出数据为 BAM 文件的统计信息。
vcfstats	VCF 统计	VCF 文件信息统计。 输入数据为 VCF 文件。 输出数据为 VCF 文件的统计信息。

单模块流程		说明
bamtocram	将 BAM 文件转换为 CRAM 文件	将 BAM 文件转换为 CRAM 文件。 输入数据为 BAM 文。 输出数据为 CRAM 文件。
WGP	CNV/SV 分析	CNV/SV 分析。 输入数据为 BAM 文件。 输出数据为 CNV/SV 分析结果文件。

--- 此页有意留白 ---

6 参数说明

程序：MegaBOLT
版本：V2.x.x
使用：`MegaBOLT [options]`

注意
- 数据量超过 SSD 容量，使用输出目录做缓存。
- 参数名严格区分大小写。

MegaBOLT 全局参数

参数	说明
`--type <string>`	选择运行流程。 可选流程：alignment、alignmentsortmarkdup、alignmentsortmarkdupbqsr、bamstats、bamtocram、basic、bqsr、bqsrindex、buildbed、builddict、buildfai、buildindex、bwaindex、extract、filtration、full、genotypegvcfs、haplotypecaller、mutect2、qc、rtgtools、somatic、sortmarkdup、vcfstats、vqsr（默认值：basic） 关于流程的详细说明参见*第 13 页“流程简介（--type）”*。
`--help | -h`	输出软件说明文档
`--version | -v`	输出软件版本
`--list <sample.list>`	list 文件是输入 FASTQ 文件的组织形式，支持 PE 数据和 SE 数据。 关于 list 文件格式说明参见*第 9 页“List 文件”*。

参数	说明
--ref <hg19.fa|hg19|hg38|hs37d5>	参考基因组序列文件 （默认值：hg19.fa）。 ● 支持使用内置参考文件集合：<hg19|hg38|hs37d5>; ● 当使用内置参考文件集合时，MegaBOLT将自动使用与指定 ref 配套的内置参考文件进行分析。此设置将覆盖其它参考文件设置，会被覆盖的参数有：--vcf，--knownSites，--resource-*，--rtg-*
--vcf <dbsnp.vcf>	dbSNP 文件（默认值：dbsnp_151.vcf.gz）。
--outputdir | -outdir <Path>	输出路径（默认值：当前目录）。
--outputprefix | -prefix <prefix>	输出文件前缀（默认值：output）。 注意 ● 仅当未输入 list 文件时有效； ● 当输入 list 文件时，以 list 文件中的 SampleName 作为输出文件前缀。
--runtype <WGS|WES>	运行模式（默认值：WGS）。 WGS：全基因组数据分析 WES：全外显子数据分析

参数	说明
--bed <BV4|BV5|AV2|AV5|AV6|ACV6|AV7|NV3|NME|TV1.2|IDT|AiJi|BV4-38|BV5-38|AV2-38|AV5-38|AV6-38|ACV6-38|AV7-38|NV3-38|NME-38|TV1.2-38|IDT- 38|AiJi-38|use r_defined_path>	hg19 对应的区间文件： BV4 MGI_Exome_V4_kit BV5 MGI_Exome_V5_kit AV2 Agilent_Exome_V2 AV5 Agilent_Exome_V5 AV6 Agilent_Exome_V6 ACV6 Agilent.V6COSMIC AV7 Agilent_Exome_V7 NV3 Roche_SeqCapEZ_Exome_v3.0 NME Nimblegen_MedExome_V2 TV1.2 Illumina.truseq.v1.2 IDT xgen_target AiJi AIJI_Exome hg38 对应的区间文件： BV4-38 MGI_Exome_V4_kit BV5-38 MGI_Exome_V5_kit AV2-38 Agilent_Exome_V2 AV5-38 Agilent_Exome_V5 AV6-38 Agilent_Exome_V6 ACV6-38 Agilent.V6COSMIC AV7-38 Agilent_Exome_V7 NV3-38 Roche_SeqCapEZ_Exome_v3.0 NME-38 Nimblegen_MedExome_V2 TV1.2-38 Illumina.truseq.v1.2 IDT-38 xgen_target AiJi-38 AIJI_Exome 用户自定义区间文件（建议使用绝对路径），如：*/home/my.bed* 注意 仅当运行模式为 WES 时有效。
--no-stats <0|1>	不运行 BAM/VCF 统计（默认值：0，即默认运行）。 注意 ● 不运行 QC，流程或不运行 Germline ● 变异检测流程时不会生成 Germline 变异检测报告。
--no-qc <0|1>	不运行 QC 流程（默认值：0，即默认运行）。

参数	说明
--no-markdup <0|1>	不运行标记去重（MarkDup）（默认值：0，即默认运行）。
--no-bqsr <0|1>	不运行碱基质量值再校准（BQSR）（默认值：0，即默认运行）。
--run-genotypegvcfs <0|1>	是否运行 GenotypeGVCFs（默认值：0，即默认不运行）
--run-vqsr <0|1>	是否运行 VQSR（默认值：0，即默认不运行）
--no-hc <0|1>	不运行 Germline 变异检测流程（默认值：0，即默认运行）。
--run-postalt	进行 alt 后处理步骤后，不输出 hla fastq 文件（默认值：0，即默认不输出）
--run-variant-filtration <0|1>	是否运行 Variant Filtration（默认值：0，即默认不运行）
--run-rtgtools <0|1>	是否运行 RTG Tools（默认值：0，即默认不运行）
--run-wgp <0|1>	是否运行 WGP（默认值：0，即默认不运行） ● 仅支持 WGS 运行模式； ● 使用 BWA 作为比对软件；
--no-hla-output	进行 alt 后处理步骤后，不输出 hla fastq 文件，仅适用于含 postalt 的流程（默认值：1，即默认不输出）
--no-fastq-output <0|1>	不输出过滤后的干净 FASTQ 文件（默认值：0，即默认输出）。
--no-bam-output-for-alignment <0|1>	当运行 alignmentsortmarkdupbqsr、basic、full 或 somatic 流程时不输出排序（去重）后的 BAM 文件（默认值：1，即默认不输出）。
--no-bam-output-for-sort <0|1>	当运行 basic、full 或 somatic 流程，且不运行 BQSR 时，不输出排序（去重）后的 BAM 文件（默认值：0，即默认输出）。
--no-bam-output-for-bqsr <0|1>	当运行 basic、full 或 somatic 流程，且运行 BQSR 时，不输出 BQSR 后的 BAM 文件（默认值：0，即默认输出）。
--output-cram <0|1>	是否输出 cram 代替 bam（默认值：0，即默认不输出 cram）

参数	说明
--gatk4 <0|1>	使用 GATK 4 模式进行分析。（默认值：0，即默认使用 GATK 3.8 流程） 可替换的模块有：BQSR、HaplotyperCaller、GenotypeGVCFs、VQSR和VariantFiltration
--temp-dir, -tmpdir <string>	临时输出目录
--traffic	查看流量信息。
--verbose, -verbose <0|1>	输出 debug 信息（默认值：0，及默认不输出）

QC

参数	说明
--soapnuke-param <string>	用户自定义 SOAPnuke 参数，用于过滤。（默认值：“-n 0.1 -q 0.5 -l 12 -Q 2 -G 2 -M 2”）

Alignment

参数	说明
--bwa <0|1>	使用 BWA 作为比对软件（默认值：0，即默认使用 Minimap2 作为比对软件）。
--se <0|1>	处理 SE 数据（默认值：0，即默认处理 PE 数据）。
--mdtag <0|1>	比对文件输出 MD tag（默认值：0，即默认不输出）。 注意 仅适用于 Minimap2。

SortMarkDup

参数	说明
--sort-markdup-input <file>	SortMarkDup 流程输入 SAM/BAM 文件（单独运行 SortMarkDup 流程时必须设置，否则不用设置）。
--sortmarkdup-input-type <sam|bam>	输入文件格式（默认值：sam）。

BQSR

参数	说明
`--bqsr-input <file>`	BQSR 流程输入 BAM 文件（单独运行 BQSR 流程时必须设置，否则不用设置）。
`--knownSites <file>`	已知 SNP/INDEL 数据库文件（*.vcf.gz），默认使用如下文件设置： ● dbsnp_151.vcf.gz ● Mills_and_1000G_gold_standard.indels.hg19. vcf.gz ● 1000G_phase1.indels.hg19.vcf.gz 注意 此参数可设置多次。
`--bqsrindex <file>`	设置已生成的 BQSR 索引文件，或需要输出的 BQSR 索引文件。
`--quantize <int>`	将量化碱基质量值分成指定级数，运行 PCR-free 数据时不允许使用（默认值：93）
`--bqsr4 <0|1>`	使用 GATK BQSR 4 参数配置，即 BAM 文件不输出 I/D 质量值标签 BD 和 BI，计算时不使用 HMM BAQ 算法（默认值：0，即默认使用 GATK BQSR 3.8 参数配置）。 注意 不能与“--diq”和“--enable-baq”参数同时使用。

HaplotypeCaller

参数	说明
`--haplotypecaller-input <file>`	Germline 变异检测流程输入 BAM 文件（单独运行 Germline 变异检测流程时必须设置，否则不用设置）。
`--intervals <file>`	intervals 文件。 注意 ● 同“--bed”参数； ● 仅当未输入 bed 文件，且运行模式为 WGS 时有效。

参数	说明
--ERC <NONE|GVCF>	设置是否输出 gVCF 文件（默认值：NONE）。 NONE：输出 VCF 文件。 GVCF：输出 gVCF 文件。
--hc4 <0|1>	使用 GATK HaplotypeCaller 4.0 做为 Germline 变异检测软件（默认值：0，即默认使用 GATK HaplotypeCaller 3.8）。 注意 请不要与“--deepvariant”参数同时使用。
--interval-padding <integer>	设置 interval padding 值（默认值：0）。 注意 仅适用于HaplotypeCaller 3.8/4.0。
--stand-call-conf <integer>	设置 stand call conf 值（默认值：30）。 注意 仅适用于HaplotypeCaller 3.8/4.0。
--pcr-indel-model <CONSERVATIVE|NONE>	是否为 PCR-Free 数据（默认值：CONSERVATIVE）。 CONSERVATIVE：PCR 数据 NONE：PCR-Free 数据 注意 仅适用于HaplotypeCaller 3.8/4.0。
--scalafile <scalafile.scala>	scala 文件，可通过此文件设置 HaplotypeCaller 参数（默认值：ExampleHaplotypeCallerFPGA. scala）。 注意 ● 仅适用于 HaplotypeCaller 3.8； ● 如需对 ERC、interval_padding、stand_call_conf 参数赋值，请优先使用对应的参数设置；当这些参数在 scala 文件中被赋值时，则以 scala 文件内容为准，通过参数所赋值将被忽略； ● 不建议用户在 scala 文件设置 dbSNP，建议使用“--vcf”设置 dbSNP； ● 默认 scala 文件内容及 scala 文件参数设置方法参见 *第 52 页“通过 scala 文件设置 HaplotypeCaller 参数”*。

参数	说明
`--arguments-file <file>`	HaplotypeCaller 参数文件，详细参见 *GATK 4.0 参数说明*（默认值：arguments_file（空文件））。 注意 仅适用于HaplotypeCaller 4.0。
`--deepvariant <0|1>`	使用 DeepVariant 做为 Germline 变异检测软件（默认值：0，即默认使用 GATK HaplotypeCaller 3.8）。 注意 请不要与“--hc4”参数同时使用。
`--use-openvino <0|1>`	是否使用 Intel OpenVINOTM 工具包加速 DeepVariant 推理（默认值：1，即使用）。 注意 仅适用于 DeepVariant。
`--MGI-data <0|1>`	是否使用针对 MGI 数据优化的模型运行 DeepVariant（默认值：1，即使用）。 注意 仅适用于 DeepVariant。
`--WGS-mode <PCR|PCR-free>`	对于 WGS 测序数据，根据不同的建库方式（PCR/PCR-Free）选取深度学习模型（默认值：PCR）。 注意 仅适用于 DeepVariant。
`--fast-model <0|1>`	是否在 WGS 任务中使用 DeepVariant 快速推理模型。 注意 ● 仅适用于 DeepVariant； ● 除非您已经知晓此参数含义，否则请不要设置；
`--deepvariant-model`	使用用户自定义模型运行 DeepVariant。 注意 ● 仅适用于 DeepVariant； ● 除非您已经知晓此参数含义，否则请不要设置；

注意 仅适用于 DeepVariant。

MuTect2

参数	说明
`--mutect2-input <file>`	MuTect2 输入 BAM 文件（单独运行 MuTect2 流程时必须设置，否则不用设置）。 注意 此参数可设置多次。
`--mutect2-arguments-file <file>`	Mutect2 参数文件，详细参见 *GATK 4.0 参数说明*（默认值：arguments_file（空文件））。
`--tumor | -tumor <string>`	肿瘤样本的样本名（单独运行 MuTect2 流程时必须设置，否则不用设置）。 注意 此样本名应与肿瘤样本输入 BAM 文件中的 Read 组样本名（RGSM）一致。
`--normal | -normal <string>`	对照样本的样本名。 注意 此样本名应与对照样本输入 BAM 文件中的 Read 组样本名（RGSM）一致。

GenotypeGVCFs

参数	说明
`--genotypegvcfs-input <file>`	GenotypeGVCFs 流程输入 gVCF 文件（单独运行 GenotypeGVCFs 流程时必须设置，否则不用设置）。 注意 此参数可设置多次。
`--stand-call-conf_genotypegvcfs <integer>`	设置 GenotypeGVCFs 流程的 stand call conf 值（默认值：10）。
`--allSites <0|1>`	使用 allSites 模式输出（默认值：0，即不使用）。

BamStats

参数	说明
`--bamstats-input <file>`	BamStats 流程输入 BAM 文件（单独运行 BamStats 流程时必须设置，否则不用设置）。

VcfStats

参数	说明
`--vcfstats-input <file>`	VcfStats 流程输入 VCF 文件（单独运行 VcfStats 流程时必须设置，否则不用设置）。

Somatic

参数	说明
`--list2 <sample.list2>`	Somatic 流程的对照样本 list 文件，格式跟 list 一致。

Extract

参数	说明
`--extract-depth <int>`	提取特定乘数数据进行变异检测分析，仅 extract 流程可以使用，设置 0 为不提取，使用所有数据（默认值：0，即默认不提取）。

VQSR

参数	说明
`--vqsr-input <file>`	VQSR 流程输入 VCF 文件（单独运行 VQSR 流程时必须设置，否则不用设置）。
`--resource-hapmap <file>`	VQSR 的 hapmap 数据库（默认值：hapmap_3.3.hg19.vcf.gz）。

参数	说明
--resource-omni <file>	VQSR 的 omni 数据库（默认值：1000G_omni2.5.hg19.vcf.gz）。
--resource-1000G <file>	VQSR 的 1000G phase1 snps high confidence 数据库（默认值：1000G_phase1.snps.high_confidence.hg19.vcf.gz）。
--resource-dbsnp <file>	VQSR 的 dbsnp 数据库（默认值：dbsnp_151.vcf.gz）。
--resource-mills <file>	VQSR 的 mills and 1000G gold standard indels 数据库（默认值：Mills_and_1000G_gold_standard.indels.hg19.vcf.gz）。

VariantFiltration

参数	说明
--variant-filtration-input <file>	Variant Filtration 流程输入 VCF 文件（单独运行 Variant Filtration 流程时必须设置，否则不用设置）。

RTGTools

参数	说明
--rtg-input <file>	RTG Tools 流程输入 VCF 文件（单独运行 RTG Tools 流程时必须设置，否则不用设置）。
--rtg-baseline-snp <file>	RTG Tools 的 SNP 变异基准 VCF 文件。
--rtg-baseline-indel <file>	RTG Tools 的 INDEL 变异基准 VCF 文件。
--rtg-evaluation-regions <file>	RTG Tools 的评价区间 BED 文件。

参数	说明
`--rtg-sdf <path>`	RTG Tools 的参考基因组序列 SDF 文件。

WGP

参数	说明
`--wgp-input <string>`	WGP 流程输入 BAM 文件（单独运行 WGP 流程时必须设置，否则不用设置）。
`--wgp-type <CNV|CNV-SV|SV>`	WGP 运行模式（默认值：CNV-SV，即运行 CNV 和 SV 分析）

7 使用示例

basic

使用默认 reference 和 dbSNP，运行模式为 WGS 的基础流程：

```
MegaBOLT --runtype WGS --list sample.list
```

使用默认 reference 和 dbSNP，待分析数据为 Single End 类型，运行模式为 WGS 的基础流程：

```
MegaBOLT --runtype WGS --se 1 --list sample.list
```

使用自定义的 reference、dbSNP 和 knownSites，运行模式为 WGS 的基础流程：

```
MegaBOLT --runtype WGS --list sample.list --ref ref.fa --vcf
b37.vcf --knownSites  b37.vcf
```

使用默认的 reference、dbSNP，运行模式为 WGS，输入样本数据采用 PCR-Free 文库制备技术，使用 BWA 作为比对工具，不输出 BQSR 后的 BAM 文件，并使用 HaplotypeCaller 4.0 进行变异检测的基础流程：

```
MegaBOLT --runtype WGS --list sample.list --pcr-indel-model
NONE --bwa 1 --hc4 1 --no-bam-output-for-bqsr 1
```

使用默认的 reference、dbSNP，运行模式为 WGS，输入样本数据采用 PCR-Free 文库制备技术，输出排序去重后的 BAM，并使用 DeepVariant 进行变异检测的基础流程：

```
MegaBOLT --runtype WGS --list sample.list --deepvariant 1
--no-bam-output-for-alignment 0 --WGS-mode PCR-free
```

使用默认的 reference、dbSNP，预设区间文件 BV5，输出到用户自定义目录，运行模式为 WES 的基础流程：

```
MegaBOLT --runtype WES --list sample.list --bed BV5
--outputdir ./out
```

使用自定义的 reference、dbSNP、knownSites 和 intervals 文件，运行模式为 WES 的基础流程：

```
MegaBOLT --runtype WES --list sample.list --ref ref.fa --vcf
b37.vcf --knownSites  b37.vcf --deepvariant 1 --bed user.bed
```

使用默认 reference 和 dbSNP，BWA 分析后进行 alt 后处理步骤并输出 hla fastq 文件，运行模式为 WGS 的基础流程：

```
MegaBOLT --runtype WGS --list sample.list --run-postalt 1
--no-hla-output 0
```

注意 上述示例的参数，在运行 full 流程时可实现同样的效果。

full

使用默认 reference、dbSNP，运行模式为 WGS 的全流程：

```
MegaBOLT --type full --runtype WGS --list sample.list
```

使用自定义的 reference、bed、dbSNP，不输出质控后的 FASTQ 文件，不做 BQSR 处理，运行模式为 WGS 的全流程：

```
MegaBOLT --type full --runtype WGS --list sample.list --ref
ref.fa --vcf b37.vcf --no-fastq-output 1 --no-bqsr 1
```

使用默认的 reference、dbSNP 和 interval 区间文件，运行模式为 WES 的全流程：

```
MegaBOLT --type full --runtype WES --list sample.list
```

使用默认 reference、dbSNP，使用 BWA 进行比对，使用 GATK BQSR 4 参数，使用 GATK HaplotypeCaller 4 进行变异检测，运行 GenotypeGVCFs，运行 VQSR，运行 RTG Tools 评价，运行模式为 WGS 的全流程：

```
MegaBOLT --type full --runtype WGS --list sample.list --bwa 1
--bqsr4 1 --hc4 1 --run-genotypegvcfs 1 --run-vqsr 1 --run-
rtgtools 1
```

somatic

使用默认 reference、dbSNP, 运行模式为 WGS，采用肿瘤 / 对照模式的体细胞变异检测流程：

```
MegaBOLT --type somatic --runtype WGS --list tumor.list --list2
normal.list
```

使用默认 reference、dbSNP，运行模式为 WGS，采用肿瘤模式的体细胞变异检测流程：

```
MegaBOLT --type somatic --runtype WGS --list tumor.list
```

使用自定义 reference、dbSNP，不做质控，不做 Haplotype 变异检测，不做统计，运行模式为 WGS，采用肿瘤 / 对照模式的体细胞变异检测流程：

```
MegaBOLT --type somatic --runtype WGS --list tumor.list --list2
normal.list --ref ref.fa --vcf b37.vcf --no-qc 1 --no-hc 1
--no-stats 1
```

使用默认 reference、dbSNP, 运行模式为 WES，使用预置 interval 文件 BV5，输出到用户指定目录，采用肿瘤 / 对照模式的体细胞变异检测流程：

```
MegaBOLT --type somatic --runtype WGS --bed BV5 --list tumor.
list --list2 normal.list --outputdir ./out
```

buildindex

使用 buildindex 构建参考基因组序列（例如：hg19.fa）相关的各种索引文件：

```
MegaBOLT --type buildindex --ref hg19.fa --knownSites
dbsnp_151.vcf.gz --knownSites Mills_and_1000G_gold_standard.
indels.hg19.vcf.gz --knownSites 1000G_phase1.indels.hg19.vcf.
gz
```

qc

使用 qc 完成对原始 FASTQ 文件的过滤和统计：

```
MegaBOLT --type qc --list sample.list
```

使用自定义 SOAPnuke 过滤参数，不输出质控后的 FASTQ 文件：

```
MegaBOLT --type qc --list sample.list --no-fastq-output 1
--soapnuke-param “-n 0.05 -q 0.5 -l 12 -Q 2 -G 2 -M 2”
```

alignment

使用 Minimap2 进行比对：

```
MegaBOLT --type alignment --list sample.list
```

使用 BWA 进行比对，待分析数据为 Single End 类型：

```
MegaBOLT --type alignment --list sample.list --bwa 1 --se 1
```

sortmarkdup

使用 sortmarkdup 进行排序与去重：

```
MegaBOLT --type sortmarkdup --sortmarkdup-input input.sam
```

使用 sortmarkdup 进行排序，输入类型为 bam，并指定输出文件名前缀：

```
MegaBOLT --type sortmarkdup --sortmarkdup-input input.bam
--sortmarddup-input-type bam --outputprefix myoutputprefix
--no-markdup 1
```

alignmentsortmarkdup

使用组合流程进行比对与排序去重：

```
MegaBOLT --type alignmentsortmarkdup --list sample.list
```

alignmentsortmarkdupbqsr

使用组合流程进行比对与排序去重以及碱基质量值再校准：

```
MegaBOLT --type alignmentsortmarkdupbqsr --list sample.list
```

bqsrindex

使用自定义 reference、dbSNP、knownSites 生成 BQSR 的 index，并将 index 保存为指定文件：

```
MegaBOLT --type bqsrindex --ref ref.fa --vcf dbsnp_151.vcf.gz
--bqsrindex ref.fa.vcfi --knownSites dbindel.vcf.gz
```

bqsr

使用默认 reference、dbSNP、knownSites 进行 BQSR 处理：

```
MegaBOLT --type bqsr --bqsr-input input.bam
```

使用自定义 reference、dbSNP 和 knownSites 进行 BQSR 处理，指定输出文件名前缀，并将生成的 BQSR index 文件保存为指定文件：

```
MegaBOLT --type bqsr --bqsrindex ref.fa.vcfi --bqsr-input
input.bam --ref ref.fa --vcf dbsnp_151.vcf.gz --knownSites
dbindel.vcf.gz --outputprefix myoutputprefix
```

haplotypecaller

使用默认 reference、dbSNP，运行模式为 WGS，使用 HaplotypeCaller 3.8 进行变异检测：

```
MegaBOLT --type haplotypecaller --runtype WGS
--haplotypecaller-input input.bam
```

使用自定义 reference、dbSNP，运行模式为 WGS，设置自定义 interval-padding、stand-call-conf，使用 HaplotypeCaller 4.0 进行变异检测：

```
MegaBOLT --type haplotypecaller --runtype WGS  --ref ref.fa
--vcf b37.vcf --scalafile example.scala --haplotypecaller-input
input.bam --interval-padding 10 --stand-call-conf 10 --hc4 1
```

使用默认 reference、dbSNP、interval 区间文件，运行模式为 WES，使用自定义 scala 文件，指定输出文件名前缀，使用 HaplotypeCaller 3.8 进行变异检测，并输出 genotype 信息：

```
MegaBOLT --type haplotypecaller --runtype WES
--haplotypecaller-input input.bam --scalafile exapmle.scala
--ERC GVCF --outputprefix myoutputprefix
```

deepvariant

使用默认 reference、dbSNP，使用 DeepVariant 快速推理模型，对 PCR 建库的 WGS 测序比对数据进行变异检测：

```
MegaBOLT --type haplotypecaller --runtype WGS --deepvariant 1
--haplotypecaller-input input.bam
```

使用自定义 reference、dbSNP，使用 DeepVariant 标准推理模型，对 PCR-Free 建库的 WGS 测序比对数据进行变异检测：

```
MegaBOLT --type haplotypecaller --deepvariant 1 --ref ref.fa
--vcf b37.vcf --haplotypecaller-input input.bam --runtype WGS
--WGS-mode PCR-free --fast-model 0
```

使用默认 reference、dbSNP、interval 区间文件，使用 DeepVariant 标准推理模型，对 WES 测序比对数据进行变异检测，并输出 genotype 信息：

```
MegaBOLT --type haplotypecaller --deepvariant 1
--haplotypecaller-input input.bam --runtype WES --intervals
BV4 --ERC GVCF
```

mutect2

使用默认 reference、dbSNP，采用肿瘤 / 对照模式的体细胞变异检测：

```
MegaBOLT --type mutect2 --mutect2-input tumor.bam
--mutect2-input normal.bam --tumor tumorsamplename --nomal
normalsamplename
```

使用自定义 reference、dbSNP，采用肿瘤模式的体细胞变异检测，结果输出到指定目录：

```
MegaBOLT --type mutect2 --mutect2-input tumor.bam --tumor
tumorsamplename --ref ref.fa --vcf b37.vcf --outputdir ./out
```


genotypegvcfs

使用默认 reference、dbSNP，进行联合基因分型：

```
MegaBOLT --type genotypegvcfs --genotypegvcfs-input input.
g.vcf.gz
```

使用自定义 reference、dbSNP、genotypegvcfs-stand-call-conf，进行联合基因分型，输出所有位点信息：

```
MegaBOLT --type genotypegvcfs --genotypegvcfs-input input.
g.vcf.gz --ref ref.fa --vcf b37.vcf --genotypegvcfs-stand-
call-conf 30 --allSites 1
```

vqsr

使用默认数据库，对变异检测结果中变异质量值进行数据内部再校准：

```
MegaBOLT --type vqsr --vqsr-input input.vcf.gz
```

使用自定义数据库，对变异检测结果中变异质量值进行数据内部再校准：

```
MegaBOLT --type vqsr --vqsr-input input.vcf.gz --resource-
hapmap hapmap_3.3.hg19.vcf.gz --resource-omni 1000G_
omni2.5.hg19.vcf.gz --resource-1000G 1000G_phase1.snps.high_
confidence.hg19.vcf.gz --resource-dbsnp dbsnp_151.vcf.gz
--resource-mills Mills_and_1000G_gold_standard.indels.hg19.
vcf.gz
```

filtration

使用 filtration 对变异检测数据进行过滤：

```
MegaBOLT --type filtration --variant-filtration-input input.
vcf.gz
```

rtgtools

基于默认变异标准集，对 VCF 文件中的 SNP/INDEL 变异位点的假阳性、假阴性、准确度和灵敏度进行评价：

```
MegaBOLT --type rtgtools --rtg-input input.vcf.gz
```

基于自定义变异标准集对 WES 数据 VCF 文件中的 SNP/INDEL 变异位点的假阳性、假阴性、准确度和灵敏度进行评价：

```
MegaBOLT --type rtgtools --rtg-input input.vcf.gz --runtype
WES --bed BV5.bed --ref hg19.fa --rtg-baseline-snp snp.tp.vcf.
gz --rtg-baseline-indel indel.tp.vcf.gz --rtg-evaluation-
regions highconf.bed --rtg-sdf hg19.fasta.SDF
```

bamstats

统计 Paired End 数据生成的 BAM 文件，输出到指定目录：

```
MegaBOLT --type bamstats --bamstats-input input.bam --outputdir
./out
```

评价 Single End 数据生成的 BAM 文件，输出到指定目录：

```
MegaBOLT --type bamstats --bamstats-input input.bam --outputdir
./out --se 1
```

vcfstats

评价 vcf 文件，输出到指定目录：

```
MegaBOLT --type vcfstats --vcfstats-input input.vcf.gz
--outputdir ./out
```

builddict

为指定 reference 生成字典文件：

```
MegaBOLT --type builddict --ref ref.fa
```

buildfai

为指定 reference 生成索引文件：

```
MegaBOLT --type buildfai --ref ref.fa
```

buildbed

为指定 reference 生成 effective bed 文件：

```
MegaBOLT --type buildbed --ref ref.fa
```

bwaindex

为指定 reference 生成 bwa 索引文件：

```
MegaBOLT --type bwaindex --ref ref.fa
```

extract

使用默认 reference、dbSNP，运行模式为 WES 的提取流程（提取 100X）：

```
MegaBOLT --type extract --runtype WES --list sample.list
--extract-depth 100
```

bamtocram

将 BAM 文件转换为 CRAM 文件：

```
MegaBOLT --type bamtocram --bam-input input.bam
```

wgp

使用 BAM 文件进行 CNV 和 SV 分析：

```
MegaBOLT --type wgp --wgp-input input.bam
```

使用 BAM 文件进行 CNV 分析：

```
MegaBOLT --type wgp --wgp-input input.bam --wgp-type CNV
```

在 WGS 的全流程中，进行 CNV 和 SV 分析，将使用 BWA 作为比对软件：

```
MegaBOLT --type full --runtype WGS --list sample.list
--run-wgp 1
```

--- 此页有意留白 ---

8 输出目录和结果

输出文件名

程序输出文件名根据选择的步骤添加对应的后缀，顺序为程序执行顺序。
如默认流程输出结果为：
Bam: samplename.mm2.sortdup.bqsr.bam
顺序运行 Minimap2，SortMarkDup 和 BQSR
Vcf: samplename.mm2.sortdup.bqsr.hc.vcf.gz
顺序运行 Minimap2，SortMarkDup，BQSR 和 HaplotypeCaller

选择流程时添加的对应后缀如下：

流程选择	后缀添加
默认比对，Minimap2	mm2
可选比对，bwa	bwa
默认排序去重	sortdup
可选只排序	sort
碱基质量值校正	bqsr
默认变异检测，HaploeyptCaller 3.8	hc
可选变异检测，HaplotypeCaller 4.0	hc4
可选深度学习变异检测，DeepVariant	dv
可选 MuTect2	mutect2
可选 GenotypeGVCFs	genotype
运行 GenotypeGVCFs 时设置 allSites	genotype.allsites
可选 VQSR	vqsr
可选 VariantFiltration	filtration
可选提取流程，extract（BQSR 之后提取，提取后默认运行 markdup）	extract.mkdup
可选提取流程，extract（BQSR 之后提取，设置”--no-markdup 1”）	extract

注：输出 GVCF 文件为 *.g.vcf.gz

Germline 变异检测流程

程序运行成功，将在任务输出目录下生成如下目录树：

```
├── megabolt.log
├── megabolt.out
└── samplename/
    ├── samplename_1.fq.gz
    ├── samplename_2.fq.gz
    ├── samplename.list
    ├── samplename.log
    ├── samplename.mm2.sortdup.bqsr.bam
    ├── samplename.mm2.sortdup.bqsr.bam.bai
    ├── samplename.mm2.sortdup.bqsr.bam.grp
    ├── samplename.mm2.sortdup.bqsr.hc.vcf.gz
    ├── samplename.mm2.sortdup.bqsr.hc.vcf.gz.out
    ├── samplename.mm2.sortdup.bqsr.hc.vcf.gz.tbi
    ├── samplename.out
    ├── WGP_output/（仅当包含 WGP 流程时）
    │   ├── log/
    │   ├── cnvnator/
    │   ├── plot/
    │   ├── sv_breakdancer/
    │   ├── WGP_output.zip
    ├── report/（仅全流程）
    │   ├── samplename_cn.html
    │   ├── samplename_en.html
    │   └── samplename.report.zip
    │   └── Statistics_of_Filtered_Reads.txt
    └── stat/（仅全流程）
        ├── bam_stats/
        │   ├── cumu.txt
        │   ├── depth_frequency.txt
        │   ├── samplename.bamstat.xls
        │   ├── samplename.CollectInsertSizeMetrics.txt
        │   ├── samplename.cumuPlot.png
        │   ├── samplename.depthstat.xls
        │   ├── samplename.gc_bias_metrics.xls
        │   ├── samplename.gcbias.png
```

```
|   ├── samplename.histPlot.png
|   ├── samplename.insertsize.png
|   ├── samplename.samtoolsstat.xls
|   └── samplename.Summary.xls
|   └── hs_metrics.txt（仅 WES 流程）
|   └── coverage.report（仅 WES 流程）
|   └── chromosomes.report（仅 WES 流程）
|   └── insertsize.plot（仅 WES 流程）
|   └── depth_distribution.plot（仅 WES 流程）
|   └── depth.tsv.gz（仅 WES 流程）
|   └── region.tsv.gz（仅 WES 流程）
|   └── uncover.bed（仅 WES 流程）
├── qc/
|   ├── samplename_1.fq.gz.check
|   ├── samplename_2.fq.gz.check
|   ├── samplename.base.png
|   ├── samplename.fqstat.xls
|   └── samplename.qual.png
├── Statistics_of_Filtered_Reads.txt
├── vcf_stats/
|   └── samplename.vcfstat.xls
└── rtgtools/（仅运行 rtgtools 流程时）
    ├── indel/
    ├── snp/
    └── highconf.bed（仅 WES 流程）
```

任务输出目录结构如下：

megabolt.log	client 程序标准输出流 / 标准错误流信息。
megabolt.out	client 程序运行时日志。
samplename/	结果目录。

结果目录结构如下（以 100GB gzip 文件输入为例，子模块输出结果和运行时产生的 log 文件存放在同级目录下）：

samplename.fq.gz	过滤后的 Reads 文件（100GB）。
samplename.list	转置后的样品 list 文件。
samplename.bam*	经过排序标重 BQSR 后的比对文件和 index（200GB）。
samplename.vcf*	变异检测结果和 index（VCF 200MB 和 gVCF 6GB）。
samplename.log	分析任务日志。
samplename.out	程序运行时产生的日志。
WGP_output/	WGP 流程分析结果（仅当包含 WGP 流程时）

report/	样本报告，以及样本报告的压缩包（仅全流程）。
stat/	统计文件（仅全流程）。

以下文件仅在包含 WGP 流程时产生：

WGP 流程分析结果（WGP_output/）：

log/	WGP 流程分析日志。
cnvnator/	CNV 分析结果。
plot/	CNV/SV 绘图结果。
sv_breakdancer/	SV 分析结果。
WGP_output.zip	打包的 WGP 流程分析结果。

以下文件在运行全流程后产生：

全流程结果报告（report/）：

report.zip	打包的所有样本的报告（*/*.html）。
*_en.html	英文版样本分析报告。
*_cn.html	中文版样本分析报告。

全流程统计文件（stat/）：

bam_stats/	比对结果统计信息。
qc/	质控结果统计信息。
vcf_stats/	变异检测结果统计信息。
rtgtools/	变异位点评价信息。

samplename.fqstat.xls（qc/）：

Sample	WGS
Read_length	100:100
Read_raw	941148380
Read_clean	941148366
Rate_clean	100%
Q20_raw	97.1%
Q20_1_raw	98.3%
Q20_2_raw	95.89%
Q30_raw	88.5%
Q30_1_raw	91.99%
Q30_2_raw	85%
GC_raw	41%
Q20_clean	97.1%
Q20_1_clean	98.3%
Q20_2_clean	95.89%
Q30_clean	88.5%
Q20_2_clean	91.99%
Q30_clean	85%
GC_clean	41%

AT_1_Separation	0.00%
AT_2_Separation	0.00%
GC_1_Separation	0.20%
GC_2_Separation	0.20%

samplename.bamstat.xls（bam_stats/）：

Sample	WGS
Mapping_Rate	99.01%
PE_Mapping_Rate	98.62%
Unique_Rate	95.07%
Duplication_Rate	1.79%
Mismatch_Rate	0.54%
Insert_size	387.8
Average_depth(rmdup)	31.22
Coverage(>=1X)	99.10%
Coverage(>=2X)	98.92%
Coverage(>=3X)	98.73%
Coverage(>=4X)	98.53%
Coverage(>=5X)	98.32%
Coverage(>=10X)	97.47%
Coverage(>=20X)	92.41%
Uniformity(>0.2f)	97.94%

samplename.vcfstat.xls（vcf_stats）

Sample	WGS
Total_SNP	3844409
dbSNP_rate	98.66%
Novel_SNP	51597
Novel_SNP_Rate	1.34%
Ti/Tv	2.01
Total_INDEL	859574
dbINDEL_Rate	86.29%

Somatic 变异检测流程

程序运行成功，将在结果输出目录下生成如下目录树：

```
├── megabolt.log
├── megabolt.out
└── tumor-name_normal-name/
    ├── normal-name/
    ├── tumor-name/
    ├── tumor-name_normal-name.log
    ├── tumor-name_normal-name.Mutect2.vcf
```

└—— tumor-name_normal-name.out

任务输出目录结构如下：

megabolt.log	client 程序标准输出流 / 标准错误流信息。
megabolt.out	client 程序运行时日志。
tumor-name_normal-name/	结果目录。

注意 关于 MegaBOLT client 的详细信息，参见 *MegaBOLT 高级用户手册*。

结果目录结构如下（以 100GB gzip 文件输入为例，子模块输出结果和运行时产生的 log 文件存放在同级目录下）：

normal-name/	肿瘤样本分析结果目录。
tumor-name/	对照样本分析结果目录。
tumor-name_normal-name.log	分析任务日志。
tumor-name_normal-name.Mutect2.vcf	Somatic 变异检测结果 VCF 文件。
tumor-name_normal-name.out	程序运行时产生的日志。

肿瘤样本分析结果目录结构、对照样本分析结果目录结构与 Germline 变异检测流程的结果目录一致。

9 参数设置注意事项

--ref、--vcf 和 --knownSites 参数关系

通过 --ref 设置的参考基因组文件（以下简称 ref 文件），通过 --vcf 设置的 dbSNP 文件和通过 --knownSites 设置的已知 SNP/INDEL 数据库文件（以下简称 knownSites 文件）需要保证相互匹配，否则 MegaBOLT 不能保证分析结果的正确性。

当输入 --ref、--vcf 和 --knownSites 参数中的一个或者多个，程序的具体行为是不同的，详细说明如下表所示：

是否设置 --ref	是否设置 --vcf	是否设置 --knownSites	是否允许	说明
否	否	否	是	使用默认 ref 文件（hg19.fa）、默认 dbSNP 文件（dbsnp_151.vcf.gz）和默认 knownSites 文件（dbsnp_151.vcf.gz、Mills_and_1000G_gold_standard.indels.hg19.vcf.gz、1000G_phase1.indels.hg19.vcf.gz）。
否	否	是	否	因为用户未设置 ref 文件，无法确认用户设置的 dbSNP/knownSites 文件是否与默认 ref 文件（hg19.fa）匹配。
否	是	否		
否	是	是		
是	否	否	是	用户设置了 ref 文件，但没有设置 dbSNP 和 knownSites 文件。Germline 变异检测流程将不会使用 dbSNP。程序尝试在用户设置的 ref 文件所在目录查找 BQSR 索引文件，如果找到索引文件，则使用该索引文件运行 BQSR 流程，否则 BQSR 流程不会运行。
是	否	是	是	用户设置了 ref 文件，但没有设置的 dbSNP 文件。Germline 变异检测流程将不会使用 dbSNP。

是否设置--ref	是否设置--vcf	是否设置--knownSites	是否允许	说明
是	是	否	是	用户设置了 ref 文件和 dbSNP 文件，但没有设置的 knownSites 文件。程序将使用用户设置的 dbSNP 文件作为 knownSites 文件。
是	是	是	是	使用用户设置的 ref、dbSNP 和 knownSites 文件。不对文件的匹配性做验证。

构建 BQSR 索引文件（--bqsrindex）

BQSR 流程运行前，需要构建 BQSR 索引文件。构建 BQSR 索引一般会自动完成，用户也可以根据需求运行 bqsrindex 流程构建。

根据用户是否设置 --ref、--vcf、--knownSites 和 --bqsrindex，构建 BQSR 索引文件的策略有所不同，详细说明如下表所示：

是否设置--ref	是否设置--vcf 或--knownSites	是否设置--bqsrindex	是否允许	说明
否	否	否	是	使用已经提前构建的默认 BQSR 索引。
否	否	是	否	因为用户未设置 ref 文件，无法确认用户设置的 dbSNP/knownSites/BQSR 索引文件是否与默认 ref 文件（hg19.fa）匹配。
否	是	否		
否	是	是		
是	否	否	是	程序尝试在用户设置的 ref 文件所在目录查找 BQSR 索引文件，如果找到索引文件，则使用该索引文件运行 BQSR 流程，否则 BQSR 流程不会运行。
是	否	是	是	使用用户通过 --bqsrindex 参数设置的文件作为 BQSR 索引文件运行 BQSR 流程。

是否设置 --ref	是否设置 --vcf 或 --knownSites	是否设置 --bqsrindex	是否允许	说明
是	是	否	是	程序尝试在用户设置的 ref 文件所在目录查找 BQSR 索引文件，如果找到索引文件，则使用该索引文件运行 BQSR 流程。否则，首先尝试使用用户设置的 ref/dbSNP/knownSites 文件在 ref 文件所在目录构建 BQSR 索引文件；如果该目录不可写，则在用户输出目录构建 BQSR 索引文件。
是	是	是	是	如果用户通过 --bqsrindex 参数设置的文件存在，则使用该文件作为 BQSR 索引文件运行 BQSR 流程。否则使用用户设置的 ref/dbSNP/knownSites 文件在 --bqsrindex 参数指定的位置构建 BQSR 索引文件，并使用该文件运行 BQSR 流程。

--ref、--bed 和 --runtype 参数关系

当用户使用 --ref、--bed 和 --runtype 参数组合时，需要注意它们之间存在的一些约束关系，详细说明如下表所示：

是否设置 --ref	是否设置 --bed	是否设置 --runtype	是否允许	说明
否	否	否 / 是（WGS）	是	运行 WGS 模式。
否	否	是（WES）	是	运行 WES 模式，默认使用 BV4 作为 bed 文件。
否	是	否/是（WES）	是	如果用户设置的 bed 在已知 bed 列表中，则使用用户设置的 bed 文件运行 WES 模式。否则程序无法运行，因为程序无法确认用户输入的 bed 文件是否与默认 ref 文件（hg19.fa）匹配。
否	是	是（WGS）	否	WGS 运行模式下不允许指定 bed 文件。
是	否	否 / 是（WGS）	是	运行 WGS 模式。

是否设置 --ref	是否设置 --bed	是否设置 --runtype	是否允许	说明
是	否	是（WES）	否	用户设置了 ref 文件，并且选择运行 WES 模式，但是没有设置与 ref 文件相匹配的 bed 文件，程序无法运行。
是	是	否 / 是（WES）	是	如果用户设置的 bed 在已知 bed 列表中，若设置的 ref 文件包含”19”或”38”字段，则运行 WES 模式，若 ref 文件包不含”19”和”38”字段，程序无法运行。 如果用户设置的 bed 不在已知 bed 列表中，则运行 WES 模式，并且使用用户设置的 bed 文件。
是	是	是（WGS）	否	WGS 运行模式下不允许指定 bed 文件。

通过 scala 文件设置 HaplotypeCaller 参数

参数设置说明

如需对 ERC、interval_padding、stand_call_conf 参数赋值，请优先使用对应的参数设置；当这些参数在 scala 文件中被赋值时，则以 scala 文件内容为准，通过参数所赋值将被忽略；

不建议用户在 scala 文件设置 dbSNP，建议使用“--vcf”设置 dbSNP。

基于默认 scala 文件设置参数，方法如下：

GATK 参数	scala 文件设置方法（varcall 函数内）
-stand_call_conf 30	this.stand_call_conf = 30
-emitRefConfidence	this.emitRefConfidence = ReferenceConfidenceMode.GVCF

默认 scala 文件

文件名：*ExampleHaplotypeCallerFPGA.scala*

文件内容：

```
package org.broadinstitute.gatk.queue.qscripts.examples
import org.broadinstitute.gatk.queue.QScript
import org.broadinstitute.gatk.queue.extensions.gatk._
import org.broadinstitute.gatk.utils.commandline.Hidden
import org.broadinstitute.gatk.utils.commandline._
import org.broadinstitute.gatk.queue.util.QScriptUtils
import org.broadinstitute.gatk.queue.function.
ListWriterFunction
import org.broadinstitute.gatk.utils.variant.GATKVCFIndexType
import org.broadinstitute.gatk.tools.walkers.haplotypecaller.
ReferenceConfidenceMode
import org.broadinstitute.gatk.utils.pairhmm.PairHMM.HMM_
IMPLEMENTATION
import org.broadinstitute.gatk.tools.walkers.haplotypecaller.
PairHMMLikelihoodCalculationEngine.PCR_ERROR_MODEL

class ExampleHaplotypeCaller extends QScript {
  qscript =>
  @Input(doc="The reference file for the bam files.",
shortName="R")
  var referenceFile: File = _ // _ is scala shorthand for null
  @Input(doc="Bam file to indel realigner.", shortName="I")
  var bamFile: File = _
  @Input(doc="Vcf file.", shortName="O")//, required=false
  var vcfFile: File = _
  @Input(doc="an intervals file to be used by GATK - output
bams at intervals only", fullName="gatk_interval_file",
shortName="intervals", required=false)
  var intervals: File = _
  @Argument(doc="Is output gvcf file.", shortName="ERC",
required=false)
  var emitRefConfidence: String = _
  @Argument(doc="Parameter stand_call_conf.", shortName="stand_
call_conf", required=false)
  var stand_call_conf: Int = 10
  @Input(doc="Parameter dbsnp.", shortName="dbsnp",
required=false)
  var dbsnp: File = _
  @Argument(doc="Parameter interval_padding.",
shortName="interval_padding", required=false)
```

```
  var interval_padding: Int = 0
  @Argument(doc="Parameter pcr_indel_model.", shortName="pcr_
indel_model", required=false)
  var pcr_indel_model: String = _
  @Hidden
  @Argument(doc="How many ways to scatter/gather",
fullName="scatter_gather", shortName="sg", required=false)
  var nContigs: Int = -1
  trait CommandLineGATKArgs extends CommandLineGATK {
    this.reference_sequence = qscript.referenceFile
  }

  case class varcall (inBam: File, outVCF: File) extends
HaplotypeCaller with CommandLineGATKArgs {
    this.input_file :+= inBam
    if(qscript.emitRefConfidence != null && qscript.
emitRefConfidence == "GVCF"){
        this.emitRefConfidence = ReferenceConfidenceMode.GVCF
    }
    if(this.emitRefConfidence == ReferenceConfidenceMode.GVCF){
        if(!outVCF.endsWith(".g.vcf") && !outVCF.endsWith(".
g.vcf.gz")){
                this.out = outVCF.replace(".vcf", ".g.vcf")
                print("Changing name for GVCF file to " +
this.out + "\n")
        }
        else{
                this.out = outVCF
        }
    }
    else{
        if(outVCF.endsWith(".g.vcf") || outVCF.endsWith(".
g.vcf.gz")){
                this.out = outVCF.replace(".g.vcf", ".vcf")
                print("Changing output name for VCF file to "
+ this.out + "\n")
        }
        else{
                this.out = outVCF
        }
```

```
}
    if(qscript.pcr_indel_model != null)
    {
        if(qscript.pcr_indel_model == “NONE”)
        {
                this.pcr_indel_model = PCR_ERROR_MODEL.NONE
        }
        if(qscript.pcr_indel_model == “CONSERVATIVE”)
        {
                this.pcr_indel_model = PCR_ERROR_MODEL.
CONSERVATIVE
        }
        if(qscript.pcr_indel_model == “HOSTILE”)
        {
                this.pcr_indel_model = PCR_ERROR_MODEL.HOSTILE
        }
        if(qscript.pcr_indel_model == “AGGRESSIVE”)
        {
                this.pcr_indel_model = PCR_ERROR_MODEL.
AGGRESSIVE
        }
    }

//    this.pcr_indel_model = PCR_ERROR_MODEL.NONE
    this.interval_padding = qscript.interval_padding
    this.stand_call_conf = qscript.stand_call_conf
    if(qscript.dbsnp != null){
        this.dbsnp = qscript.dbsnp
    }
    this.intervals = if (qscript.intervals == null) Nil else
List(qscript.intervals)

// 增减参数

    this.nct = 3
    this.variant_index_type =  GATKVCFIndexType.LINEAR
    this.variant_index_parameter = 128000
    this.scatterCount = qscript.nContigs
    this.memoryLimit = 4
```

```
    this.pair_hmm_implementation = HMM_IMPLEMENTATION.VECTOR_
LOGLESS_CACHING_FPGA_EXPERIMENTAL
  }

  def script() {
      nContigs = 24
      val recalBam = qscript.bamFile
      val finalVCF = qscript.vcfFile
      add(varcall(recalBam, finalVCF))
    }
}
```

DeepVariant 参数说明

DeepVariant 变异检测模块是 MegaBOLT 的可选子模块，DeepVariant 从 HaplotypeCaller 流程继承变异检测基本参数，下述 HaplotypeCaller 流程参数对 DeepVariant 有效：

```
--haplotypecaller-input
--ERC
--deepvariant
--use-openvino
--MGI-data
--WGS-mode
--fast-model
--deepvariant-model
```

DeepVatiant 对不同类型的待分析数据提供对应的深度学习模型，并根据用户指定的下列参数：

```
--mgi-data
--runtype
--WGS-mode
```

选取对应的深度学习模型。

此外，当用户使用 --deepvariant-model、--fast-model 与 --use-openvino 参数组合时，需要注意它们之间存在的一些约束关系，这些参数组合的合法性与如下表所示：

--deepvariant-model 是否指定	--fast-model 参数值	--use-openvino 参数值	是否允许	说明
否	0	0	是	使用 TensorFlow 进行深度学习变异检测，速度比使用 OpenVINO 慢。
否	0	1	是	通过 OpenVINO 加速模型推理流程，采用标准的 DeepVariant 模型结构。
否	1	0	否	目前 MegaBOLT 不支持通过 TensorFlow 使用快速推理模型变异检测。
否	1	1	是	通过 OpenVINO 加速模型推理流程，采用 DeepVariant 快速推理模型。
是	0	0	是	使用用户自定义深度学习模型进行变异检测。
是	0	1	否	OpenVINO 不支持用户训练的 TensorFlow 模型。
是	1	0	是	如果用户指定了自定义模型，将会跳过快速模型选取。
是	1	1	否	OpenVINO 不支持用户训练的 TensorFlow 模型。

DeepVariant 允许用户通过 --deepvariant-model 参数使用自定义的推理模型进行变异检测，但是本功能仅适用于有一定深度学习开发经验的用户。除非您已经知晓此参数含义，否则请不要设置。例如使用默认 reference、dbSNP，使用自定义 DeepVariant 推理模型，对输入 WGS 测序比对数据进行变异检测的命令行如下：

```
MegaBOLT --type haplotypecaller --deepvariant 1
--haplotypecaller-input input.bam --sample-mode WGS
--deepvariant-model user.model.ckpt --fast-model 0 --use-
openvino 0
```

--- 此页有意留白 ---

10 软件更新日志

日期	版本	更新记录
2017/12/15	V1.0	初始版本。
2019/05/07	V1.5.4	● Minimap2 作为默认比对软件（Qaligner 可选）。 ● 支持 SE 数据。 ● 包含提取固定深度分析流程。 ● 更新 BQSR，使结果与原版基本保持一致。 ● 流程包含 MegaBOLT 与 MegaBOLT-full 两个流程。
2019/09/23	V1.5.6	● 添加 allnomarkdup、alignmentsort 和 sort 流程。 ● 添加 DeepVariant 流程。 ● 增加 BWA 作为比对软件选择。
2019/11/30	V2.1.0	● 多任务调度版本。 ● 合并原 MegaBOLT（basic）和原 MegaBOLT-full （full）流程。 ● 增加 Somatic 全流程分析。 ● 所有全流程各个步骤可单独运行。 ● 暂不支持提取固定深度分析流程。 ● 暂不支持 Qaligner 作为比对软件。 ● 暂不支持 SOAPnuke 参数设置。 ● 更新默认 references。
2020/04/23	V2.2.1.1	● 支持用户设置临时分析目录。 ● 支持 SOAPnuke 参数设置，修改默认参数。 ● 支持构建 BWA 索引。 ● 支持 GenotypeGVCFs。 ● 添加提取流程 extract。 ● 更新 DeepVariant 模型。 ● 加速 GVCF 输出。 ● 修复 bam 索引匹配问题。 ● 更新 GATK 到 3.8-1。 ● 用 MegaBOLT 替代 root 运行 server。

日期	版本	更新记录
2020/05/15	V2.2.2.1	● 支持 cram 输出。 ● 支持 VQSR。 ● 支持输出性别信息。 ● 优化 QC 步骤 IO。 ● 添加统计信息。
2020/06/08	V2.2.2.2	● 优化变异检测 driver。 ● 优化 server。 ● 优化 BWA 资源管理。 ● 优化 index 构建。 ● 添加构建 effective bed。 ● 添加 QC 统计。 ● 过滤 supplementary 比对的 bam 统计。 ● 更新 BQSR 到 v2.3.5。 ● 更新 BWA，需更新 index。 ● 修复已知 bugs。
2020/10/16	V2.2.3.0	● 支持 Varint Filtration 流程 ● 支持 RTG Tools 评价流程 ● 支持 BQSR 使用 GATK 4 参数 ● 支持构建 reference 索引流程 ● 在 QcStats 中添加输出过滤统计信息 ● 在 BamStats 中添加“Fold 80 base penalty”统计指标（仅 WES 流程） ● 修复 QcStats 对不等长 read 分布统计异常的问题 ● 修复 HaplotypeCaller 内存异常问题 ● 修复 FPGA driver 停止异常问题 ● 修复 BWA 已知 bug ● 升级 HaplotypeCaller4 到 GATK 4.1.8.1 ● 优化报告展示 ● 优化加密锁逻辑

日期	版本	更新记录
2021/05/18	V2.3.0.0	● 支持使用内置参考文件集合 hg19/hg38/hs37d5 ● 支持使用 GATK4 模式进行分析 ● 支持 WGP 流程进行 CNV 和 SV 分析，在 WGP 流程中使用 BWA 作为比对软件 ● 优化 BAM 统计模块 ● 优化流量展示，输出流量使用比例，增加设置最大流量使用比例 ● 优化报告展示模块，支持 RTG tools 评价结果展示 ● 升级内置 GATK4 版本至 v4.1.8.1 ● 升级 BWA 模块，需要重新构建索引文件 ● 修复 HaplotypeCaller4 缓存冲突问题 ● 修复构建 BQSR 索引时，不支持 ref 和 knownsite 文件 contig 不一致的问题 ● 其它优化
2022/05/30	V2.4.0.0	● 删除 BQSR 的 diq、baq 参数 ● 新增 BQSR 的 quantize 参数 ● 更新 bed 文件名 ● 新增全局参数 run-postalt 和 no-hla-output

--- 此页有意留白 ---