Sentieon

用户手册

Sentieon

2025 年 12 月 30 日

目录

1 Sentieon 简介

Sentieon软件为完整的纯软件基因变异检测二级分析方案，其分析流程完全忠于BWA/GATK/MuTect2/STAR/Minimap2/Fgbio/picard等金标准的数学模型。在匹配开源流程分析结果的前提下，大幅提升WGS、WES、Panel、UMI、ctDNA、RNA等测序数据的分析效率和检出精度，并匹配目前全部第二代、三代测序平台，对短读长NGS、长读长测序数据进行SNP/INDEL/SV分析。

Sentieon Genomics Software		
支持平台	illumina、MGI、PacBio、ONT、GeneMind、IonTorrent、Element、Ultima...	
应用领域	群体队列测序	大样本量全基因组、全外显子测序
		人群队列项目、大规模联合变异检测
	遗传病诊断	临床全基因组/外显子组/Panel测序
	肿瘤诊断	准确从冷冻、FFPE、血样中检测肿瘤基因变异
	农业基因组学	分析动植物等包含超大基因组或者复杂染色体的样本
	基因编辑	在全基因组范围内计算on-target和off-target的编辑效率
软件能力	测序数据质控	为FASTQ及BAM文件质控提供高速质控统计工具
	测序序列比对	为DNA/RNA/三代数据提供加速版BWA、STAR、Minimap2
	胚系变异检测	DNAseq：匹配GATK多个版本结果，提速5-10倍
		DNAscope：机器学习模型，适配所有二代、三代测序平台
	肿瘤变异检测	TNseq：匹配MuTect2 v4.2结果，提速5-10倍
		TNscope：速度匹配TNseq，更高灵敏度，适用于ctDNA
	结构变异检测	支持ONT、PacBio三代长读长数据
		支持NGS短读长测序数据
	肿瘤MRD监测	UMI模块：多种类型、UMI去重，提速10倍以上
		TNscope：提供高灵敏低频变异检测
	群体队列分析	大样本量的joint calling，一次处理超过20万个WGS样本
		96线程单机处理万人gvcf>vcf用时不超过4天
		支持跨不同测序平台测序数据联合分析
	RNA变异检测	完全匹配GATK的RNAseq最佳实践，加速5-10倍
	IVD检测一体机	USB密钥离线授权，后台运行，高速稳定分析

图 1-1 Sentieon 基因组学软件

Sentieon 为纯 CPU 计算加速软件，完全适配主流 CPU 计算架构：Intel、AMD、海光等 X86 架构 CPU，华为鲲鹏、阿里倚天、Ampare、Apple Silicon 等 ARM 架构 CPU。可灵活部署在单机工作站、HPC 集群、超算中心和云计算中心，保持同一套流程下不同规模数据计算结果的一致性。

Sentieon 软件团队拥有丰富的软件开发及算法优化工程经验，致力于解决生物数据分析中的速度与准确度瓶颈，**为来自于分子诊断、药物研发、临床医疗、人群队列、动植物等多个领域的合作伙伴提供高效精准的软件解决方案，共同推动基因技术的发展。**

截至 2025 年 7 月份，Sentieon 软件已经在全球范围内为 1860+用户提供服务，用户处理超过 4980+PB 数据量，被世界一级影响因子刊物如 NEJM、Cell、Nature 等广泛引用，引用次数超过 1500 篇。此外，Sentieon 连续数年摘得了 Precision FDA、Dream Challenges 等多个权威评比的桂冠，在业内获得广泛认可。

2 Sentieon 快速入门指南

2.1 运行环境

2.1.1 硬件要求

Sentieon Genomics 软件设计用于在 Linux 和其他 POSIX 兼容平台。

对于 Linux 系统，我们建议使用具有以下配置的 Linux 服务器：

- 运行以下分发版本之一或更高版本的 Linux：RedHat/CentOS 6.5、Debian 7.7、OpenSUSE-13.2 或 Ubuntu-14.04。
- 小型面板或全外显子组需要 16GB 内存，全基因组需要 64GB 内存。
- （推荐）首选高速 SSD 驱动器，以获得理想的 I/O 性能，实现最大 CPU 利用率。

2.1.2 软件要求

需要 Python 2.6.x、Python 2.7.x 或 python3.x。您可以通过输入以下命令检查 Python 版本：

```
python --version
```

2.1.3 Sentieon®软件发布包

从 Sentieon 技术支持提供的链接下载软件包。

通过运行以下命令解压缩软件包，其中 VERSION 是您使用的版本，例如 202503.02：

```
tar xvzf sentieon-genomics-VERSION.tar.gz
```

2.1.4 许可证要求

Sentieon®软件是一个商用许可的软件。用户需要正确设置许可证才能运行软件。我们提供两种类型的许可证：

- **单机评估许可证：**此许可证用于在单台机器上评估 Sentieon®软件。它允许新用户快速开始使用软件，而无需 IT 部门的帮助。要使用此许可证，计划运行 Sentieon®软件的计算机需要外部 Internet 访问。
- **集群许可证：**此许可证用于集群环境。使用此许可证，一个轻量级的浮动许可证服务器进程在集群中的一个节点上运行，通过 TCP 向所有其他与许可证服务器有网络连接的节点提供许可证。此许可证服务器在集群外围的一个特殊非计算节点上运行，该节点可以通过 HTTPS 不受限制地访问外部世界，并通过监听集群内需要开放的特定 TCP 端口向集群中的其余节点提供许可证。

（1）设置单机评估许可证

要使用单机评估许可证，计算节点需要能够访问 Internet。这允许 Sentieon®软件验证许可证。

要使用单机评估许可证，请按照以下步骤操作：

1. 将许可证文件复制到计算节点。例如，许可证文件 LICENSE_FILE.lic 现在位于 LICENSE_DIR。
2. 按如下方式设置环境变量：

```
export  SENTIEON_LICENSE=LICENSE_DIR/LICENSE_FILE.lic
```

（2）设置许可证服务器

如图 2-1 所示，许可证服务器需要满足以下条件：

1. 许可证服务器应该能够访问 Internet 以执行许可证验证。

2. 计算节点应该能够通过主机名 LICSRVR_HOST 访问许可证服务器。

3. 运行许可证服务器的机器有一个开放的端口供许可证服务监听，并且计算节点可以访问该端口。这里我们假设可用端口是 LICSRVR_PORT。

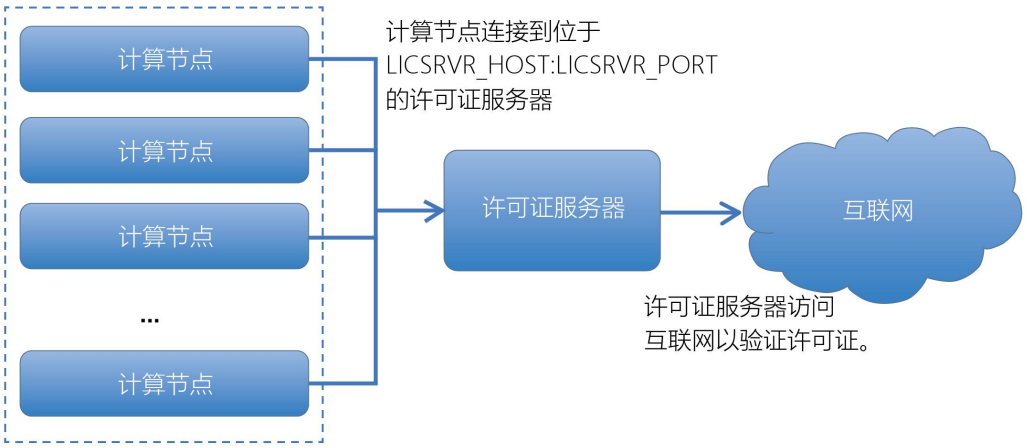

图 2-1　计算节点和许可证服务器的拓扑结构

您可能需要 IT 同事支持来获取 LICSRVR_HOST：LICSRVR_PORT，并确认满足上述要求。

注意：如果许可证服务器位于防火墙后面，通过 NAT 与计算节点分离，则节点可见的许可证服务器主机名/IP 可能与其实际主机名/IP 不同。如果是这种情况，您需要将许可证服务器绑定到实际 IP 地址，而计算节点从 NAT 后的 IP 地址请求许可证。请联系 sentieon@insvast.com。

按照以下步骤获取许可证文件，设置和测试许可证服务器：

1. 将以下信息发送给 sentieon@insvast.com 以接收许可证文件：

- 指定运行许可证服务的机器的 FQDN（Hostname）LICSRVR_HOST。
- 指定的端口 LICSRVR_PORT。

2. 将收到的许可证文件复制到许可证服务器 LICSRVR_HOST。我们假设许可证文件位于 LICENSE_PATH/LICENSE_FILE。在许可证服务器上运行以下命令以启动许可证服务器进程：

```
<SENTIEON_INSTALL_DIR>/bin/sentieon licsrvr --start --log LOG_FILE LICENSE_PATH/LICENSE_FILE
```

3. 或者，您可以按照“*2.3 设置许可为系统服务*” - 将许可证服务器（LICSRVR）作为系统服务运行的说明，将许可证服务器配置并启动为系统守护进程。

4. 进入 Sentieon®安装目录。在许可证服务器上运行以下命令，确认许可证服务器已启动并正在运行。

```
<SENTIEON_INSTALL_DIR>/bin/sentieon licclnt ping -s LICSRVR_HOST:LICSRVR_PORT
```

如果命令返回时没有错误消息，则许可证服务器已启动并正在运行。

5. 登录到其中一个计算节点，进入 Sentieon®安装目录，再次运行上述命令：

```
<SENTIEON_INSTALL_DIR>/bin/sentieon licclnt ping -s LICSRVR_HOST:LICSRVR_PORT
```

如果命令返回时没有错误消息，则计算节点现在也可以访问许可证服务器。

6. 设置以下环境变量，您就可以开始使用了。

```
export SENTIEON_LICENSE=LICSRVR_HOST:LICSRVR_PORT
```

2.1.5 系统环境要求

- 如果 Python 2.6.x、Python 2.7.x 或 python3.x 不是默认 Python 版本，您可以设置以下环境变量：

```
export SENTIEON_PYTHON=Python_location
```

- 如果您使用本地许可证文件，请设置以下环境变量，其中 LICENSE_DIR 是许可证文件所在的位置，LICENSE_FILE.lic 是许可证文件名：

```
export SENTIEON_LICENSE=LICENSE_DIR/LICENSE_FILE.lic
```

- 如果用户使用许可证服务器，请设置以下环境变量，其中 LICSRVR_HOST 和 LICSRVR_PORT 是许可证服务器的主机名和端口。详情请参阅下一节。

```
export SENTIEON_LICENSE=LICSRVR_HOST：LICSRVR_PORT
```

- 为方便起见，请如下所示设置二进制路径，其 PATH_TO_SENTIEON_BINARY_DIRECTORY 为 Sentieon®二进制文件的安装目录：

```
Export SENTIEON_INSTALL_DIR=PATH_TO_SENTIEON_BINARY_DIRECTORY
```

- 使用 NFS 存储时，为提高性能，请将 SENTIEON_TMPDIR 环境变量设置为指向本地快速临时存储：

```
export SENTIEON_TMPDIR=/tmp
```

2.2 首次运行作业

我们提供了一个快速入门包，其中包含示例脚本和数据，以帮助您快速测试安装并诊断潜在问题。

quikstart 演示项目链接：*https://ftp.insvast.com/sentieon_quickstart.tar.gz*

快速入门包包含单个染色体的数据，包括样本的序列数据和参考材料。作业脚本使用 Sentieon DNAscope 流程处理一组双端 Illumina fastq 文件:

- **BWA**：将读段比对到参考基因组。
- **Metrics、LocusCollector**：收集读段统计信息。
- **Dedup**：去除重复读段。
- **Variant calling**：DNAscope 变异检测。

注意： DNAscope 仅推荐用于二倍体生物的样本。对于其他样本，请使用 DNAseq。

2.2.1 运行快速入门包

要开始使用，请将下载的快速入门包复制到新目录，并通过运行以下命令解压：

```
tar xzvf sentieon_quickstart.tar.gz
```

包中包含以下内容：

- **sentieon_quickstart.sh**: 驱动整个流程的示例 shell 脚本。
- **reference**: 包含人类基因组参考文件和已知 SNP 位点数据库文件的目录。
- **models**: 包含 DNAscope 模型文件的目录。
- **FASTQ files**: 样本序列文件。

在运行脚本之前，您需要确保正确设置了上述环境变量，包括许可证和目录路径。

然后使用您喜欢的编辑器编辑 sentieon_quickstart.sh 中的用户设置。

```
# Update with the location of the Sentieon software package
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-202503.02

# Update with the location of temporary fast storage and uncomment
#SENTIEON_TMPDIR=/tmp

# It is important to assign meaningful names in actual cases.
# It is particularly important to assign different read group names.
sample="sample_name"
group="read_group_name"
platform="ILLUMINA"

# Other settings
nt=16 #number of threads to use in computation

# Is the data prepared with a PCR free library prep
PCRFREE=true
```

注意：在用户设置 shell 脚本 sentieon_quickstart.sh 中：

- 在实际情况下分配有意义的名称很重要。
- 特别重要的是要为不同的读组分配不同的名称。

要获取 CPU 核心数，用户可以运行 nproc，如下所示：

```
nproc
```

要更好地理解 sentieon_quickstart.sh 脚本的其余部分，请阅读每个部分的注释以及手册中的相应章节。

现在，只需运行 sentieon_quickstart.sh 来启动脚本，并观察结果展开。在典型的 Linux 服务器上，整个运行大约需要 3-5 分钟。实际时间取决于计算环境。

```
sh sentieon_quickstart.sh &
```

2.2.2 首次运行作业的输出文件

以下是文件列表及其对应的含义与参考说明。欲了解更多详情，请参阅相关文档。

文件名	描述
sorted.bam	使用 Sentieon® BWA mem 进行比对后，并按基因组坐标排序的 BAM 文件
score.txt	重复读段的数据文件
aln_metrics.txt	双端测序读段的比对指标及常规统计信息
gc_summary.txt	GC 偏好性统计摘要
gc_metrics.txt qc-report.pdf	GC 偏好性统计原始数据及 PDF 可视化报告
qd_metrics.txt qd-report.pdf	碱基质量分值分布数据文件及报告
mq_metrics.txt mq-report.pdf	平均质量分值随测序循环变化的数据文件及报告
is_metrics.txt is-report.pdf	插入片段大小分布的数据文件及报告
deduped.bam	去重阶段生成的输出 BAM 文件，已移除重复读段
dnascope.vcf.gz	DNAscope 变异检测生成的压缩格式 VCF 文件

2.3 设置许可为系统服务

2.3.1 使用 sysvinit 将许可证服务器作为系统服务运行

如果您的系统遵循传统的 System V 启动脚本，您可以通过以 root 身份运行以下命令来设置许可证服务器在系统中自动启动：

1. 创建并自定义配置文件；配置文件通常为 /etc/sysconfig/licsrvr；但在 Ubuntu 中，配置文件将是 /etc/default/licsrvr。以下是配置文件的示例，使用推荐的用户名 sentieon 设置：

- /home/sentieon/release/latest 是指向最新 Sentieon® 软件包安装目录的符号链接
- /home/sentieon/licsrvr 是运行 licsrvr 服务的文件夹
- /home/sentieon/licsrvr/licsrvr.lic 是 Sentieon® 许可证文件

```
licsrvr="/home/sentieon/release/latest/bin/sentieon licsrvr"
licfile=/home/sentieon/licsrvr/licsrvr.lic
logfile=/home/sentieon/licsrvr/licsrvr.log
```

2. 将许可证服务器启动脚本安装到 /etc/init.d 目录中。启动脚本包含在发布包中。

```
install -m 0755 $SENTIEON_INSTALL_DIR/doc/licsrvr.sh /etc/init.d/licsrvr
```

3. 安装并启用服务。根据您的系统，您将运行不同的命令：

- 如果您的系统安装了 Linux 标准基础核心规范，执行系统 init 脚本安装脚本。

```
/usr/lib/lsb/install_initd /etc/init.d/licsrvr
```

- 如果您的系统未安装 lsb 一致性包，使用 chkconfig 命令启用服务。

```
chkconfig --add licsrvr
chkconfig licsrvr on
```

-

- 对于 Ubuntu 和 Debian 系统，如果您没有 lsb/install_initd 二进制文件并选择不安装 lsb-core 包，使用 update-rc.d 命令安装并启用服务。

```
update-rc.d licsrvr defaults
update-rc.d licsrvr enable
```

4. 您可以使用 service 命令来启动/停止/重启/检查服务状态。

```
service licsrvr [start|stop|restart|status]
```

2.3.2 使用 systemd 将许可证服务器作为系统服务运行

您可以使用操作系统的 systemd 系统和服务功能来设置许可证服务器在系统中自动启动。为此，以 root 身份运行以下命令：

1. 如果您使用 Sentieon® 软件发布包的 doc 文件夹中的 licsrvr.service 许可证服务器启动脚本，您需要创建脚本所需的必要文件，包括使用用户名 sentieon：

- /home/sentieon/release/latest 是指向最新 Sentieon® 软件包安装目录的符号链接
- /home/sentieon/licsrvr 是运行 licsrvr 服务的文件夹
- /home/sentieon/licsrvr/licsrvr.lic 是 Sentieon® 许可证文件

或者，您可以编辑许可证服务器启动脚本以指向您特定的用户名和/或文件位置信息。

2. 将许可证服务器启动脚本安装到 /etc/systemd/system 目录中。

```
install -m 0644 $SENTIEON_INSTALL_DIR/doc/licsrvr.service /etc/systemd/system
```

3. 运行以下命令以启用计算机启动时自动启动许可证服务器：

```
systemctl enable licsrvr.service
```

4. 您可以使用 systemctl 命令手动启动和停止服务。

```
systemctl start licsrvr.service

systemctl stop licsrvr.service
```

3 Sentieon 工具集

3.1 Sentieon 工具列表

Sentieon®产品	Sentieon®工具	典型用途	等效 GATK 流程工具
Sentieon® BWA	Sentieon® BWA	读段比对和映射	BWA
DNAscope	DNAscope	改进的胚系 SNV/Indel/SV 检测	-
DNAseq®	Genotyper	胚系 SNV/Indel 检测，非单倍型基础	UnifiedGenotyper
DNAseq®	Haplotyper	胚系 SNV/Indel 检测	HaplotypeCaller
DNAseq®	GVCFtyper	群体联合检测，已验证可用于多达 200，000 个样本	GenotypeGVCFs
DNAseq®	VarCal	计算变异质量分数重校准	VariantRecalibrator
DNAseq®	ApplyVarCal	应用变异质量分数重校准	ApplyRecalibration
RNAseq	RNASplitReadsAtJunction	RNA SNV/Indel 检测	SplitNCigarReads
RNAseq	Haplotyper	RNA SNV/Indel 检测	HaplotypeCaller
TNseq®	TNsnv	体细胞 SNV 检测，非单倍型基础	MuTect
TNseq®	TNhaplotyper	体细胞 SNV/Indel 检测	MuTect2
TNseq®	TNhaplotyper2 + TNfilter	体细胞 SNV/Indel 检测	GATK4 中的 Mutect2 和 FilterMutectCalls
TNscope®	TNscope®	改进的体细胞 SNV/Indel/SV 检测	-
通用工具	Dedup 和 LocusCollector	执行去重复	Picard MarkDuplicates
通用工具	Realigner	为非单倍型基础的检测器执行 Indel 重比对	RealignerTargetCreator 和 IndelRealigner

通用工具	QualCal	执行碱基质量分数重校准	BaseRecalibrator、AnalyzeCovariates
通用工具	ReadWriter	创建 BAM 文件	PrintReads
通用工具	AlignmentStat	QC 指标	Picard CollectAlignmentSummaryMetrics
通用工具	BaseDistributionByCycle	QC 指标	Picard CollectBaseDistributionByCycle
通用工具	CollectVCMetrics	QC 指标	Picard CollectVariantCallingMetrics
通用工具	ContaminationAssessment	QC 指标	ContEst
通用工具	CoverageMetrics	QC 指标	DepthOfCoverage
通用工具	GCBias	QC 指标	Picard CollectGcBiasMetrics
通用工具	HsMetricAlgo	QC 指标	Picard CollectHsMetrics
通用工具	InsertSizeMetricAlgo	QC 指标	Picard Collect InsertSizeMetrics
通用工具	MeanQualityByCycle	QC 指标	Picard MeanQualityByCycle
通用工具	QualDistribution	QC 指标	Picard QualityScore Distribution
通用工具	QualityYield	QC 指标	Picard Collect QualityYieldMetrics
通用工具	SequenceArtifact MetricsAlgo	QC 指标	PicardCollectSequencing ArtifactMetric, ConvertSequencing ArtifactToOxoG
通用工具	WgsMetricsAlgo	QC 指标	Picard CollectWgsMetrics

4 Sentieon 典型工作流程

4.1 DNAseq®

Sentieon® Genomics 软件的一个典型用途是执行 Broad 研究所最佳实践中推荐的 DNA 分析生物信息学流程，详见 *https://www.broadinstitute.org/gatk/guide/best-practices* 。图 4-1 展示了这样一个典型的生物信息学流程。

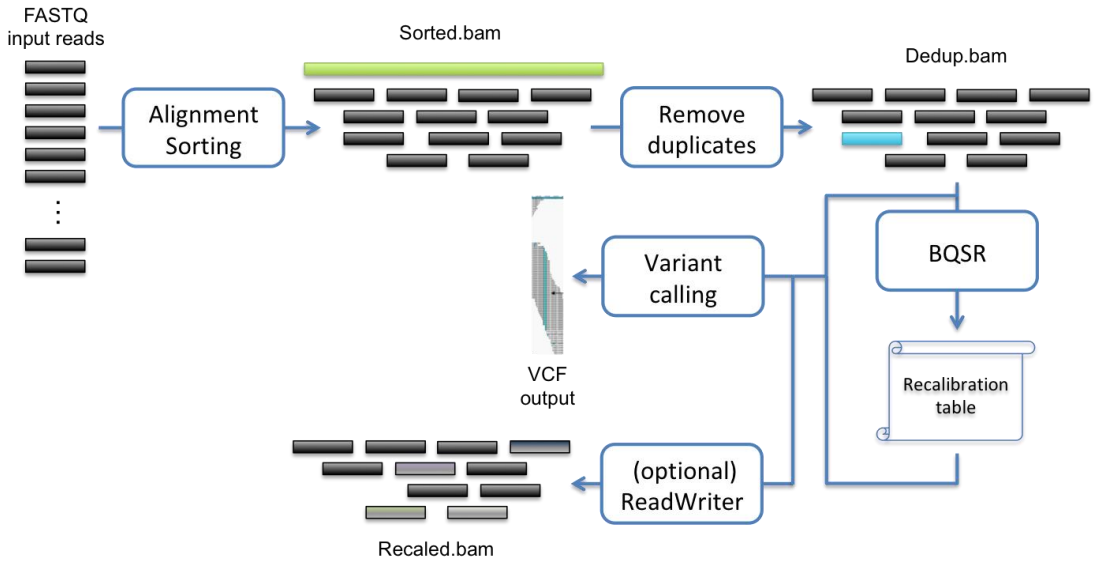

图 4-1 推荐用于 DNA 变异检测分析的生物信息学流程

4.1.1 概述

在这个生物信息学流程中，您需要以下输入：

- 包含与您将分析的样本相对应的参考基因组核苷酸序列的 FASTA 文件。参考数据需要预处理，以便软件可以使用表 4-1 中指定的数据。您可以参考"准备参考文件以供使用"的说明来生成所需的文件。

表 4-1 参考核苷酸序列的数据要求

数据	描述
.dict	序列字典文件
.fasta	参考序列文件
fasta.amb	BWA 索引文件
fasta.ann	BWA 索引文件
.fasta.bwt	BWA 索引文件
.fasta.fai	索引文件
.fasta.pac	BWA 索引文件
.fasta.sa	BWA 索引文件

- 一个或多个包含待分析样本核苷酸序列的 FASTQ 文件。这些文件包含来自 DNA 测序的原始读数。软件支持输入使用 GZIP 压缩的 FASTQ 文件。软件仅支持包含 Sanger 格式（Phred+33）质量分数的文件。
- （可选）您想在流程中包含的单核苷酸多态性数据库（dbSNP）数据。数据以 VCF 文件的形式使用；您可以使用用 bgzip 压缩并索引的 VCF 文件。
- （可选）您想在流程中包含的多个已知位点集合。数据以 VCF 文件的形式使用；您可以使用用 bgzip 压缩并索引的 VCF 文件。

典型的生物信息学流程包括以下步骤：

1. 将读数映射到参考：此步骤将 FASTQ 文件中包含的读数与 FASTA 文件中包含的参考基因组对齐。这一步确保数据可以放在上下文中。
2. 计算数据指标：此步骤生成数据质量和流程数据分析质量的统计摘要。

3. 去除或标记重复：此步骤检测表明同一 DNA 分子被测序多次的读数。这些重复不具有信息性，不应被视为额外的证据。

4. （可选）碱基质量分数重校准（BQSR）：此步骤修改分配给序列读数数据的单个读数碱基的质量分数。这一操作消除了测序方法造成的实验偏差。

5. 变异检测：此步骤识别您的数据相对于参考基因组显示变异的位点，并计算每个样本在该位点的基因型。

4.1.2 DNAseq® 使用步骤

（1）将读数映射到参考

运行单个命令以高效执行使用 BWA 的比对，以及使用 Sentieon®软件创建 BAM 文件和排序：

```
(sentieon bwa mem -R '@RG\tID:GROUP_NAME\tSM:SAMPLE_NAME\tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE FASTQ [FASTQ2] || echo -n 'error' ) \
  | sentieon util sort -r REFERENCE -o SORTED_BAM -t NUMBER_THREADS --sam2bam -i
-
```

BWA 的输入和选项在手册中有描述。

或者，您可以使用其他能在 stdout 中生成遵循 SAM 格式的文件的比对器，并替换命令中的 BWA 部分。

该命令需要以下输入：

- **GROUP_NAME**：将添加到读组头行的读组标识符。RG：ID 在所有您计划使用的数据集中需要是唯一的，这在处理多个输入文件或进行肿瘤-正常分析时尤为重要。
- **SAMPLE_NAME**：将添加到读组头行的样本名称。

- **PLATFORM**：用于测序 DNA 的测序平台名称。可能的选项有：ILLUMINA（当 fastq 文件在 Illumina™机器上产生时）；IONTORRENT（当 fastq 文件在 Life Technologies™ Ion-Torrent™机器上产生时）；ELEMENT（当 fastq 文件在 Element Biosciences™机器上产生时）；DNBSEQ（当 fastq 文件在 MGI™机器上产生时）；ULTIMA（当 fastq 文件在 Ultima Genomics™机器上产生时）。
- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。我们建议对 BWA 和 util 二进制文件使用相同数量的线程。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保表 4-1 中指定的所有额外参考数据都在相同位置，并具有一致的命名。
- **FASTQ**：样本 FASTQ 文件的位置。如果数据来自双端测序技术，您还需要输入 FASTQ2 作为相应的配对样本 FASTQ 文件。
- **SORTED_BAM**：排序后的映射 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。

BWA 会根据命令中使用的线程数产生略微不同的结果。这是因为 BWA 在一个块上计算插入大小分布，而块的大小取决于线程数。为了保证结果与使用的线程数无关，您应该使用选项-K 10000000 来固定块大小（以碱基为单位）。

（2）计算数据指标

运行单个命令以生成 5 个数据质量和流程数据分析质量结果的统计摘要：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SORTED_BAM \
  --algo GCBias --summary GC_SUMMARY_TXT GC_METRIC_TXT \
  --algo MeanQualityByCycle MQ_METRIC_TXT \
  --algo QualDistribution QD_METRIC_TXT \
```

```
  --algo  InsertSizeMetricAlgo  IS_METRIC_TXT  \
  --algo  AlignmentStat  ALN_METRIC_TXT
```

运行四个命令以从统计摘要生成图表：

```
sentieon  plot  GCBias  -o  GC_METRIC_PDF  GC_METRIC_TXT
sentieon  plot  MeanQualityByCycle  -o  MQ_METRIC_PDF  MQ_METRIC_TXT
sentieon  plot  QualDistribution  -o  QD_METRIC_PDF  QD_METRIC_TXT
sentieon  plot  InsertSizeMetricAlgo  -o  IS_METRIC_PDF  IS_METRIC_TXT
```

这些命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考基因组 FASTA 文件的位置。您应确保参考与映射阶段使用的相同。
- **SORTED_BAM**：前一个映射阶段存储结果的位置。
- **GC_SUMMARY_TXT**：GC 偏差指标摘要结果输出文件的位置和文件名
- **GC_METRIC_TXT**：GC 偏差指标结果输出文件的位置和文件名
- **MQ_METRIC_TXT**：映射质量指标结果输出文件的位置和文件名。
- **QD_METRIC_TXT**：质量/深度指标结果输出文件的位置和文件名。
- **IS_METRIC_TXT**：插入大小指标结果输出文件的位置和文件名。
- **ALN_METRIC_TXT**：比对指标结果输出文件的位置和文件名。
- **GC_METRIC_PDF**：GC 偏差指标报告输出文件的位置和文件名。
- **MQ_METRIC_PDF**：映射质量指标报告输出文件的位置和文件名。
- **QD_METRIC_PDF**：质量/深度指标报告输出文件的位置和文件名。
- **IS_METRIC_PDF**：插入大小指标报告输出文件的位置和文件名。

（3）去除或标记重复

在比对和排序后，运行两个单独的命令来移除或标记 BAM 文件中的重复。第一个命令收集读数信息，第二个命令执行去重；--rmdup 选项控制是移除重复的读数（如果存在该选项）还是仅将其标记为重复。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz \
  --metrics DEDUP_METRIC_TXT DEDUPED_BAM
```

这些命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **SORTED_BAM**：前一个映射阶段存储结果的位置。
- **SCORE.gz**：临时得分输出文件的位置和文件名。确保两个命令使用相同的文件。
- **DEDUP_METRICS_TXT**：去重指标结果输出文件的位置和文件名。
- **DEDUPED_BAM**：去重 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。

（4）碱基质量分数重校准（BQSR；可选）

运行单个命令来计算对序列读数数据中分配给单个读数碱基的质量分数所需的修改；实际的重校准在变异检测阶段应用。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE \
  -i DEDUPED_BAM --algo QualCal [-k KNOWN_SITES] RECAL_DATA.TABLE
```

运行三个命令来应用重校准并创建碱基质量分数重校准的报告。第一个命令应用重校准以计算校准后的数据表，并额外对 BAM 文件应用重校准；第二个命令创

建用于绘图的数据；第三个命令将校准前后的数据表绘制成 PDF 中的图表。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
  -q RECAL_DATA.TABLE --algo QualCal [-k KNOWN_SITES] \
  RECAL_DATA.TABLE.POST [--algo ReadWriter RECALIBRATED_BAM]
sentieon driver -t NUMBER_THREADS --algo QualCal --plot \
  --before RECAL_DATA.TABLE --after RECAL_DATA.TABLE.POST RECAL_RESULT.CSV
sentieon plot QualCal -o BQSR_PDF RECAL_RESULT.CSV
```

这些命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的相同。
- **DEDUPED_BAM**：前一个去重阶段存储结果的位置。
- **RECAL_DATA.TABLE**：重校准表的位置和文件名。
- **RECAL_DATA.TABLE.POST**：临时后重校准表的位置和文件名。
- **RECAL_RESULT.CSV**：用于绘图的临时重校准结果输出文件的位置和文件名。
- **BQSR_PDF**：BSQR 结果输出文件的位置和文件名。

以下输入是可选的：

- **KNOWN_SITES**：用作已知位点集合的 VCF 文件的位置。可以通过重复-k KNOWN_SITES 选项来包含多个已知位点集合。
- **RECALIBRATED_BAM**：重校准 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。这个输出是可选的，因为 Sentieon®变异检测器可以使用重校准前的 BAM 加上重校准表来即时执行重校准。

（5）变异检测

运行单个命令来检测变异并额外应用之前计算的 BQSR。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
  -q RECAL_DATA.TABLE --algo Haplotyper [-d dbSNP] VARIANT_VCF
```

您可能只想重新运行变异检测，例如使用 Unified Genotyper 变异检测算法。

在这种情况下，您不需要重新应用 BQSR，可以使用之前重校准的 BAM：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i RECALIBRATED_BAM \
  --algo Genotyper [-d dbSNP] VARIANT_VCF
```

在这两种情况下，使用重校准的 BAM 或重校准前的 BAM 加上重校准数据表将给出相同的结果；但是，您应该小心不要将重校准数据表与已经重校准的 BAM 一起使用，因为这会导致重校准被应用两次，从而导致不正确的结果。

这些命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的相同。
- **DEDUPED_BAM**：前一个去重阶段存储结果的位置。
- **RECAL_DATA.TABLE**：前一个 BQSR 阶段存储结果的位置。
- **RECALIBRATED_BAM**：重校准 BAM 文件的位置。
- **VARIANT_VCF**：变异检测输出文件的位置和文件名。将创建相应的索引文件。工具将通过使用.gz 扩展名输出压缩文件。

以下输入是可选的：

- **dbSNP**：将用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。您只能使用一个 dbSNP 文件。

4.1.3 DNAseq® CCDG 标准分析脚本

以下是遵循 CCDG 标准化流程的 DNA 测序的 Sentieon DNAseq 分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ********************************************************************************
*****
# Script to perform DNA seq variant calling using Sentieon following
# the functional equivalent pipeline described in
# https://github.com/CCDG/Pipeline-Standardization/blob/master/PipelineStandard.md
# ********************************************************************************
*****

set -eu

# Update with the fullpath location of your sample FASTQ
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_1="${SM}_r1.fastq.gz"
FASTQ_2="${SM}_r2.fastq.gz" #if using 2 FASTQ inputs

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/hg38bundle"
FASTA="$FASTA_DIR/Homo_sapiens_assembly38.fasta"
KNOWN_DBSNP="$FASTA_DIR/Homo_sapiens_assembly38.dbsnp138.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/Homo_sapiens_assembly38.known_indels.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.hg38.vcf.gz"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
NT=$(nproc) #number of threads to use in computation
SAMBLASTER=/home/release/other_tools/samblaster-0.1.23/samblaster
START_DIR="$PWD/test/CCDG" #Determine where the output files will be stored
```

```
# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping BWA-MEM 0.7.15 util sort
# ****************************************
SENTIEON_VERSION=$($SENTIEON_INSTALL_DIR/bin/sentieon driver --version)
if (( $(echo "${SENTIEON_VERSION##*-} < 201911" |bc -l) )); then
        ( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\t
PL:$PL" -t $NT \
            -K 100000000 -Y $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'bwa error'; exi
t 1; } ) | \
            ( $SAMBLASTER --addMateTags -a || { echo -n 'samblaster error'; exit 1; }
 ) | \
        $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT -
-sam2bam -i -
else
        #Sentieon 201911 and higher use BWA 0.7.17, which already produce MC tags
 in the output
        ( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\t
PL:$PL" -t $NT \
            -K 100000000 -Y $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'error'; exit 1; }
 ) | \
        $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT -
-sam2bam -i -
fi
```

```
# ****************************************
# 2. Mark Duplicates with Sentieon
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup --score_inf
o score.txt \
    --metrics mark_dup_metrics.txt --output_dup_read_name tmp_dup_qname.txt || \
    { echo "Dedup1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --dup_read_name tmp_dup_qname.txt markduped.bam || { echo "Dedup2 failed"; exi
t 1; }

# ****************************************
# 3. Base Quality Score Recalibration with Sentieon
# ****************************************
interval_arg="--interval chr1,chr2,chr3,chr4,chr5,chr6,chr7,chr8,chr9,chr10,chr11,\
chr12,chr13,chr14,chr15,chr16,chr17,chr18,chr19,chr20,chr21,chr22"
$SENTIEON_INSTALL_DIR/bin/sentieon driver $interval_arg -r $FASTA -t $NT -i markduped.
bam \
    --algo QualCal -k $KNOWN_MILLS -k $KNOWN_INDELS -k $KNOWN_DBSNP recal_data.
table || \
    { echo "QualCal failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i markduped.bam \
    --read_filter QualCalFilter,table=recal_data.table,prior=-1.0,indel=false,levels=10/20/30,min
_qual=6 \
    --algo ReadWriter recaled_RW.cram || { echo "ReadWriter failed"; exit 1; }

# ****************************************
# 4. Haplotyper with Sentieon
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i recaled_RW.cram --algo
Haplotyper \
    Haplotyper.vcf.gz || { echo "Haplotyper failed"; exit 1; }
```

4.1.4 DNAseq® 多样本 FASTQ 分析脚本

以下是 Sentieon DNAseq 用于处理单个样本多组 fastq 文件的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform DNA seq variant calling
# using a single sample with more than one
# set of input fastq files (in this example
# named set1_1.fastq.gz, set2_1.fastq.gz
# set3_1.fastq.gz and set4_1.fastq.gz)
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID_PREFIX="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
NUM_SETS="4"
FASTQ_PREFIX="set"
FASTQ_SUFFIX_1="_1.fastq.gz"
FASTQ_SUFFIX_2="_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/DNAseq_multiFASTQ" #Determine where the output files will be s
```

```
tored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping each set of input fastq with BWA-MEM, sorting
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
BAM_INPUT=""
for i in $(seq 1 $NUM_SETS); do
 BAM_INPUT="$BAM_INPUT -i sorted_set$i.bam"
 ( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem \
      -R "@RG\tID:${RGID_PREFIX}_$i\tSM:$SM\tPL:$PL" -t $NT -K 10000000 $FASTA \
      $FASTQ_FOLDER/$FASTQ_PREFIX$i$FASTQ_SUFFIX_1 \
      $FASTQ_FOLDER/$FASTQ_PREFIX$i$FASTQ_SUFFIX_2 || { echo -n 'bwa error'; exit 1;
 } ) | \
      $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted_set$i.bam \
      -t $NT --sam2bam -i - || { echo "Alignment failed"; exit 1; }
done

# ****************************************
# 2. Metrics on the multiple sorted BAM files
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT $BAM_INPUT \
```

```
    --algo MeanQualityByCycle mq_metrics.txt --algo QualDistribution qd_metrics.txt \
    --algo GCBias --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
    { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metri
cs.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.t
xt

# ****************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT $BAM_INPUT --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT $BAM_INPUT --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 2a. Coverage metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    --algo CoverageMetrics coverage_metrics || { echo "CoverageMetrics failed"; exit 1;
 }

# ****************************************
# 5. Base recalibration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam --algo Qual
Cal \
    -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_data.table
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_da
ta.table \
    --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_dat
a.table.post
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT --algo QualCal --plot \
    --before recal_data.table --after recal_data.table.post recal.csv
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualCal -o recal_plots.pdf recal.csv
```

```
# ****************************************
# 6b. HC Variant caller
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_data.table \
    --algo Haplotyper -d $KNOWN_DBSNP output-hc.vcf.gz || { echo "Haplotyper failed"; exit 1; }
```

4.1.5 DNAseq® 全外显子测序分析脚本

以下是 Sentieon DNAseq 在全外显子测序（WES）中的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform DNA seq variant calling
# using an exome sample with fastq files
# named 1.fastq.gz and 2.fastq.gz
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1="$FASTQ_FOLDER/1.fastq.gz"
FASTQ_2="$FASTQ_FOLDER/2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
INTERVAL_FILE="$FASTA_DIR/TruSeq_exome_targeted_regions.b37.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the server
START_DIR="$PWD/test/DNAseq_interval" #Determine where the output files will be stored
```

```
# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
**********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping reads with BWA-MEM, sorting
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'BWA error'; exit 1; }
 ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT \
     --sam2bam -i - || { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
     ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i sorted.bam --algo MeanQualityByCycle
 \
     mq_metrics.txt --algo QualDistribution qd_metrics.txt --algo GCBias \
     --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat --adapter_seq '' \
     aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
     { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metri
```

```
cs.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.t
xt

# ****************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 2a. Coverage metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i deduped.bam --algo CoverageMetrics \
    coverage_metrics || { echo "CoverageMetrics failed"; exit 1; }

# ****************************************
# 5. Base recalibration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i deduped.bam --algo QualCal \
    -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_data.table
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i deduped.bam -q recal_data.table \
    --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_dat
a.table.post
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT --algo QualCal --plot \
    --before recal_data.table --after recal_data.table.post recal.csv
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualCal -o recal_plots.pdf recal.csv

# ****************************************
# 6b. HC Variant caller
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -t $NT -i deduped.bam -q recal_data.tabl
e \
    --algo Haplotyper -d $KNOWN_DBSNP --emit_conf=30 --call_conf=30 output-hc.vcf.gz
```

```
|| \
    { echo "Haplotyper failed"; exit 1; }

# ****************************************
# 5b. ReadWriter to output recalibrated bam
# This stage is optional as variant callers
# can perform the recalibration on the fly
# using the before recalibration bam plus
# the recalibration table
# This stage should not include interval
# option to prevent read loss
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    -q recal_data.table --algo ReadWriter recaled.bam || \
    { echo "ReadWriter failed"; exit 1; }
```

4.1.6 DNAseq® 全基因组测序分析脚本

以下是 Sentieon DNAseq 在全基因组测序（WGS）中的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform DNA seq variant calling
# using a single sample with fastq files
# named 1.fastq.gz and 2.fastq.gz
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1="$FASTQ_FOLDER/1.fastq.gz"
FASTQ_2="$FASTQ_FOLDER/2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/DNAseq" #Determine where the output files will be stored
```

```
# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping reads with BWA-MEM, sorting
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'BWA error'; exit 1; }
 ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT \
     --sam2bam -i - || { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i sorted.bam \
     --algo MeanQualityByCycle mq_metrics.txt --algo QualDistribution qd_metrics.txt \
     --algo GCBias --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
     { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metri
cs.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.t
xt
```

```
# ****************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 5. Base recalibration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam --algo QualCal \
    -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_data.table
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_data.table \
    --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_data.table.post
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT --algo QualCal --plot \
    --before recal_data.table --after recal_data.table.post recal.csv
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualCal -o recal_plots.pdf recal.csv

# ****************************************
# 6b. HC Variant caller
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_data.table \
    --algo Haplotyper -d $KNOWN_DBSNP output-hc.vcf.gz || \
    { echo "Haplotyper failed"; exit 1; }

# ****************************************
# 5b. ReadWriter to output recalibrated bam
# This stage is optional as variant callers
# can perform the recalibration on the fly
# using the before recalibration bam plus
# the recalibration table
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_data.table \
    --algo ReadWriter recaled.bam || { echo "ReadWriter failed"; exit 1; }
```

4.2 DNAscope

Sentieon® Genomics 软件包含一个改进的算法来执行胚系 DNA 分析的变异检测步骤。DNAscope 使用的流程类似于 DNAseq®中描述的流程，但在比对和变异检测方面都有所不同。DNAscope 接受模型文件以提高处理速度和准确性，除了检测 SNP 和小 indel 外，它还可以进行结构变异检测。DNAscope 推荐用于人类或其他哺乳动物样本的测序数据集。

4.2.1 概述

在这个生物信息学流程中，您需要以下输入：

- 包含与您将分析的样本相对应的参考基因组核苷酸序列的 FASTA 文件。
- 一个或多个包含要分析样本的核苷酸序列的 FASTQ 文件。这些文件包含来自 DNA 测序的原始读数。该软件支持输入使用 GZIP 压缩的 FASTQ 文件。该软件仅支持包含 Sanger 格式（Phred+33）质量分数的文件。
- 可从 *https：//github.com/Sentieon/sentieon-models* 访问的特定测序平台机器学习模型文件。
- （可选）包含变异检测区间的 BED 文件。推荐用于全外显子组或靶向测序数据。
- （可选）您想在流程中包含的单核苷酸多态性数据库（dbSNP）数据。数据以 VCF 文件的形式使用；您可以使用 bgzip 压缩并索引的 VCF 文件。

DNAscope 的典型生物信息学流程包括以下步骤：

1. 将读数映射到参考：这一步将 FASTQ 文件中包含的读数映射到 FASTA 文件中包含的参考基因组。这一步确保数据可以放在上下文中。

2. 计算数据指标：这一步生成数据质量和流程数据分析质量的统计摘要。

3. 去除或标记重复：这一步检测表明同一 DNA 分子被测序多次的 reads。这些重复不具有信息性，不应被计为额外的证据。

4. 使用带机器学习模型的 DNAscope 进行变异检测：这一步识别您的数据相对于参考基因组显示变异的位点，并计算每个样本在该位点的基因型。

4.2.2 DNAscope 使用步骤

4.2.2.1 将读段映射到参考基因组

运行单个命令以使用 BWA 高效执行比对，并使用 Sentieon®软件创建 BAM 文件和排序：

```
(sentieon bwa mem -R '@RG\tID:GROUP_NAME\tSM:SAMPLE_NAME\tPL:PLATFORM' \
  -t NUMBER_THREADS -x DNASCOPE_MODEL/bwa.model REFERENCE FASTQ [FASTQ2] \
  || echo -n 'error' ) \
  | sentieon util sort -r REFERENCE -o SORTED_BAM -t NUMBER_THREADS --sam2bam -i
-
```

BWA 的输入和选项在手册中有描述。

与 DNAseq®中描述的 BWA 用法相比，添加了 DNASCOPE_MODEL，参数为 -x DNASCOPE_MODEL/bwa.model

该命令需要以下输入：

- **GROUP_NAME**：将添加到读组头行的读组标识符。RG：ID 在您计划使用的所有数据集中需要是唯一的，这在处理多个输入文件或进行肿瘤-正常分析时很重要。
- **SAMPLE_NAME**：将添加到读组头行的样本名称。

- **PLATFORM**：用于测序 DNA 的测序平台名称。可能的选项有：当 fastq 文件在 Illumina™机器上产生时为 ILLUMINA；当 fastq 文件在 Element Biosciences™机器上产生时为 ELEMENT；当 fastq 文件在 MGI™机器上产生时为 DNBSEQ；当 fastq 文件在 Ultima Genomics™机器上产生时为 ULTIMA。
- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。我们建议对 BWA 和 util 二进制文件使用相同数量的线程。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保表 4-1 中指定的所有额外参考数据都在相同位置，并具有一致的命名。
- **FASTQ**：样本 FASTQ 文件的位置。如果数据来自双端测序技术，您还需要输入 FASTQ2 作为相应的配对样本 FASTQ 文件。
- **SORTED_BAM**：排序后的映射 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。
- **DNASCOPE_MODEL**：DNAscope 模型包的位置。该模型将用于确定比对和变异检测中使用的设置。

BWA 会根据命令中使用的线程数产生略微不同的结果。这是因为 BWA 在一个块上计算插入大小分布，而块的大小取决于线程数。为了保证结果与使用的线程数无关，您应该使用选项-K 10000000 固定块大小（以碱基为单位）。

对于指标收集和重复去除阶段，请参考 *第 4.1.2 节* 的获取详细的使用说明。

4.2.2.2 使用机器学习模型进行胚系变异检测

建议使用带机器学习模型的 DNAscope 来执行变异检测，通过改进候选检测和过滤来提高准确性。

Sentieon®可以为您提供特定测序平台的模型，该模型使用 *https://github.com/genome-in-a-bottle* 中 GiAB 真实集的部分数据进行训练。这些模型是通过将参考样本 HG001-HG007 处理通过由 Sentieon® BWA-mem 比对和 Sentieon®去重组成的流程创建的，并使用变异检测结果来校准模型以适应真实集。

此外，Sentieon®可以协助您使用自己的数据创建模型，这将校准您的测序和生物信息学处理的具体情况。

（1）在 DNAscope 中使用机器学习模型

运行两个单独的命令来检测变异并应用机器学习模型。输入的 BAM 文件应该来自只执行了比对和去重（没有 BQSR 或 indel 重新比对）的流程，以匹配模型创建方法。

```
PCRFREE=true #PCRFREE=true means the sample is PCRFree, change it to false for PCR samples.
if [ "$PCRFREE" = true ] ; then
    sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
    [--interval INTERVAL_FILE] --algo DNAscope [-d dbSNP] \
    --pcr_indel_model none --model DNASCOPE_MODEL/dnascope.model \
    TMP_VARIANT_VCF
else
    sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
    [--interval INTERVAL_FILE] --algo DNAscope [-d dbSNP] \
    --model DNASCOPE_MODEL/dnascope.model TMP_VARIANT_VCF
fi
sentieon driver -t NUMBER_THREADS -r REFERENCE --algo DNAModelApply \
  --model DNASCOPE_MODEL/dnascope.model -v TMP_VARIANT_VCF VARIANT_VCF
```

提醒： 如果您使用的数据是不含 PCR 的，在运行 DNAscope 时添加 --pcr_indel_model NONEDNAscope 选项很重要。

根据是否涉及 PCR，DNAscope 使用不同的先验来查找显著的 INDEL 变异，这可以通过--pcr_indel_model 选项来控制。默认设置是针对 PCR 样本的。因此，对于 PCR-free 样本，设置--pcr_indel_model NONE 很重要。

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参考相同。
- **DEDUPED_BAM**：输入 BAM 文件的位置。
- **TMP_VARIANT_VCF**：DNAscope 变异检测输出的位置和文件名。这是一个临时文件。
- **VARIANT_VCF**：变异检测输出的位置和文件名。将创建相应的索引文件。该工具将通过使用.gz 扩展名输出压缩文件。
- **DNASCOPE_MODEL**：机器学习模型文件的位置。在 DNAscope 命令中，该模型将用于确定比对和变异检测中使用的设置。

提醒： 使用机器学习模型运行时，大多数高级设置由模型本身决定，将 --pcr_indel_model 选项以外的选项设置为特定值可能会对结果产生负面影响。

该命令的以下输入是可选的：

- **INTERVAL_FILE**：BED 文件的位置。

- **dbSNP**：将用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。您只能使用一个 dbSNP 文件。

（2）使用 DNAscope 生成 GVCF 输出文件

从 202112.04 版本开始，DNAscope 接受模型以 Genomic VCF （GVCF）格式生成变异检测。GVCF 格式包含正在处理的样本中对参考等位基因纯合的位点的额外信息。需要最近训练的 DNAscope 模型，使用 Sentieon 202112.01 或更早版本训练的 DNAscope 模型将导致错误。

运行两个单独的命令来检测变异并应用机器学习模型。输入的 BAM 文件应该来自只执行了比对和去重的流程，以匹配模型创建方法。

```
PCRFREE=true #PCRFREE=true means the sample is PCRFree, change it to false for PCR samples.
if [ "$PCRFREE" = true ] ; then
      sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
      [--interval INTERVAL_FILE] --algo DNAscope [-d dbSNP] \
      --pcr_indel_model none --model DNASCOPE_MODEL/dnascope.model \
      --emit_mode gvcf TMP_VARIANT_GVCF
else
      sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
      [--interval INTERVAL_FILE] --algo DNAscope [-d dbSNP] \
      --model DNASCOPE_MODEL/dnascope.model --emit_mode gvcf TMP_VARIANT_GVCF
fi
sentieon driver -t NUMBER_THREADS -r REFERENCE --algo DNAModelApply \
  --model DNASCOPE_MODEL/dnascope.model -v TMP_VARIANT_GVCF VARIANT_GVCF
```

提醒：如果您使用的数据是不含 PCR 的，在运行 DNAscope 时添加 --pcr_indel_model NONE 选项很重要。

根据是否涉及 PCR，DNAscope 使用不同的先验来查找显著的 INDEL 变异，这可以通过--pcr_indel_model 选项控制。默认的--pcr_indel_model 设置是针对 PCR 样本的。因此，对于 PCR-free 样本，将--pcr_indel_model 设置为 NONE 很重要。

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参考相同。
- **DEDUPED_BAM**：输入 BAM 文件的位置。
- **TMP_VARIANT_GVCF**：DNAscope GVCF 输出的位置和文件名。这是一个临时文件。
- **VARIANT_GVCF**：GVCF 输出的位置和文件名。将创建相应的索引文件。该工具将通过使用.gz 扩展名输出压缩文件。
- **DNASCOPE_MODEL**：机器学习模型文件的位置。在 DNAscope 命令中，该模型将用于确定变异检测中使用的设置。

该命令的以下输入是可选的：

- **INTERVAL_FILE**：BED 文件的位置。
- **dbSNP**：将用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。您只能使用一个 dbSNP 文件。

（3）对 DNAscope 生成的 GVCF 文件进行基因型分析

GVCF 输出文件可以单独进行基因型分析，或者在 Sentieon 202112.06 及更高版本中使用 GVCFtyper 算法与其他样本的 GVCF 联合进行基因型分析，并输出单样本或多样本 VCF。

```
sentieon driver -r REFERENCE --algo GVCFtyper \
  -v s1_VARIANT_GVCF -v s2_VARIANT_GVCF -v s3_VARIANT_GVCF VARIANT_VCF
```

请查看 Sentieon 手册*第五章节：5.1.2 DRIVER 算法中（22）GVCFtyper*以获取有关 GVCFtyper 算法的更多详细信息

提醒：GVCFtyper 可用于将来自多个测序平台的 DNAscope GVCF 基因型分析为单个多样本 VCF。

GVCFtyper 不支持将 DNAscope GVCF 与不使用机器学习模型的 DNAscope 生成的 GVCF 或其他工具生成的 GVCF 混合进行联合基因型分析。

4.2.2.3 结构变异检测

为了执行结构变异检测，您需要在 DNAscope 命令中添加选项以输出断点（BND）信息；这是通过启用 bnd（断点）变异类型来完成的。运行两个单独的命令来执行结构变异检测。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
  --algo DNAscope --var_type bnd \
  [-d dbSNP] TMP_VARIANT_VCF
sentieon driver -t NUMBER_THREADS -r REFERENCE --algo SVSolver  \
  -v TMP_VARIANT_VCF STRUCTURAL_VARIANT_VCF
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参考相同。
- **DEDUPED_BAM**：输入 BAM 文件的位置。
- **TMP_VARIANT_VCF**：DNAscope 变异检测输出文件的位置和文件名，包括 BND 信息。在检测结构变异时，这是一个临时文件。
- **STRUCTURAL_VARIANT_VCF**：包含结构变异的变异检测输出文件的位置

和文件名。将创建相应的索引文件。该工具将通过使用.gz 扩展名输出压缩文件。

该命令的以下输入是可选的：

- **dbSNP**：将用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。您只能使用一个 dbSNP 文件。

请注意，结构变异检测与 DNAscope 模型输入文件不兼容。特别是，当设置了--var_type BND 时，应避免使用--model 选项。

4.2.3 DNAscope 全基因组测序分析脚本

以下是 Sentieon DNAscope 在全基因组测序（WGS）中的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2023 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform DNAscope variant calling
# using a single sample with fastq files
# named 1.fastq.gz and 2.fastq.gz
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1="$FASTQ_FOLDER/1.fastq.gz"
FASTQ_2="$FASTQ_FOLDER/2.fastq.gz" #If using Illumina paired data

# Update with the location to the DNAscope model file
# Model files can be found at, https://github.com/Sentieon/sentieon-models
DNASCOPE_MODEL=/home/pipeline/models/DNAscopeIlluminaWGS2.0.bundle

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
PCRFREE=true # The data was sequenced with a PCR-free library prep
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
```

```
START_DIR="$PWD/test/DNAscope" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping reads with BWA-MEM, sorting
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\tPL:$PL" \
     -t $NT -K 10000000 -x $DNASCOPE_MODEL/bwa.model \
     $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT \
     --sam2bam -i - || { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i sorted.bam \
     --algo MeanQualityByCycle mq_metrics.txt --algo QualDistribution qd_metrics.txt \
     --algo GCBias --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
     { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
```

```
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.txt

# ****************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 4a. DNAscope variant calling
# ****************************************
indel_model_arg=""
if [ "$PCRFREE" = true ]; then
    indel_model_arg="--pcr_indel_model none"
fi
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    --algo DNAscope $indel_model_arg --model $DNASCOPE_MODEL/dnascope.model \
    -d $KNOWN_DBSNP output-ds_tmp.vcf.gz || \
    { echo "DNAscope failed"; exit 1; }

# ****************************************
# 4b. Variant filtering and genotyping
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    --algo DNAModelApply --model $DNASCOPE_MODEL/dnascope.model \
    -v output-ds_tmp.vcf.gz output-ds.vcf.gz || \
    { echo "DNAModelApply failed"; exit 1; }
```

4.2.4 DNAscope 外显子测序分析脚本

以下是 Sentieon DNAscope 在全外显子测序（WES）中的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2023 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform DNAscope variant calling
# using an exome sample with fastq files
# named 1.fastq.gz and 2.fastq.gz
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1="$FASTQ_FOLDER/1.fastq.gz"
FASTQ_2="$FASTQ_FOLDER/2.fastq.gz" #If using Illumina paired data

# Update with the location to the DNAscope model file
# Model files can be found at, https://github.com/Sentieon/sentieon-models
DNASCOPE_MODEL=/home/pipeline/models/DNAscopeIlluminaWES2.0.bundle

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
INTERVAL_FILE="$FASTA_DIR/TruSeq_exome_targeted_regions.b37.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
PCRFREE=true # The data was sequenced with a PCR-free library prep
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
```

```
 server
START_DIR="$PWD/test/DNAscope" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping reads with BWA-MEM, sorting
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\tPL:$PL" \
     -t $NT -K 10000000 -x $DNASCOPE_MODEL/bwa.model \
     $FASTA $FASTQ_1 $FASTQ_2 || { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT \
     --sam2bam -i - || { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Metrics
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
     ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i sorted.bam --algo MeanQualityByCycle
 \
     mq_metrics.txt --algo QualDistribution qd_metrics.txt --algo GCBias \
     --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat --adapter_seq '' \
     aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
     { echo "Metrics failed"; exit 1; }
```

```
$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metri
cs.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.t
xt

# ****************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 4a. DNAscope variant calling
# ****************************************
indel_model_arg=""
if [ "$PCRFREE" = true ]; then
    indel_model_arg="--pcr_indel_model none"
fi
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} --algo DNAscope $indel_model_arg \
    --model $DNASCOPE_MODEL/dnascope.model -d $KNOWN_DBSNP output-ds_tmp.vcf.g
z || \
    { echo "DNAscope failed"; exit 1; }

# ****************************************
# 4b. Variant filtering and genotyping
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    --algo DNAModelApply --model $DNASCOPE_MODEL/dnascope.model \
    -v output-ds_tmp.vcf.gz output-ds.vcf.gz || \
    { echo "DNAModelApply failed"; exit 1; }
```

4.3 TNseq®

Sentieon® Genomics 软件的另一个典型用途是执行 Broad 研究所推荐的肿瘤-正常分析生物信息学流程，用于体细胞短变异发现（SNV + Indel）。图 4-2 说明了这样一个典型的生物信息学流程。

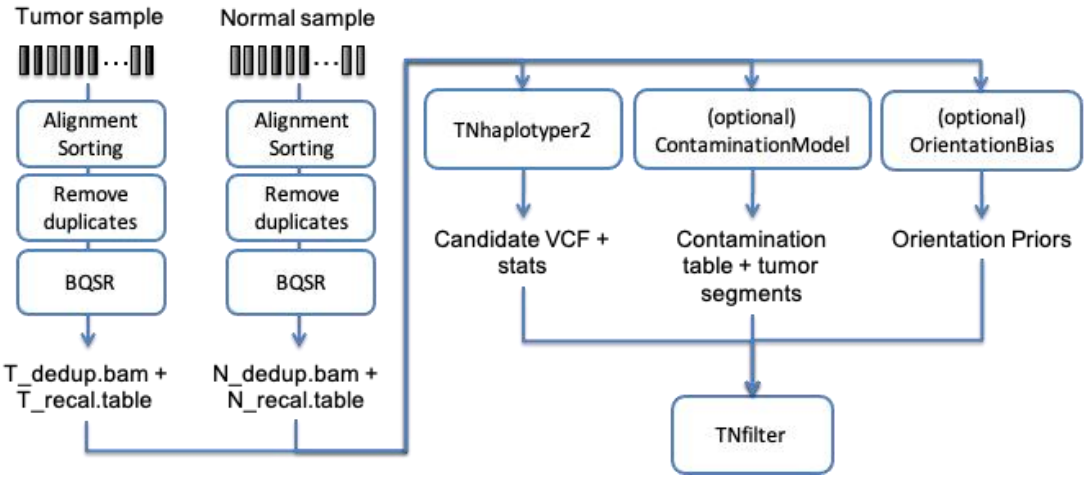

图 4-2　肿瘤-正常分析推荐的生物信息学流程

请查看 *https://support.sentieon.com/appnotes/somatic* 中的应用说明，了解不同输入的推荐体细胞变异检测流程。

4.3.1 概述

在这个生物信息学流程中，您需要以下输入：

- 包含与您将分析的样本相对应的参考基因组核苷酸序列的 FASTA 文件。
- 两组包含要分析样本的核苷酸序列的 FASTQ 文件，一组用于肿瘤样本，一组用于配对的正常样本。这些文件包含来自 DNA 测序的原始读数。该软件支持输入使用 GZIP 压缩的 FASTQ 文件。该软件仅支持包含 Sanger 格式

（Phred+33）质量分数的文件。

您还可以在流程中包含以下可选输入，这将帮助算法检测伪影并去除假阳性：

- 正常样本面板 VCF：来自多个不相关正常样本显示为变异的常见错误列表。此文件的内容将用于识别更可能是胚系变异的变异，并将其过滤掉。
- 人群资源 VCF：将用于过滤可能的胚系变异并注释结果的人群等位基因特异频率列表。

肿瘤-正常配对的典型生物信息学流程包括以下步骤：

1. 使用类似 *第 4.2.1 节* 介绍的 DNAseq 流程独立预处理肿瘤和正常样本，包括以下阶段：
   a 将读段映射到参考；您需要确保肿瘤和正常样本之间的 SM 样本标签不同，因为您需要在体细胞变异检测中将其作为参数。
   b 计算数据指标。
   c 去除重复。
   d （可选）碱基质量分数重校准（BQSR）。
2. 体细胞变异发现，包括以下阶段：
   a （可选）估计跨样本污染和肿瘤分段。
   b （可选）估计测序中可能存在的任何方向偏差。
   c 对两个单独的 BAM 文件进行体细胞候选变异检测：这一步识别癌症基因组数据相对于正常基因组显示体细胞变异的潜在位点，并计算该位点的基因型。
   d 过滤变异。

4.3.2 TNseq® 使用步骤

对于映射、重复去除和碱基质量分数重校准阶段，请参考 *第 4.1.2 节* 获取详细的使用说明。

（1）使用配对正常样本进行变异发现

运行两个主要命令对肿瘤-正常配对样本进行变异检测。为优化计算性能并减少 I/O 负担，我们建议将可选步骤（估计跨样本污染和估计方向偏差）与 TNhaplotyper 放在同一个 Sentieon driver 在同一命令中运行，以提高性能。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE \
  -i TUMOR_DEDUPED_BAM [-q TUMOR_RECAL_DATA.TABLE] \
  -i NORMAL_DEDUPED_BAM [-q NORMAL_RECAL_DATA.TABLE] \
  --algo TNhaplotyper2 --tumor_sample TUMOR_SAMPLE_NAME \
      --normal_sample NORMAL_SAMPLE_NAME \
      [--germline_vcf GERMLINE_RESOURCE] \
      [--pon PANEL_OF_NORMAL] \
      TMP_OUT_TN_VCF \
  [ --algo OrientationBias --tumor_sample TUMOR_SAMPLE_NAME \
      ORIENTATION_DATA ] \
  [ --algo ContaminationModel --tumor_sample TUMOR_SAMPLE_NAME \
      --normal_sample NORMAL_SAMPLE_NAME \
      --vcf GERMLINE_RESOURCE \
      --tumor_segments CONTAMINATION_DATA.segments \
      CONTAMINATION_DATA ]

sentieon driver -r REFERENCE \
  --algo TNfilter --tumor_sample TUMOR_SAMPLE_NAME \
  --normal_sample NORMAL_SAMPLE_NAME \
  -v TMP_OUT_TN_VCF \
  [--contamination CONTAMINATION_DATA] \
  [--tumor_segments CONTAMINATION_DATA.segments] \
  [--orientation_priors ORIENTATION_DATA] \
  OUT_TN_VCF
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。

- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参考相同。
- **TUMOR_DEDUPED_BAM**：肿瘤样本去重后的预处理 BAM 文件的位置。
- **NORMAL_DEDUPED_BAM**：正常样本去重后的预处理 BAM 文件的位置。
- **TUMOR_SAMPLE_NAME**：在映射读数到参考阶段使用的肿瘤样本名称。
- **NORMAL_SAMPLE_NAME**：在映射读数到参考阶段使用的正常样本名称。
- **TMP_OUT_TN_VCF**：TNhaplotyper2 输出文件的位置和文件名；这是一个临时文件。
- **OUT_TN_VCF**：包含变异的输出文件的位置和文件名。

该命令的以下输入是可选的：

- **TUMOR_RECAL_DATA.TABLE**：肿瘤样本 BQSR 阶段存储结果的位置。
- **NORMAL_RECAL_DATA.TABLE**：正常样本 BQSR 阶段存储结果的位置。
- **PANEL_OF_NORMAL**：正常样本面板 VCF 文件的位置和名称。
- **GERMLINE_RESOURCE**：人群种系资源的位置。资源文件中的 AF INFO 字段将用于注释变异等位基因的人群等位基因频率，然后用于识别可能不是真正体细胞变异的胚系变异。
- **ORIENTATION_DATA**: OrientationBias 生成的包含方向偏差信息的文件的位置和文件名。
- **CONTAMINATION_DATA**：ContaminationModel 生成的包含污染信息的文件的位置和文件名。
- **CONTAMINATION_DATA.segments**：ContaminationModel 生成的包含肿瘤分段信息的文件的位置和文件名。

（2）缺少配对正常样本时的特殊考虑

当缺少配对正常样本时，我们建议对上一节所示的流程进行以下更改：

- 应包括 PANEL_OF_NORMAL 和 GERMLINE_RESOURCE，因为它们在一定程度上替代了配对正常样本的作用，用于过滤测序背景噪音和人群中常见的胚系变异。同时缺少配对正常样本和这些种系资源，输出的 VCF 将包含大量胚系变异背景导致的假阳性。
- 不应包括--normal_sample 参数以及正常样本的输入 BAM 和重校准表。

（3）生成 GERMLINE_RESOURCE 种系人群资源

包含群体等位基因频率的种系人群资源文件可以通过以下程序从 gnomAD 获取并进行后处理：

- 下载您想要版本的所有每条染色体.vcf.bgz 文件。
- 对于每条染色体，删除不必要的注释以减小文件大小：

```
bcftools annotate -x ^INFO/AF,INFO/AC INPUT.chrN.vcf.bgz | bcftools norm -m +any -O z -o OUTPUT.chrN.vcf.gz
```

- 将所有文件连接到单个文件，并创建相应的索引：

```
bcftools concat -Oz -o OUTPUT.vcf.gz OUTPUT.chr*.vcf.gz && bcftools index -t OUTPUT.vcf.gz
```

例如，要从 Google 下载 GnomAD v3.1 版本，您可以运行：

```
for chr in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 X Y; do
        curl https://storage.googleapis.com/gcp-public-data--gnomad/release/3.1/vcf/genomes/gnomad.genomes.v3.1.sites.chr$chr.vcf.bgz \
            | bcftools annotate -x ^INFO/AF,INFO/AC - | bcftools norm -m +any -Oz -o tmp_OUTPUT.chr$chr.vcf.gz
        file_list="$file_list tmp_OUTPUT.chr$chr.vcf.gz"
done
bcftools concat -Oz -o OUTPUT.vcf.gz $file_list && bcftools index -t OUTPUT.vcf.gz
rm $file_list
```

4.3.3 TNseq® 配对样本全基因组分析脚本

以下是 Sentieon TNseq 对肿瘤-正常样本分析的全基因组脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform TN seq variant calling
# using a matched paired Tumor+normal sample with fastq
# files named normal_1.fastq.gz, normal_2.fastq.gz
# tumor_1.fastq.gz, tumor_2.fastq.gz
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
NORMAL_SM="normal_sample" #sample name
NORMAL_RGID="rg_$NORMAL_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data
NORMAL_FASTQ_1="$FASTQ_FOLDER/normal_1.fastq.gz"
NORMAL_FASTQ_2="$FASTQ_FOLDER/normal_2.fastq.gz"

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
CONTAMINATION_VCF="$FASTA_DIR/germline_vcf-af-only-gnomad.raw.sites.vcf"  # A VCF of
 germline sites to use for contamination detection
PON=  # the Mutect2 panel-of-normals VCF file
GERMLINE_VCF=  # A VCF of known germline sites

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443
```

```
# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/TNseq" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment1 failed"; exit 1; }

# ****************************************
# 1b. Mapping reads with BWA-MEM, sorting for normal sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
```

```
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$NORMAL_RGID\tSM:$NORMAL_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $NORMAL_FASTQ_1 $NORMAL_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o normal_sorted.bam -t $NT --sam2bam -i - || \
     { echo "Alignment2 failed"; exit 1; }

# ****************************************
# 2a. Metrics for tumor sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_sorted.bam \
     --algo MeanQualityByCycle tumor_mq_metrics.txt \
     --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
     --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' tumor_aln_metrics.txt \
     --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
     --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
     { echo "Metrics1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
     -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
     -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
     -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 2b. Metrics for normal sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_sorted.bam \
     --algo MeanQualityByCycle normal_mq_metrics.txt \
     --algo QualDistribution normal_qd_metrics.txt --algo GCBias \
     --summary normal_gc_summary.txt normal_gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' normal_aln_metrics.txt \
     --algo InsertSizeMetricAlgo normal_is_metrics.txt \
     --algo CoverageMetrics --omit_base_output normal_coverage_metrics || \
     { echo "Metrics2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o normal_gc-report.pdf normal_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
```

```
    -o normal_qd-report.pdf normal_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o normal_mq-report.pdf normal_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o normal_is-report.pdf normal_is_metrics.txt

# ****************************************
# 3a. Remove Duplicate Reads for tumor
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusCollector \
    --fun score_info tumor_score.txt || { echo "LocusCollector1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_deduped.bam || \
    { echo "Dedup1 failed"; exit 1; }

# ****************************************
# 3b. Remove Duplicate Reads for normal
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo LocusCollector \
    --fun score_info normal_score.txt || { echo "LocusCollector2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo Dedup \
    --score_info normal_score.txt --metrics normal_dedup_metrics.txt normal_deduped.bam || \
    { echo "Dedup2 failed"; exit 1; }

# ****************************************
# 4a. Somatic Variant Calling - TNhaplotyper2
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
    -i normal_deduped.bam \
    --algo TNhaplotyper2 --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
    ${PON:+--pon $PON} ${GERMLINE_VCF:+--germline_vcf $GERMLINE_VCF} output-tnhap2-tmp.vcf.gz \
```

```
    --algo OrientationBias --tumor_sample $TUMOR_SM output-orientation \
    --algo ContaminationModel --tumor_sample $TUMOR_SM --normal_sample $NORMAL
_SM \
    --vcf $CONTAMINATION_VCF \
    --tumor_segments output-contamination-segments output-contamination || \
    { echo "TNhaplotyper2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA --algo TNfilter \
    -v output-tnhap2-tmp.vcf.gz --tumor_sample $TUMOR_SM --normal_sample $NORMAL
_SM \
    --contamination output-contamination --tumor_segments output-contamination-segmen
ts \
    --orientation_priors output-orientation output-tnhap2.vcf.gz || \
    { echo "TNfilter failed"; exit 1; }


# Uncomment the following commands to run somatic variant calling with
# TNhaplotyper
# ****************************************
# 4b. Somatic Variant Calling - TNhaplotyper
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#     -i normal_deduped.bam \
#     --algo TNhaplotyper --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
#     --dbsnp $KNOWN_DBSNP output-tnhaplotyper.vcf.gz || \
#     { echo "TNhaplotyper failed"; exit 1; }


# Uncomment the following commands to run indel realignment, corealignment and
# somatic variant calling with TNsnv
# ****************************************
# 5a. Indel realigner for tumor sample
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tumor_realigned.bam || \
#     { echo "Realigner1 failed"; exit 1; }

# ****************************************
# 5b. Indel realigner for normal sample
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_deduped.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS normal_realigned.bam || \
#     { echo "Realigner2 failed"; exit 1; }
```

```
# ****************************************
# 6. Corealignment of tumor and normal
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_realigned.bam \
#     -i normal_realigned.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tn_corealigned.bam || \
#     { echo "Corealignment failed"; exit 1; }

# ****************************************
# 7. Somatic Variant Calling - TNsnv
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tn_corealigned.bam \
#     --algo TNsnv --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM --dbsnp
$KNOWN_DBSNP \
#     --call_stats_out output-call.stats output-tnsnv.vcf.gz || \
#     { echo "TNsnv failed"; exit 1; }
```

4.3.4 TNseq® 配对样本全外显子组分析脚本

以下是 Sentieon TNseq 对肿瘤-正常样本分析的全外显子组脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform TN seq variant calling
# using a matched paired Tumor+normal sample with fastq
# files named normal_1.fastq.gz, normal_2.fastq.gz
# tumor_1.fastq.gz, tumor_2.fastq.gz
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
NORMAL_SM="normal_sample" #sample name
NORMAL_RGID="rg_$NORMAL_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data
NORMAL_FASTQ_1="$FASTQ_FOLDER/normal_1.fastq.gz"
NORMAL_FASTQ_2="$FASTQ_FOLDER/normal_2.fastq.gz"

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
CONTAMINATION_VCF="$FASTA_DIR/germline_vcf-af-only-gnomad.raw.sites.vcf"  # A VCF of
 germline sites to use for contamination detection
PON=  # the Mutect2 panel-of-normals VCF file
GERMLINE_VCF=  # A VCF of known germline sites
INTERVAL_FILE="$FASTA_DIR/TruSeq_exome_targeted_regions.b37.bed"
# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443
```

```
# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/TNseq" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment1 failed"; exit 1; }

# ****************************************
# 1b. Mapping reads with BWA-MEM, sorting for normal sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
```

```
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$NORMAL_RGID\tSM:$NORMAL_SM\tPL:$PL" \
    -t $NT -K 10000000 $FASTA $NORMAL_FASTQ_1 $NORMAL_FASTQ_2 || \
    { echo -n 'BWA error'; exit 1; } ) | \
    $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o normal_sorted.bam -t $NT --sam2bam -i - || \
    { echo "Alignment2 failed"; exit 1; }

# ****************************************
# 2a. Metrics for tumor sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i tumor_sorted.bam \
    --algo MeanQualityByCycle tumor_mq_metrics.txt \
    --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
    --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' tumor_aln_metrics.txt \
    --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
    { echo "Metrics1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 2b. Metrics for normal sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval $INTERVAL_FILE} -i normal_sorted.bam \
    --algo MeanQualityByCycle normal_mq_metrics.txt \
    --algo QualDistribution normal_qd_metrics.txt --algo GCBias \
    --summary normal_gc_summary.txt normal_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' normal_aln_metrics.txt \
    --algo InsertSizeMetricAlgo normal_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output normal_coverage_metrics || \
    { echo "Metrics2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o normal_gc-report.pdf normal_gc_met
```

```
rics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o normal_qd-report.pdf normal_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o normal_mq-report.pdf normal_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o normal_is-report.pdf normal_is_metrics.txt

# ****************************************
# 3a. Remove Duplicate Reads for tumor
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusColle
ctor \
    --fun score_info tumor_score.txt || { echo "LocusCollector1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_deduped.bam
|| \
    { echo "Dedup1 failed"; exit 1; }

# ****************************************
# 3b. Remove Duplicate Reads for normal
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo LocusColl
ector \
    --fun score_info normal_score.txt || { echo "LocusCollector2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo Dedup \
    --score_info normal_score.txt --metrics normal_dedup_metrics.txt normal_deduped.ba
m || \
    { echo "Dedup2 failed"; exit 1; }

# ****************************************
# 4a. Somatic Variant Calling - TNhaplotyper2
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
$INTERVAL_FILE} -i tumor_deduped.bam \
    -i normal_deduped.bam \
```

```
    --algo TNhaplotyper2 --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
    ${PON:+--pon $PON} ${GERMLINE_VCF:+--germline_vcf $GERMLINE_VCF} output-tnhap
2-tmp.vcf.gz \
    --algo OrientationBias --tumor_sample $TUMOR_SM output-orientation \
    --algo ContaminationModel --tumor_sample $TUMOR_SM --normal_sample $NORMAL
_SM \
    --vcf $CONTAMINATION_VCF \
    --tumor_segments output-contamination-segments output-contamination || \
    { echo "TNhaplotyper2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA --algo TNfilter \
    -v output-tnhap2-tmp.vcf.gz --tumor_sample $TUMOR_SM --normal_sample $NORMAL
_SM \
    --contamination output-contamination --tumor_segments output-contamination-segmen
ts \
    --orientation_priors output-orientation output-tnhap2.vcf.gz || \
    { echo "TNfilter failed"; exit 1; }


# Uncomment the following commands to run somatic variant calling with
# TNhaplotyper
# ****************************************
# 4b. Somatic Variant Calling - TNhaplotyper
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
 $INTERVAL_FILE} -i tumor_deduped.bam \
#     -i normal_deduped.bam \
#     --algo TNhaplotyper --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
#     --dbsnp $KNOWN_DBSNP output-tnhaplotyper.vcf.gz || \
#     { echo "TNhaplotyper failed"; exit 1; }


# Uncomment the following commands to run indel realignment, corealignment and
# somatic variant calling with TNsnv
# ****************************************
# 5a. Indel realigner for tumor sample
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tumor_realigned.bam || \
#     { echo "Realigner1 failed"; exit 1; }

# ****************************************
# 5b. Indel realigner for normal sample
# ****************************************
```

```
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_deduped.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS normal_realigned.bam || \
#     { echo "Realigner2 failed"; exit 1; }

# ****************************************
# 6. Corealignment of tumor and normal
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_realigned.bam \
#     -i normal_realigned.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tn_corealigned.bam || \
#     { echo "Corealignment failed"; exit 1; }

# ****************************************
# 7. Somatic Variant Calling - TNsnv
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
 $INTERVAL_FILE} -i tn_corealigned.bam \
#     --algo TNsnv --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM --dbsnp
$KNOWN_DBSNP \
#     --call_stats_out output-call.stats output-tnsnv.vcf.gz || \
#     { echo "TNsnv failed"; exit 1; }
```

4.3.5 TNseq® 单样本全基因组分析脚本

以下是 Sentieon TNseq 在只使用肿瘤样本，没有配对正常样本时的全基因组分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform TN seq variant calling
# using a Tumor sample with fastq files
# named tumor_1.fastq.gz, tumor_2.fastq.gz
# and data from a Panel of Normals and Cosmic DB
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
# Update with the location of the panel of normal and CosmicDB vcf files
# We recommend that you create the panel of normal file with the corresponding algo
rithm that you plan to use for the somatic mutation calling.
PANEL_OF_NORMAL_TNSNV=
PANEL_OF_NORMAL_TNHAPLOTYPER=
PANEL_OF_NORMAL_TNHAPLOTYPER2=
COSMIC_DB="/home/regression/references/b37/b37_cosmic_v54_120711.vcf.gz"
CONTAMINATION_VCF="$FASTA_DIR/germline_vcf-af-only-gnomad.raw.sites.vcf"  # A VCF of
 germline sites to use for contamination detection
GERMLINE_VCF=  # A VCF of known germline sites
```

```
# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the server
START_DIR="$PWD/test/TNseq_tumoronly" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pipeline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have number of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUMOR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2bam -i - || \
     { echo "Alignment failed"; exit 1; }
```

```
# ****************************************
# 2a. Metrics for tumor sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_sorted.bam \
    --algo MeanQualityByCycle tumor_mq_metrics.txt \
    --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
    --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' tumor_aln_metrics.txt \
    --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
    { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 3a. Remove Duplicate Reads for tumor
# sample. It is possible
# to mark instead of remove duplicates
# by ommiting the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusCollector \
    --fun score_info tumor_score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --rmdup --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 4a. Somatic Variant Calling - TNhaplotyper2
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
    --algo TNhaplotyper2 --tumor_sample $TUMOR_SM \
    ${PANEL_OF_NORMAL_TNHAPLOTYPER2:+--pon $PANEL_OF_NORMAL_TNHAPLOTYPER2} \
    ${GERMLINE_VCF:+--germline_vcf $GERMLINE_VCF} output-tnhap2-tmp.vcf.gz \
```

```
    --algo OrientationBias --tumor_sample $TUMOR_SM output-orientation \
    --algo ContaminationModel --tumor_sample $TUMOR_SM  --vcf $CONTAMINATION_VC
F \
    --tumor_segments output-contamination-segments output-contamination || \
    { echo "TNhaplotyper2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA --algo TNfilter \
    -v output-tnhap2-tmp.vcf.gz --tumor_sample $TUMOR_SM \
    --contamination output-contamination --tumor_segments output-contamination-segmen
ts \
    --orientation_priors output-orientation output-tnhap2.vcf.gz || \
    { echo "TNfilter failed"; exit 1; }


# Uncomment the following commands to run somatic variant calling with TNhaplotyper
# ****************************************
# 4b. Somatic Variant Calling - TNhaplotyper
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#     --algo TNhaplotyper --tumor_sample $TUMOR_SM \
#     ${PANEL_OF_NORMAL_TNHAPLOTYPER:+--pon $PANEL_OF_NORMAL_TNHAPLOTYPER}
\
#     --cosmic $COSMIC_DB --dbsnp $KNOWN_DBSNP \
#     output-tnhaplotyper.vcf.gz || { echo "TNhaplotyper failed"; exit 1; }


# Uncomment the following commands to run indel realignment, and somatic variant
# calling with TNsnv
# ****************************************
# 5. Indel realigner for tumor sample
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#     --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tumor_realigned.bam || \
#     { echo "Realigner failed"; exit 1; }

# ****************************************
# 6. Somatic Variant Calling - TNsnv
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_realigned.bam \
#     --algo TNsnv --tumor_sample $TUMOR_SM \
#     ${PANEL_OF_NORMAL_TNSNV:+--pon $PANEL_OF_NORMAL_TNSNV} \
#     --cosmic $COSMIC_DB --dbsnp $KNOWN_DBSNP \
#     --call_stats_out output-call.stats output-tnsnv.vcf.gz || \
#     { echo "TNsnv failed"; exit 1; }
```

4.3.6 TNseq® 单样本全外显子组分析脚本

以下是 Sentieon TNseq 在只使用肿瘤样本，没有配对正常样本时的全外显子组分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform TN seq variant calling
# using a Tumor sample with fastq files
# named tumor_1.fastq.gz, tumor_2.fastq.gz
# and data from a Panel of Normals and Cosmic DB
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
# Update with the location of the panel of normal and CosmicDB vcf files
# We recommend that you create the panel of normal file with the corresponding algo
rithm that you plan to use for the somatic mutation calling.
PANEL_OF_NORMAL_TNSNV=
PANEL_OF_NORMAL_TNHAPLOTYPER=
PANEL_OF_NORMAL_TNHAPLOTYPER2=
COSMIC_DB="/home/regression/references/b37/b37_cosmic_v54_120711.vcf.gz"
CONTAMINATION_VCF="$FASTA_DIR/germline_vcf-af-only-gnomad.raw.sites.vcf"  # A VCF of
 germline sites to use for contamination detection
GERMLINE_VCF=  # A VCF of known germline sites
INTERVAL_FILE="$FASTA_DIR/TruSeq_exome_targeted_regions.b37.bed"
```

```
# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the server
START_DIR="$PWD/test/TNseq_tumoronly" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pipeline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have number of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUMOR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2bam -i - || \
     { echo "Alignment failed"; exit 1; }
```

```
# ****************************************
# 2a. Metrics for tumor sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
$INTERVAL_FILE} -i tumor_sorted.bam \
    --algo MeanQualityByCycle tumor_mq_metrics.txt \
    --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
    --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' tumor_aln_metrics.txt \
    --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
    { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metric
s.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 3a. Remove Duplicate Reads for tumor
# sample. It is possible
# to mark instead of remove duplicates
# by ommiting the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusColle
ctor \
    --fun score_info tumor_score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --rmdup --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_dedup
ed.bam || \
    { echo "Dedup failed"; exit 1; }

# ****************************************
# 4a. Somatic Variant Calling - TNhaplotyper2
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
$INTERVAL_FILE} -i tumor_deduped.bam \
    --algo TNhaplotyper2 --tumor_sample $TUMOR_SM \
    ${PANEL_OF_NORMAL_TNHAPLOTYPER2:+--pon $PANEL_OF_NORMAL_TNHAPLOTYPER2}
```

```
 \
     ${GERMLINE_VCF:+--germline_vcf $GERMLINE_VCF} output-tnhap2-tmp.vcf.gz \
     --algo OrientationBias --tumor_sample $TUMOR_SM output-orientation \
     --algo ContaminationModel --tumor_sample $TUMOR_SM  --vcf $CONTAMINATION_VC
F \
     --tumor_segments output-contamination-segments output-contamination || \
     { echo "TNhaplotyper2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA --algo TNfilter \
     -v output-tnhap2-tmp.vcf.gz --tumor_sample $TUMOR_SM \
     --contamination output-contamination --tumor_segments output-contamination-segmen
ts \
     --orientation_priors output-orientation output-tnhap2.vcf.gz || \
     { echo "TNfilter failed"; exit 1; }


# Uncomment the following commands to run somatic variant calling with TNhaplotyper
# ****************************************
# 4b. Somatic Variant Calling - TNhaplotyper
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
 $INTERVAL_FILE} -i tumor_deduped.bam \
#      --algo TNhaplotyper --tumor_sample $TUMOR_SM \
#      ${PANEL_OF_NORMAL_TNHAPLOTYPER:+--pon $PANEL_OF_NORMAL_TNHAPLOTYPER}
\
#      --cosmic $COSMIC_DB --dbsnp $KNOWN_DBSNP \
#      output-tnhaplotyper.vcf.gz || { echo "TNhaplotyper failed"; exit 1; }


# Uncomment the following commands to run indel realignment, and somatic variant
# calling with TNsnv
# ****************************************
# 5. Indel realigner for tumor sample
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
#      --algo Realigner -k $KNOWN_MILLS -k $KNOWN_INDELS tumor_realigned.bam || \
#      { echo "Realigner failed"; exit 1; }

# ****************************************
# 6. Somatic Variant Calling - TNsnv
# ****************************************
#$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT ${INTERVAL_FILE:+--interval
 $INTERVAL_FILE} -i tumor_realigned.bam \
#      --algo TNsnv --tumor_sample $TUMOR_SM \
```

```
#     ${PANEL_OF_NORMAL_TNSNV:+--pon $PANEL_OF_NORMAL_TNSNV} \
#     --cosmic $COSMIC_DB --dbsnp $KNOWN_DBSNP \
#     --call_stats_out output-call.stats output-tnsnv.vcf.gz || \
#     { echo "TNsnv failed"; exit 1; }
```

4.4 TNscope®

Sentieon® Genomics 软件的用途是执行生物信息学流程，用于仅使用肿瘤或肿瘤-正常分析的体细胞变异和结构变异检测算法。图 4-3 说明了这样一个典型的生物信息学流程。

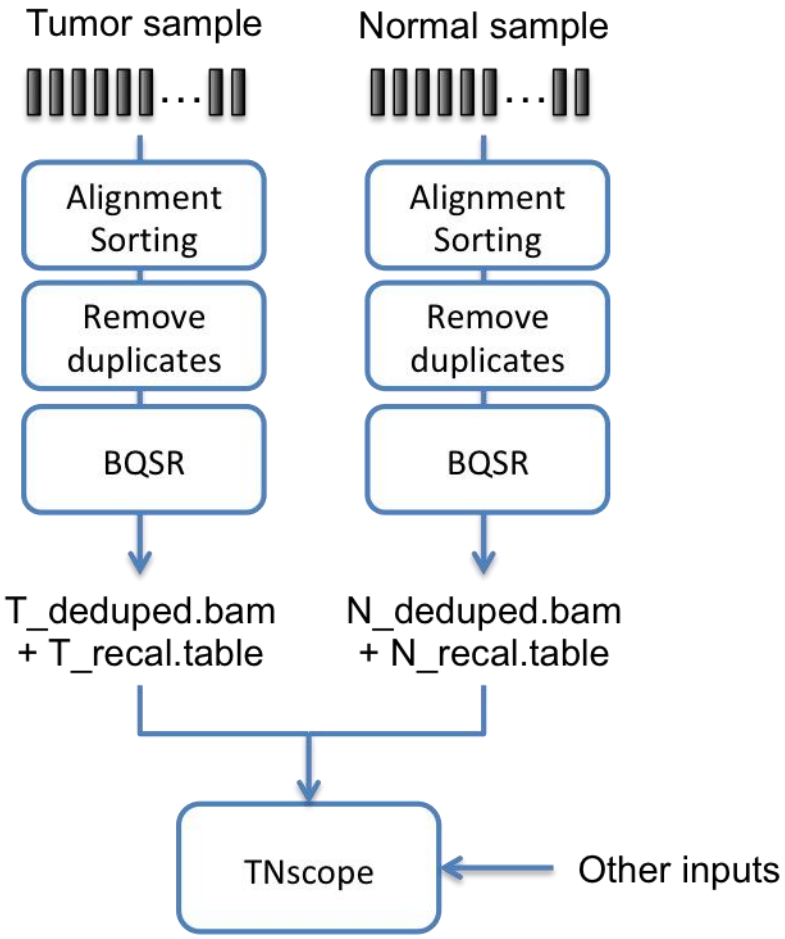

图 4-3 基于 TNscope® 的肿瘤-正常分析的推荐流程

请查看 *https://support.sentieon.com/appnotes/somatic* 中的应用说明以获取推荐的具有不同输入的体细胞变异检测流程。

4.4.1 概述

在此生物信息学流程中，您将需要以下输入：

- 包含与您将分析的样本相对应的参考基因组核苷酸序列的 FASTA 文件。
- 两组 FASTQ 文件，包含要分析的样本，一个用于肿瘤样本，一个用于匹配的正常样本。这些文件包含来自 DNA 测序的原始读数。该软件支持输入使用 GZIP 压缩的 FASTQ 文件。该软件仅支持包含 Sanger 格式的质量分数（Phred+33）的文件。
- （可选）单核苷酸多态性数据库（dbSNP）数据，以 VCF 文件的形式包含在流程中。
- （可选）您想要添加到流程中的其他数据。数据以 VCF 文件的形式使用。

肿瘤-正常配对样本的流程包括以下步骤：

1. 使用 DNA 独立预处理肿瘤和正常样品 seq 管道（如 *第 4.1.2 节* 中介绍的管道）具有以下阶段：
   a 将读段映射到参考；您需要确保肿瘤样本和正常样本之间的 SM 样本标签不同，因为您将需要它作为体细胞变异检测中的参数，变异检测算法需要根据 SM 标签区分样本类别。您还需要确保两个样本的 RG：ID 不同且唯一。
   b 计算数据指标。
   c 去除重复。
   d （可选）碱基质量分数重校准（BQSR）。
2. 体细胞变异检出：此步骤确定癌症基因组数据显示相对于正常基因组的体细胞变异的位点，并计算该位点的基因型。

4.4.2 TNscope® 使用步骤

对于映射、重复去除和碱基质量分数重校准阶段，请参考 *第 4.1.2 节* 获取详细的使用说明。

（1）有或没有匹配正常样本的体细胞变异发现

运行单个命令以对肿瘤-正常配对进行变异检测：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE \
  -i TUMOR_DEDUPED_BAM [-q TUMOR_RECAL_DATA.TABLE] \
  -i NORMAL_DEDUPED_BAM [-q NORMAL_RECAL_DATA.TABLE] \
  --algo TNscope \
  --tumor_sample TUMOR_SAMPLE_NAME --normal_sample NORMAL_SAMPLE_NAME \
  [--dbsnp DBSNP] OUT_TN_VCF
```

如果您没有匹配的正常样本，则可以跳过正常样本 BAM 文件和样本名称输入，因为这些不是必需的；然而，由于缺乏正常样本，胚系变异将存在于输出文件中：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE \
  -i TUMOR_DEDUPED_BAM [-q TUMOR_RECAL_DATA.TABLE] \
  --algo TNscope --tumor_sample TUMOR_SAMPLE_NAME \
  [--dbsnp DBSNP] OUT_TN_VCF
```

为了在缺少匹配的正常样本时，您可以将匹配的正常样本替换为使用下文 *第 4.4.2 节（2）生成正常 VCF 面板文件* 中描述的方法生成的通用正常面板。

运行单个命令以仅对肿瘤样本进行变异检测：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE \
  -i TUMOR_DEDUPED_BAM [-q TUMOR_RECAL_DATA.TABLE] \
  --algo TNscope --tumor_sample TUMOR_SAMPLE_NAME \
  --pon PANEL_OF_NORMAL_VCF [--dbsnp DBSNP] OUT_TN_VCF
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参

考相同。

- **TUMOR_DEDUPED_BAM**：肿瘤样本去重后的预处理 BAM 文件的位置。
- **NORMAL_DEDUPED_BAM**：正常样本的预处理 BAM 文件的位置。
- **TUMOR_SAMPLE_NAME**：在映射读数到参考阶段使用的肿瘤样本名称。
- **OUT_TN_VCF**：包含变异的输出文件的位置和文件名。

该命令的以下输入是可选的：

- **TUMOR_RECAL_DATA.TABLE**：肿瘤样本 BQSR 阶段存储结果的位置。
- **NORMAL_RECAL_DATA.TABLE**：正常样本 BQSR 阶段存储结果的位置。
- **NORMAL_SAMPLE_NAME**：在映射读数到参考阶段使用的正常样本名称。
- **DBSNP**：单核苷酸多态性数据库（dbSNP）的位置。dbSNP 中的变异更有可能被标记为种系，因为它们需要更多证据证明是体细胞变异。您只能使用一个 dbSNP 文件。
- **PANEL_OF_NORMAL_VCF**：正常样本面板 VCF 文件的位置和名称。

（2）生成正常 VCF 面板文件

为了生成您自己的正常 VCF 面板文件以用于 TNscope®，您需要对面板中的每个正常样本运行以下命令：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i NORMAL_DEDUPED_BAM \
    [-q NORMAL_RECAL_DATA.TABLE] --algo TNscope \
    --tumor_sample NORMAL_SAMPLE_NAME OUT_NORMAL_VCF
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参

考相同。

- **NORMAL_DEDUPED_BAM**：正常样本的预处理 BAM 文件的位置。
- **NORMAL_SAMPLE_NAME**：在映射读数到参考阶段使用的正常样本名称。
- **OUT_NORMAL_VCF**：包含输入正常样本相关变异的输出 VCF 文件的位置和名称。

该命令的以下输入是可选的：

- **NORMAL_RECAL_DATA.TABLE**：正常样本 BQSR 阶段存储结果的位置。

生成要包含在面板中的所有 VCF 文件后，您需要将它们合并到一个正常 VCF 面板中。您可以使用 bcftools 实现该目的：

```
BCF=/path_to_bcftools
export BCFTOOLS_PLUGINS=$BCF/plugins
DIR=/path_to_normal_vcf_file
$BCF/bcftools merge -m all -f PASS,. --force-samples $DIR/*.vcf.gz |\
$BCF/bcftools plugin fill-AN-AC |\
$BCF/bcftools filter -i 'SUM(AC)>1' > panel_of_normal.vcf
```

4.4.3 TNscope® PCR 扩增子分析脚本

以下是 TNscope 使用配对正常样本对扩增子数据进行体细胞变异检测的脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform TNscope variant calling on
# PCR amplicon tumor sample with coverage over
# 200x and a matched normal sample.
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
NORMAL_SM="normal_sample" #sample name
NORMAL_RGID="rg_$NORMAL_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data
NORMAL_FASTQ_1="$FASTQ_FOLDER/normal_1.fastq.gz"
NORMAL_FASTQ_2="$FASTQ_FOLDER/normal_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
INTERVAL_FILE="targeted_regions.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
TNSCOPE_FILTER=/home/sentieon-scripts/tnscope_filter/tnscope_filter.py
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
```

```
 server
START_DIR="$PWD/test/TNscope" #Determine where the output files will be stored

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for the tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment1 failed"; exit 1; }

# ****************************************
# 1b. Mapping reads with BWA-MEM, sorting for the normal sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$NORMAL_RGID\tSM:$NOR
MAL_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $NORMAL_FASTQ_1 $NORMAL_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o normal_sorted.bam -t $NT --sam2b
am -i - || \
     { echo "Alignment2 failed"; exit 1; }

# ****************************************
# 2a. Metrics for the tumor sample
```

```
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_sorted.bam \
    --algo MeanQualityByCycle tumor_mq_metrics.txt \
    --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
    --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' tumor_aln_metrics.txt \
    --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
    { echo "Metrics1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 2b. Metrics for the normal sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_sorted.bam \
    --algo MeanQualityByCycle normal_mq_metrics.txt \
    --algo QualDistribution normal_qd_metrics.txt --algo GCBias \
    --summary normal_gc_summary.txt normal_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' normal_aln_metrics.txt \
    --algo InsertSizeMetricAlgo normal_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output normal_coverage_metrics || \
    { echo "Metrics2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o normal_gc-report.pdf normal_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o normal_qd-report.pdf normal_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o normal_mq-report.pdf normal_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o normal_is-report.pdf normal_is_metrics.txt

# ****************************************
# 4. Somatic and Structural variant calling
# ****************************************
# Consider adding `--disable_detector sv` if not interested in SV calling
```

```
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    ${INTERVAL_FILE:+--interval_padding 10 --interval $INTERVAL_FILE} \
    -i tumor_sorted.bam -i normal_sorted.bam \
    --algo TNscope \
    --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
    --dbsnp $KNOWN_DBSNP \
    --min_tumor_allele_frac 0.009 \
    --max_fisher_pv_active 0.05 \
    --max_normal_alt_cnt 10 \
    --max_normal_alt_frac 0.01 \
    --max_normal_alt_qsum 250 \
    --sv_mask_ext 10 \
    --prune_factor 0 \
    --assemble_mode 2 \
    output_tnscope.pre_filter.vcf.gz || \
    { echo "TNscope failed"; exit 1; }

# ****************************************
# 6. Variant filtration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon pyexec $TNSCOPE_FILTER \
    -v output_tnscope.pre_filter.vcf.gz \
    --tumor_sample $TUMOR_SM \
    --normal_sample $NORMAL_SM \
    -x amplicon --min_tumor_af 0.0095 --min_depth 200 \
    output_tnscope.filter.vcf.gz || \
    { echo "TNscope filter failed"; exit 1; }
```

4.4.4 TNscope ® UMI 标签分析脚本

以下是 TNscope 在专门处理带有 UMI 标签的样本数据的脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform TNscope variant calling on
# UMI-tagged samples
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
INTERVAL_FILE="targeted_regions.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
TNSCOPE_FILTER=/home/sentieon-scripts/tnscope_filter/tnscope_filter.py
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

#UMI information
READ_STRUCTURE="12M11S+T,+T" #an example duplex: "3M2S+T,3M2S+T" where duplex
UMI extraction requires an identical read structure for both strands
DUPLEX_UMI="false" #set to "true" if duplex

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/TNscope_umi" #Determine where the output files will be stored
```

```
# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Pre-processing of FASTQ containing UMIs
# ****************************************
if [ "$DUPLEX_UMI" = "true" ] ; then
     READ_STRUCTURE="-d $READ_STRUCTURE"
fi
( $SENTIEON_INSTALL_DIR/bin/sentieon umi extract $READ_STRUCTURE $TUMOR_FASTQ_1
 $TUMOR_FASTQ_2 || \
     { echo -n 'Extract error' >&2; exit -1; } ) | \
  ( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -p -C \
  -R "@RG\tID:$TUMOR_RGID\tSM:$TUMOR_SM\tPL:$PL" -t $NT \
  -K 10000000 $FASTA - || { echo -n 'BWA error'; exit 1; } ) | \
  $SENTIEON_INSTALL_DIR/bin/sentieon util sort -t $NT -r $FASTA --sam2bam -i - \
  -o tumor_sorted.bam || \
  { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Remove duplicate reads for the tumor sample
# use the consensus-based approach with UMI.
# It is possible to remove instead of mark
# duplicates by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusColle
ctor \
     --consensus --umi_tag XR --fun score_info tumor_score.txt || { echo "LocusCollector
 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam -r $FASTA --algo
 Dedup \
     --score_info tumor_score.txt --metrics umi.dedup_metrics.txt tumor_deduped.bam ||
\
     { echo "Dedup failed"; exit 1; }

# ****************************************
```

```
# 3. Somatic and Structural variant calling
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} \
    --algo TNscope \
    --tumor_sample $TUMOR_SM \
    --dbsnp $KNOWN_DBSNP \
    --disable_detector sv \
    --min_tumor_allele_frac 3e-3 \
    --min_tumor_lod 3.0 \
    --min_init_tumor_lod 1.0 \
    --assemble_mode 4 \
    --pcr_indel_model NONE \
    --resample_depth 100000 \
    output_tnscope.pre_filter.vcf.gz || \
    { echo "TNscope failed"; exit 1; }

# ****************************************
# 4. Variant filtration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon pyexec $TNSCOPE_FILTER \
    -v output_tnscope.pre_filter.vcf.gz \
    --tumor_sample $TUMOR_SM \
    -x ctdna --min_tumor_af 0.001 --min_depth 1000 \
    output_tnscope.filter.vcf.gz || \
    { echo "TNscope filter failed"; exit 1; }
```

4.4.5 TNscope® cfDNA 分析脚本

以下是 TNscope 在无 UMI 游离 DNA 样本的脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform TNscope variant calling on
# hybridization captured cfDNA tumor sample
# without UMI.
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
INTERVAL_FILE="targeted_regions.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
TNSCOPE_FILTER=/home/sentieon-scripts/tnscope_filter/tnscope_filter.py
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.co
m:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/TNscope" #Determine where the output files will be stored

# ***************************************************************************
***************************************************************************
**************
```

```
# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1. Mapping reads with BWA-MEM, sorting for the tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment failed"; exit 1; }

# ****************************************
# 2. Remove duplicate reads for the tumor sample
# use the consensus-based approach.
# It is possible to remove instead of mark
# duplicates by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusColle
ctor \
     --consensus --fun score_info tumor_score.txt || { echo "LocusCollector failed"; exit 1;
 }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam -r $FASTA --algo
 Dedup \
     --score_info tumor_score.txt tumor_deduped.bam || \
     { echo "Dedup failed"; exit 1; }

# ****************************************
# 3. Somatic and Structural variant calling
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
     ${INTERVAL_FILE:+--interval $INTERVAL_FILE} \
```

```
    --algo TNscope \
    --tumor_sample $TUMOR_SM \
    --dbsnp $KNOWN_DBSNP \
    --disable_detector sv \
    --min_tumor_allele_frac 3e-3 \
    --min_tumor_lod 3.0 \
    --min_init_tumor_lod 1.0 \
    --assemble_mode 4 \
    --pcr_indel_model NONE \
    --min_base_qual 40 \
    --resample_depth 100000 \
    output_tnscope.pre_filter.vcf.gz || \
    { echo "TNscope failed"; exit 1; }

# ****************************************
# 4. Variant filtration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon pyexec $TNSCOPE_FILTER \
    -v output_tnscope.pre_filter.vcf.gz \
    --tumor_sample $TUMOR_SM \
    -x ctdna --min_tumor_af 0.005 --min_depth 400 \
    output_tnscope.filter.vcf.gz || \
    { echo "TNscope filter failed"; exit 1; }
```

4.4.6 TNscope® 杂交捕获分析脚本

以下是 TNscope 使用配对正常样本对杂交捕获数据进行体细胞变异检测的脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform TNscope variant calling on
# hybridization captured tumor sample with
# coverage over 200x and a matched normal sample.
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
NORMAL_SM="normal_sample" #sample name
NORMAL_RGID="rg_$NORMAL_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data
NORMAL_FASTQ_1="$FASTQ_FOLDER/normal_1.fastq.gz"
NORMAL_FASTQ_2="$FASTQ_FOLDER/normal_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
INTERVAL_FILE="targeted_regions.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
TNSCOPE_FILTER=/home/sentieon-scripts/tnscope_filter/tnscope_filter.py
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
```

```
 server
START_DIR="$PWD/test/TNscope" #Determine where the output files will be stored

# ********************************************************************************
**********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for the tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment1 failed"; exit 1; }

# ****************************************
# 1b. Mapping reads with BWA-MEM, sorting for the normal sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$NORMAL_RGID\tSM:$NOR
MAL_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $NORMAL_FASTQ_1 $NORMAL_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o normal_sorted.bam -t $NT --sam2b
am -i - || \
     { echo "Alignment2 failed"; exit 1; }

# ****************************************
# 2a. Metrics for the tumor sample
```

```
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_sorted.bam \
    --algo MeanQualityByCycle tumor_mq_metrics.txt \
    --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
    --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' tumor_aln_metrics.txt \
    --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output tumor_coverage_metrics || \
    { echo "Metrics1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 2b. Metrics for the normal sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_sorted.bam \
    --algo MeanQualityByCycle normal_mq_metrics.txt \
    --algo QualDistribution normal_qd_metrics.txt --algo GCBias \
    --summary normal_gc_summary.txt normal_gc_metrics.txt --algo AlignmentStat \
    --adapter_seq '' normal_aln_metrics.txt \
    --algo InsertSizeMetricAlgo normal_is_metrics.txt \
    --algo CoverageMetrics --omit_base_output normal_coverage_metrics || \
    { echo "Metrics2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o normal_gc-report.pdf normal_gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o normal_qd-report.pdf normal_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o normal_mq-report.pdf normal_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o normal_is-report.pdf normal_is_metrics.txt

# ****************************************
# 3a. Remove duplicate reads for the tumor sample.
# It is possible to remove instead of mark
# duplicates by adding the --rmdup option in Dedup
```

```
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusCollector \
    --fun score_info tumor_score.txt || { echo "LocusCollector1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_deduped.bam || \
    { echo "Dedup1 failed"; exit 1; }

# ****************************************
# 3b. Remove duplicate reads for the normal sample.
# It is possible to remove instead of mark
# duplicates by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo LocusCollector \
    --fun score_info normal_score.txt || { echo "LocusCollector2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo Dedup \
    --score_info normal_score.txt --metrics normal_dedup_metrics.txt normal_deduped.bam || \
    { echo "Dedup2 failed"; exit 1; }

# ****************************************
# 4. Somatic and Structural variant calling
# ****************************************
# Consider adding `--disable_detector sv` if not interested in SV calling
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT \
    ${INTERVAL_FILE:+--interval $INTERVAL_FILE} \
    -i tumor_deduped.bam -i normal_deduped.bam \
    --algo TNscope \
    --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
    --dbsnp $KNOWN_DBSNP \
    --min_tumor_allele_frac 0.009 \
    --max_fisher_pv_active 0.05 \
    --max_normal_alt_cnt 10 \
    --max_normal_alt_frac 0.01 \
    --max_normal_alt_qsum 250 \
    --sv_mask_ext 10 \
    --prune_factor 0 \
    output_tnscope.pre_filter.vcf.gz || \
    { echo "TNscope failed"; exit 1; }
```

```
# ****************************************
# 5. Variant filtration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon pyexec $TNSCOPE_FILTER \
    -v output_tnscope.pre_filter.vcf.gz \
    --tumor_sample $TUMOR_SM \
    --normal_sample $NORMAL_SM \
    -x tissue_panel --min_tumor_af 0.0095 --min_depth 200 \
    output_tnscope.filter.vcf.gz || \
    { echo "TNscope filter failed"; exit 1; }
```

4.4.7 TNscope® 全基因组分析脚本

以下是 TNscope 在全基因组测序中适用于测序深度达到 50x 的样本的脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2024 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform TNscope variant calling on
# samples up to 50x coverage using a matched
# paired Tumor+normal sample with fastq files
# named normal_1.fastq.gz, normal_2.fastq.gz
# tumor_1.fastq.gz, tumor_2.fastq.gz
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
TUMOR_SM="tumor_sample" #sample name
TUMOR_RGID="rg_$TUMOR_SM" #read group ID
NORMAL_SM="normal_sample" #sample name
NORMAL_RGID="rg_$NORMAL_SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
TUMOR_FASTQ_1="$FASTQ_FOLDER/tumor_1.fastq.gz"
TUMOR_FASTQ_2="$FASTQ_FOLDER/tumor_2.fastq.gz" #If using Illumina paired data
NORMAL_FASTQ_1="$FASTQ_FOLDER/normal_1.fastq.gz"
NORMAL_FASTQ_2="$FASTQ_FOLDER/normal_2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
PON=  # the TNscope panel-of-normals VCF

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
```

```
NT=$(nproc) #number of threads to use in computation, set to number of cores in the
 server
START_DIR="$PWD/test/TNscope" #Determine where the output files will be stored
BCFTOOLS_BINARY="/home/release/other_tools/bcftools-1.7/bcftools" #bcftools >=1.7 is rec
ommended
MIN_QUAL=10




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR

# ****************************************
# 1a. Mapping reads with BWA-MEM, sorting for tumor sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$TUMOR_RGID\tSM:$TUM
OR_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $TUMOR_FASTQ_1 $TUMOR_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o tumor_sorted.bam -t $NT --sam2ba
m -i - || \
     { echo "Alignment1 failed"; exit 1; }

# ****************************************
# 1b. Mapping reads with BWA-MEM, sorting for normal sample
# ****************************************
#The results of this call are dependent on the number of threads used. To have numb
```

```
er of threads independent results, add chunk size option -K 10000000
( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$NORMAL_RGID\tSM:$NOR
MAL_SM\tPL:$PL" \
     -t $NT -K 10000000 $FASTA $NORMAL_FASTQ_1 $NORMAL_FASTQ_2 || \
     { echo -n 'BWA error'; exit 1; } ) | \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -o normal_sorted.bam -t $NT --sam2b
am -i - || \
     { echo "Alignment2 failed"; exit 1; }

# ****************************************
# 2a. Metrics for tumor sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_sorted.bam \
     --algo MeanQualityByCycle tumor_mq_metrics.txt \
     --algo QualDistribution tumor_qd_metrics.txt --algo GCBias \
     --summary tumor_gc_summary.txt tumor_gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' tumor_aln_metrics.txt \
     --algo InsertSizeMetricAlgo tumor_is_metrics.txt \
     --algo CoverageMetrics tumor_coverage_metrics || \
     { echo "Metrics1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o tumor_gc-report.pdf tumor_gc_metric
s.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
     -o tumor_qd-report.pdf tumor_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
     -o tumor_mq-report.pdf tumor_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
     -o tumor_is-report.pdf tumor_is_metrics.txt

# ****************************************
# 2b. Metrics for normal sample
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i normal_sorted.bam \
     --algo MeanQualityByCycle normal_mq_metrics.txt \
     --algo QualDistribution normal_qd_metrics.txt --algo GCBias \
     --summary normal_gc_summary.txt normal_gc_metrics.txt --algo AlignmentStat \
     --adapter_seq '' normal_aln_metrics.txt \
     --algo InsertSizeMetricAlgo normal_is_metrics.txt \
     --algo CoverageMetrics normal_coverage_metrics || \
     { echo "Metrics2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o normal_gc-report.pdf normal_gc_met
rics.txt
```

```
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution \
    -o normal_qd-report.pdf normal_qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle \
    -o normal_mq-report.pdf normal_mq_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo \
    -o normal_is-report.pdf normal_is_metrics.txt

# ****************************************
# 3a. Remove Duplicate Reads for tumor
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo LocusColle
ctor \
    --fun score_info tumor_score.txt || { echo "LocusCollector1 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i tumor_sorted.bam --algo Dedup \
    --score_info tumor_score.txt --metrics tumor_dedup_metrics.txt tumor_deduped.bam
|| \
    { echo "Dedup1 failed"; exit 1; }

# ****************************************
# 3b. Remove Duplicate Reads for normal
# sample. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo LocusColl
ector \
    --fun score_info normal_score.txt || { echo "LocusCollector2 failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i normal_sorted.bam --algo Dedup \
    --score_info normal_score.txt --metrics normal_dedup_metrics.txt normal_deduped.ba
m || \
    { echo "Dedup2 failed"; exit 1; }

# ****************************************
# 4. Somatic and Structural variant calling
# ****************************************
# Consider adding `--disable_detector sv --trim_soft_clip` if not interested in SV calling
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i tumor_deduped.bam \
    -i normal_deduped.bam \
    --algo TNscope --tumor_sample $TUMOR_SM --normal_sample $NORMAL_SM \
```

```
    --dbsnp $KNOWN_DBSNP ${PON:+--pon $PON} output_tnscope.pre_filter.vcf.gz || \
    { echo "TNscope failed"; exit 1; }

# ****************************************
# 5. Variant filtration
# ****************************************
( $BCFTOOLS_BINARY annotate -x "FILTER/triallelic_site" output_tnscope.pre_filter.vcf.gz ||
 \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "insignificant" -e "(PV>0.25 && PV2>0.25) || (IN
FO/STR == 1 && PV>0.05)" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "orientation_bias" -e "FMT/FOXOG[0] == 1" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "strand_bias" -e "SOR > 3" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "low_qual" -e "QUAL < $MIN_QUAL" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "short_tandem_repeat" -e "RPA[0]>=10" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "noisy_region" -e "ECNT>5" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
    ( $BCFTOOLS_BINARY filter -m + -s "read_pos_bias" -e "FMT/ReadPosRankSumPS[0]
< -8" || \
        { echo "VCF filtering failed"; exit 1; } ) | \
  $SENTIEON_INSTALL_DIR/bin/sentieon util vcfconvert - output_tnscope.filtered.vcf.gz || \
        { echo "VCF filtering failed"; exit 1; }
```

4.4.8 附录

4.4.8.1 TNscope 命令行可选参数说明

- **--assemble_mode**：设置 assemble_mode=2 可以在准确性和速度之间取得良好平衡。而在处理更复杂的基因组区域时，可以使用 assemble_mode=4。
- **--sv_mask_ext**：用于防止在结构变异断点附近进行 SNP/Indel 检测，以减少因比对紊乱导致的假阳性。
- **--max_fisher_pv_active**：开启并设置 P 值（PV）过滤阈值，该指标用于衡量特定位点在肿瘤样本与正常样本之间的统计学差异。
- **--min_tumor_allele_frac**：设置被视为潜在变异位点的最低肿瘤等位基因频率（AF）。
- **--min_init_tumor_lod**：初始变异检测阶段的最低肿瘤对数优势比。
- **--min_tumor_lod**：输出 VCF 文件中带有 TLOD 注释的最低肿瘤对数优势比过滤阈值。
- **--prune_factor**：设置 prune_factor=0 以开启自适应图剪枝算法。

4.4.8.2 常见问题解答（FAQ）

（1）AD 比率与输出 VCF 中报告的 AF 不匹配：

- 原因是两者的计算方式和定义不同，这是源于 GATK 的历史遗留问题。
- AD：基于输入 BAM 文件的原始堆叠计算，是在局部组装之前生成的。
- AF：是在局部组装完成后，支持参考等位基因（REF）与突变等位基因（ALT）读段数量的比率。

- 过滤机制不同：读段在参与这两项计算之前，会经过略有不同的过滤标准。

（2）使用正常样本对照库（Panel of Normal, PoN）：

- PoN 可以配合配对正常样本使用，也可以在无对照样本的情况下单独使用。
- 若变异位点出现在 PoN 中，会被标记上 panel_of_normal 过滤器标签。
- 在没有配对正常样本时，PoN 非常有用。您可以参考“*4.4.2 TNscope® 使用步骤*“章节中的”*（2）生成正常 VCF 面板文件*”获取使用 TNscope 生成 PoN VCF 的说明。

（3）COSMIC 和 dbSNP 数据库文件的作用：

- 它们仅对“肿瘤-正常”配对样本模式产生影响，且仅影响过滤步骤。
- 这两个数据库仅用于确定判断胚系风险时的正常样本对数优势比阈值。

（4）复杂变异：

Sentieon 提供了一个实验性的 Python 脚本，用于合并相同相位内的相邻变异。该脚本处于实验阶段，可在 *https://github.com/Sentieon/sentieon-scripts/blob/master/merge_mnp/merge_mnp.py* 中获取。

4.5 RNAseq

Sentieon® Genomics 软件的一个典型用途是执行 Broad 研究所最佳实践中推荐的 RNA 分析生物信息学流程，该流程描述在

https://gatk.broadinstitute.org/hc/en-us/articles/360035531192-RNAseq-short-variant-discovery-SNPs-Indels-

4.5.1 概述

在这个生物信息学流程中，您需要以下输入：

- 包含与您将分析的样本相对应的参考基因组核苷酸序列的 FASTA 文件。
- 一个或多个包含待分析样本核苷酸序列的 FASTQ 文件。这些文件包含来自 DNA 测序的原始读数。该软件支持输入使用 GZIP 压缩的 FASTQ 文件。该软件仅支持包含 Sanger 格式（Phred+33）质量分数的文件。
- （可选）您想在流程中包含的单核苷酸多态性数据库（dbSNP）数据。数据以 VCF 文件的形式使用；您可以使用 bgzip 压缩并索引的 VCF 文件。
- （可选）您想在流程中包含的多个已知位点集合。数据以 VCF 文件的形式使用；您可以使用 bgzip 压缩并索引的 VCF 文件。

典型的生物信息学流程包括以下步骤：

1. 将读数映射到参考：此步骤将 FASTQ 文件中包含的读数与 FASTA 文件中包含的参考基因组对齐。这一步确保数据可以放置在上下文中。
2. 去除重复：此步骤检测表明同一分析片段被测序多次的读数。这些重复没有信息价值，不应被计为额外的证据。
3. 在连接处分割读数：此步骤通过去除 N 同时保持分组信息，将 RNA 读数分割

成外显子片段，并硬修剪任何延伸到内含子区域的序列。此外，该步骤将重新分配 STAR 的映射质量，使其与后续步骤中预期的一致，将质量 255 转换为 60。

4. （可选）碱基质量分数重校准（BQSR）：此步骤修改分配给序列读数数据的单个读数碱基的质量分数。此操作消除了测序方法引起的实验偏差。

5. 变异检测：此步骤识别您的数据相对于参考基因组显示变异的位点，并计算每个样本在该位点的基因型。

4.5.2 RNAseq 使用步骤

（1）将读数映射到参考

运行单个命令以使用 STAR 高效执行对齐，并使用 Sentieon®软件创建 BAM 文件并进行排序：

```
sentieon STAR --runThreadN NUMBER_THREADS --genomeDir STAR_REFERENCE \
  --readFilesIn FASTQ FASTQ2 --readFilesCommand "zcat" \
  --outStd BAM_Unsorted --outSAMtype BAM Unsorted --outBAMcompression 0 \
  --outSAMattrRGline ID:GROUP_NAME SM:SAMPLE_NAME PL:PLATFORM \
  --twopassMode Basic --twopass1readsN -1 --sjdbOverhang READ_LENGTH_MINUS_1 \
  | sentieon util sort -r REFERENCE -o SORTED_BAM -t NUMBER_THREADS -i -
```

STAR 的输入和选项在手册中有描述。

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。我们建议 STAR 和 util 二进制文件使用相同数量的线程。
- **STAR_REFERENCE**：STAR genomeDir 参考 FASTA 文件的位置。您应确保 STAR 所需的所有必要参考数据都在同一位置，并具有一致的命名。

- **REFERENCE**：参考 FASTA 文件的位置。您应确保 FASTA 文件和相应的 FAI 索引文件与 STAR_REFERENCE genomeDir 中的版本和命名约定匹配。
- **FASTQ**：样本 FASTQ 文件的位置。如果数据来自双端测序技术，您还需要输入 FASTQ2 作为相应的配对样本 FASTQ 文件。您需要确保 --readFilesCommand 选项中使用的程序与输入 FASTQ 的压缩状态匹配，即：如果 FASTQ 文件被压缩，您应使用 zcat，而如果 FASTQ 文件未被压缩，您应使用 cat。
- **GROUP_NAME**： 将添加到读组标头行的读组标识符。RG：ID 需要在您计划使用的所有数据集中是唯一的。
- **SAMPLE_NAME**：将添加到读组标头行的样本名称。
- **PLATFORM**：用于测序 DNA 的测序平台名称。可能的选项有：ILLUMINA（当 fastq 文件在 Illumina™机器上产生时）；IONTORRENT（当 fastq 文件在 Life Technologies™ Ion-Torrent™机器上产生时）；ELEMENT（当 fastq 文件在 Element Biosciences™机器上产生时）；DNBSEQ（当 fastq 文件在 MGI™机器上产生时）；ULTIMA（当 fastq 文件在 Ultima Genomics™机器上产生时）。
- **READ_LENGTH_MINUS_1**：您输入数据的读长减 1。
- **SORTED_BAM**：排序后的映射 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。

（2）去除或标记重复

在比对和排序 BAM 文件后，需要运行两个独立的命令来移除或标记重复。

第一个命令收集读段信息，第二个命令执行去重复操作；选项--rmdup 控制是移除重复的 reads(当该选项存在时)，还是仅将其标记为重复。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --rna [--consensus] [--umi_tag XR]
  --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz DEDUPED_BAM
```

这些命令需要以下输入:

- **NUMBER_THREADS**：计算中将使用的计算机线程数。建议不要超过系统中可用的计算核心数。
- **SORTED_BAM**：前一个映射阶段存储结果的位置。
- **SCORE.gz**：临时得分输出文件的位置和文件名。确保两个命令使用相同的文件。
- **DEDUPED_BAM**：去重复 BAM 输出文件的位置和文件名。将创建相应的索引文件(.bai)。

（3）在连接处分割读数

运行单个命令以执行将读数分割成外显子片段并重新分配 STAR 的映射质量。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i DEDUPED_BAM \
  --algo RNASplitReadsAtJunction --reassign_mapq 255:60 SPLIT_BAM
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的相同。
- **DEDUPED_BAM**：前一个去重阶段存储结果的位置。

- **SPLIT_BAM**：包含分割读数的 BAM 输出文件的位置和文件名。将创建相应的索引文件（.bai）。

（4）碱基质量分数重校准（BQSR；可选）

碱基质量分数重校准（BQSR）阶段的用法与 DNAseq 阶段相同；对于此阶段，您可以参考 *第 4.1.2 节* 获取详细的使用说明。

（5）RNA 变异检测

运行单个命令以调用变异并额外应用之前计算的 BQSR。RNA 变异检测可以使用 Haplotyper 算法或 DNAscope 算法完成。对于命令，您应该使用 --trim_soft_clip 选项，由于 RNA 数据的特性，应设置比 DNAseq®变异检测更低的最小 phred 标度置信阈值，建议将 call_conf 设置为 20，emit_conf 设置为 20，而不是默认的 30。

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SPLIT_BAM \
  [-q RECAL_DATA.TABLE] --algo Haplotyper --trim_soft_clip  \
  --call_conf 20 --emit_conf 20 [-d dbSNP] VARIANT_VCF
```

如果您想使用 DNAscope 进行检测，命令将是：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SPLIT_BAM \
  [-q RECAL_DATA.TABLE] --algo DNAscope --trim_soft_clip  \
  --call_conf 20 --emit_conf 20 [-d dbSNP] VARIANT_VCF
```

该命令需要以下输入：

- **NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数量不要超过系统中可用的计算核心数。
- **REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的相同。
- **SPLIT_BAM**：前一个 RNASplitReadsAtJunction 阶段存储结果的位置。
- **VARIANT_VCF**：变异检测输出文件的位置和文件名。将创建相应的索引文件。

该工具将通过使用.gz 扩展名输出压缩文件。

该命令的以下输入是可选的：

- **RECAL_DATA.TABLE**：前一个 BQSR 阶段存储结果的位置。
- **dbSNP**：将用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。

  您只能使用一个 dbSNP 文件。

4.5.3 RNAseq 标准分析脚本

以下是 Sentieon RNAseq 标准的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ***************************************
# Script to perform RNA variant calling
# using a single sample with fastq files
# named 1.fastq.gz and 2.fastq.gz
# ***************************************

set -eu

# Update with the fullpath location of your sample fastq
SM="sample" #sample name
RGID="rg_$SM" #read group ID
PL="ILLUMINA" #or other sequencing platform
READ_LENGTH=101 #the read length for the sample
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1="$FASTQ_FOLDER/rna1.fastq.gz"
FASTQ_2="$FASTQ_FOLDER/rna2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"
#comment if STAR should generate a new genomeDir
STAR_FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta.genomeDir"
#uncomment if you would like to use a bed file
#INTERVAL_FILE="$FASTA_DIR/TruSeq_exome_targeted_regions.b37.bed"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the server
```

```
START_DIR="$PWD/test/RNAseq" #Determine where the output files will be stored




# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/${SM}"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1
cd $WORKDIR
DRIVER_INTERVAL_OPTION="${INTERVAL_FILE:+--interval $INTERVAL_FILE}"

# ****************************************
# 1. Mapping reads with STAR
# ****************************************
if [ -z "$STAR_FASTA" ]; then
  STAR_FASTA="genomeDir"
  # The genomeDir generation could be reused
  mkdir $STAR_FASTA
  $SENTIEON_INSTALL_DIR/bin/sentieon STAR --runMode genomeGenerate \
       --genomeDir $STAR_FASTA --genomeFastaFiles $FASTA --runThreadN $NT || \
       { echo "STAR index failed"; exit 1; }
fi
#perform the actual alignment and sorting
( $SENTIEON_INSTALL_DIR/bin/sentieon STAR --twopassMode Basic --genomeDir $STAR_FAS
TA \
     --runThreadN $NT --outStd BAM_Unsorted --outSAMtype BAM Unsorted \
     --outBAMcompression 0 --twopass1readsN -1 --sjdbOverhang `expr "$READ_LENGTH"
- 1` \
     --readFilesIn $FASTQ_1 $FASTQ_2 --readFilesCommand "zcat" \
     --outSAMattrRGline ID:$RGID SM:$SM PL:$PL || { echo -n 'STAR error'; exit 1; } ) |
 \
     $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam \
```

```
    -t $NT --bam_compression 1 -i - || { echo "STAR alignment failed"; exit 1; }

# ********************************************
# 2. Metrics
# ********************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver $DRIVER_INTERVAL_OPTION -r $FASTA -t $NT
 \
    -i sorted.bam --algo MeanQualityByCycle mq_metrics.txt --algo QualDistribution \
    qd_metrics.txt --algo GCBias --summary gc_summary.txt gc_metrics.txt \
    --algo AlignmentStat --adapter_seq '' aln_metrics.txt --algo InsertSizeMetricAlgo \
    is_metrics.txt || { echo "Metrics failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_metri
cs.txt
$SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metrics.t
xt

# ********************************************
# 3. Remove Duplicate Reads. It is possible
# to remove instead of mark duplicates
# by adding the --rmdup option in Dedup
# ********************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector \
    --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
    --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
    { echo "Dedup failed"; exit 1; }

# ********************************************
# 2a. Coverage metrics
# ********************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    --algo CoverageMetrics coverage_metrics || { echo "CoverageMetrics failed"; exit 1;
 }

# ********************************************
# 4. Split reads at Junction
# ********************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
    --algo RNASplitReadsAtJunction --reassign_mapq 255:60 splitted.bam || \
    { echo "RNASplitReadsAtJunction failed"; exit 1; }
```

```
# ****************************************
# 6. Base recalibration
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver $DRIVER_INTERVAL_OPTION -r $FASTA -t $NT
 \
    -i splitted.bam --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_IN
DELS \
    recal_data.table
$SENTIEON_INSTALL_DIR/bin/sentieon driver $DRIVER_INTERVAL_OPTION -r $FASTA -t $NT
 \
    -i splitted.bam -q recal_data.table --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_M
ILLS \
    -k $KNOWN_INDELS recal_data.table.post
$SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT --algo QualCal --plot \
    --before recal_data.table --after recal_data.table.post recal.csv
$SENTIEON_INSTALL_DIR/bin/sentieon plot QualCal -o recal_plots.pdf recal.csv

# ****************************************
# 7. HC Variant caller for RNA
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver $DRIVER_INTERVAL_OPTION -r $FASTA -t $NT
 \
    -i splitted.bam -q recal_data.table --algo Haplotyper -d $KNOWN_DBSNP \
    --trim_soft_clip --emit_conf=20 --call_conf=20 output-hc-rna.vcf.gz || \
    { echo "Haplotyper failed"; exit 1; }
```

4.6 DNAscope LongRead

Sentieon DNAscope LongRead 是一个用于从长读序列数据中调用胚系 SNV 和 indel 的管道。DNAscope LongRead 管道能够利用较长的读长，使用专门校准的机器学习模型进行快速准确的变异调用。

该小节提供了实现 DNAscope LongRead 管道的 shell 脚本，以及用于 VCF 操作的 Python 脚本。该管道以已进行比对和排序的 BAM 或 CRAM 文件作为输入，并以 VCF(和 gVCF)格式输出变异。

DNAscope LongRead 使用 Sentieon 软件包实现，需要有效的许可证才能使用。请联系 info@insvast.com 获取 Sentieon 软件和评估许可证。

4.6.1 先决条件

要运行管道脚本，您需要使用 Sentieon 软件包 202308 或更高版本。该管道还需要 Python 版本>2.7 或>3.3、bcftools 版本 1.10 或更高版本以及 bedtools。sentieon、python、bcftools 和 bedtools 可执行文件将通过用户的 PATH 环境变量访问。

4.6.2 输入数据要求

（1）比对读数 - PacBio HiFi

该管道接受使用 minimap2 或 pbmm2 比对到参考基因组的 PacBio HiFi 长读。使用 Sentieon minimap2 比对读数时，建议使用 Sentieon 的 HiFi 读数模型。使用开源 minimap2 时，建议使用-x map-hifi。

（2）比对读数 - ONT

该管道接受使用 minimap2 比对到参考基因组的 Oxford Nanopore (ONT)长读。使用 Sentieon minimap2 比对读数时，建议使用 Sentieon 的 ONT 模型。使用开源 minimap2 时，建议使用-x map-ont。

（3）参考基因组

DNAscope LongRead 将相对于高质量参考基因组序列调用样本中存在的变异。除了参考基因组文件外，还需要有 samtools fasta 索引文件(.fai)。我们建议比对到不含替代重叠群的参考基因组。

4.6.3 用法

使用单个命令从 PacBio HiFi 读数调用变异：

```
dnascope_HiFi.sh [-h] -r REFERENCE -i INPUT_BAM -m MODEL_BUNDLE [-d DBSNP_VCF]
[-b DIPLOID_BED] [-t NUMBER_THREADS] [-g] [--] VARIANT_VCF
```

使用单个命令从 ONT 长读调用变异：

```
dnascope_ONT.sh [-h] -r REFERENCE -i INPUT_BAM -m MODEL_BUNDLE [-d DBSNP_VCF]
[-b DIPLOID_BED] [-t NUMBER_THREADS] [-g] [--] VARIANT_VCF
```

Sentieon LongRead 管道需要以下参数：

- **-r REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的参考相同。
- **-i INPUT_BAM**：输入 BAM 或 CRAM 文件的位置。
- **-m MODEL_BUNDLE**：模型包的位置。

Sentieon LongRead 管道接受以下可选参数：

- **-d DBSNP_VCF**：用于标记已知变异的单核苷酸多态性数据库(dbSNP)的位置。仅支持一个文件。提供此文件将使用 dbSNP refSNP ID 号注释变异。

- **-b DIPLOID_BED**：限制变异调用的参考中的区间，采用 BED 文件格式。提供此文件将限制变异调用仅在 BED 文件内的区间进行。
- **-t NUMBER_THREADS**：软件用于运行并行进程的计算线程数。此参数是可选的;如果省略，管道将使用服务器拥有的所有线程。
- **-g**：除 VCF 输出文件外，还以 gVCF 格式输出变异。该工具将输出一个 bgzip 压缩的 gVCF 文件及其对应的索引文件。
- **-h**：打印命令行帮助并退出。

Sentieon LongRead 管道需要以下位置参数：

- **VARIANT_VCF**：变异调用输出的位置和文件名。该工具将输出一个 bgzip 压缩的 VCF 文件及其对应的索引文件。

4.6.4 管道输出

Sentieon LongRead 管道将输出一个 bgzip 压缩文件(.vcf.gz)，其中包含标准 VCFv4.2 格式的变异调用，以及一个 tabix 索引文件(.vcf.gz.tbi)。如果使用-g 选项，管道还将输出一个 bgzip 压缩文件(.g.vcf.gz)，其中包含 gVCF 格式的变异调用，以及一个 tabix 索引文件(.g.vcf.gz.tbi)。

4.6.5 其他注意事项

（1）二倍体变异调用

目前，该管道仅建议用于二倍体生物的样本。对于同时具有二倍体和单倍体染色体的样本，可以使用-b DIPLOID_BED 选项将变异调用限制在二倍体染色体上。

（2）修改

此小节的脚本根据 BSD 2-Clause 许可证提供。鼓励用户修改此文档中的 shell 脚本，以保留管道生成的中间文件、添加额外的处理步骤或修改命令行参数。此文档中的 Python 脚本对管道生成的中间 gVCF 和 VCF 文件进行低级操作。由于这些脚本执行低级数据处理，不建议用户修改这些文件。

4.6.6 DNAscope LongRead PacBio HiFi 分析脚本

以下是 Sentieon DNAscope LongRead 对 PacBio HiFi 数据进行全面的变异检测，包括 SNPs 和 INDELs 的分析脚本示例：

```
#!/bin/sh
#
# The script performs germline variant calling from PacBio HiFi reads with Sentieon DN
Ascope

#### Argument Parsing ####
die()
{
    _ret="${2:-1}"
    test "${_PRINT_HELP:-no}" = yes && print_help >&2
    echo "$1" >&2
    exit "${_ret}"
}


# THE DEFAULTS INITIALIZATION - POSITIONALS
# THE DEFAULTS INITIALIZATION - OPTIONALS
_arg_reference_fasta=
_arg_sample_input=
_arg_model_bundle=
_arg_dbsnp=
_arg_regions=
_arg_threads=
_arg_gvcf="off"


print_help()
{
    printf '%s\n' "Call small variants from PacBio HiFi reads with Sentieon DNAscope"
    printf 'Usage: %s -r <reference> -i <sample-input> -m <model-file> [-d <dbSNP>] [-b
 <regions>] [-t <threads>] [-g] [-h] [--] <output-vcf>\n' "$0"
    printf '\t%s\n' "<output-vcf>: The output VCF file."
    printf '\t%s\n' "-r: The reference FASTA file. (required)"
    printf '\t%s\n' "-i: The aligned BAM or CRAM file. (required)"
    printf '\t%s\n' "-m: The model bundle file. (required)"
    printf '\t%s\n' "-d: dbSNP VCF file. Supplying this file will annotate variants with th
eir dbSNP refSNP ID numbers. (no default)"
    printf '\t%s\n' "-b: Region BED file. Supplying this file will limit variant calling to th
```

```
e intervals inside the BED file. (no default)"
    printf '\t%s\n' "-t: Number of threads/processes to use. (number of available cores)
"
    printf '\t%s\n' "-g: Generate a gVCF output file along with the VCF. (default genera
tes only the VCF)"
    printf '\t%s\n' "-h: Print this help message"
}


parse_commandline()
{
    while getopts 'r:i:m:d:b:t:gh' _key
    do
        case "$_key" in
            r)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_reference_fasta="$OPTARG"
                ;;
            i)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_sample_input="$OPTARG"
                ;;
            m)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_model_bundle="$OPTARG"
                ;;
            d)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_dbsnp="$OPTARG"
                ;;
            b)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_regions="$OPTARG"
                ;;
            t)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_threads="$OPTARG"
                ;;
```

```
                g)
                    _arg_gvcf="on"
                    ;;
                h)
                    print_help
                    exit 0
                    ;;
                *)
                    _PRINT_HELP=yes die "FATAL ERROR: Got an unexpected option '-${_key}'" 1
                    ;;
            esac
        done
}


handle_passed_args_count()
{
    _required_args_string="'output-vcf'"
    test "${_positionals_count}" -ge 1 || _PRINT_HELP=yes die "ERROR: Not enough positional arguments - we require exactly 1 (namely: $_required_args_string), but got only ${_positionals_count}." 1
    test "${_positionals_count}" -le 1 || _PRINT_HELP=yes die "ERROR: There were spurious positional arguments --- we expect exactly 1 (namely: $_required_args_string), but got ${_positionals_count} (the last one was: '${_last_positional}')." 1
}


assign_positional_args()
{
    _shift_for=$1
    _positional_names="_arg_output_vcf "

    shift "$_shift_for"
    for _positional_name in ${_positional_names}
    do
        test $# -gt 0 || break
        eval "$_positional_name=\${1}" || die "Error during argument parsing, possibly an Argbash bug." 1
        shift
    done
}

parse_commandline "$@"
```

```
_positionals_count=$(($# - OPTIND + 1)); _last_positional=$(eval "printf '%s' \"\$$#\""); h
andle_passed_args_count
assign_positional_args "$OPTIND" "$@"
_positionals_count=$(($# - OPTIND + 1)); _last_positional=$(eval "printf '%s' \"\$$#\""); h
andle_passed_args_count

case "$_arg_output_vcf" in
     *.vcf.gz)
          ;;
     *)
          die "The output VCF file is expected to have a '.vcf.gz' suffix" 1
          ;;
esac

if [ -z "$_arg_threads" ]; then
     _arg_threads=$(nproc)
fi

if [ -z "$_arg_reference_fasta" ]; then
  _PRINT_HELP=yes die "Error: the '-r' argument is required" 1
fi

if [ -z "$_arg_sample_input" ]; then
  _PRINT_HELP=yes die "Error: the '-i' argument is required" 1
fi

if [ -z "$_arg_model_bundle" ]; then
  _PRINT_HELP=yes die "Error: the '-m' argument is required" 1
fi

#### Check the PATH for the required executable files ####
for exec_file in bcftools bedtools sentieon; do
     if ! which $exec_file 1>/dev/null 2>/dev/null; then
          echo "Error: no '$exec_file'  executable found in the PATH"
          exit 2
     fi
done

#### Check executables have the required versions ####
VERSION_PATTERN='^v\{0,1\}\([[:digit:]]\{1,\}\).\{0,1\}\([[:digit:]]\{1,\}\)\{0,1\}.\{0,1\}\([[:digit:]]
\{1,\}\)\{0,1\}$'

cmp() {
     num_a="$1"
```

```
    num_b="$2"
    if [ "$num_a" -lt "$num_b" ]; then
        return 1
    elif [ "$num_a" -gt "$num_b" ]; then
        return 2
    fi
    return 0
}

cmp_vers() {
    if [ "$1" = "$2" ];
    then
        return 0
    fi

    a_major=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\1/')
    a_minor=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\2/')
    [ -n "$a_minor" ] || a_minor=0
    a_patch=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\3/')
    [ -n "$a_patch" ] || a_patch=0
    b_major=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\1/')
    b_minor=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\2/')
    [ -n "$b_minor" ] || b_minor=0
    b_patch=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\3/')
    [ -n "$b_patch" ] || b_patch=0

    if cmp "$a_major" "$b_major"; then
        if cmp "$a_minor" "$b_minor"; then
            cmp "$a_patch" "$b_patch"
            return $?
        else
            cmp "$a_minor" "$b_minor"
            return $?
        fi
    else
        cmp "$a_major" "$b_major"
        return $?
    fi
}

SENTIEON_MIN_VERSION=202308
BCFTOOLS_MIN_VERSION=1.10
sentieon_version_string=$(sentieon driver --version)
sentieon_version_string="${sentieon_version_string##sentieon-genomics-}"
```

```
bcftools_version_string=$(bcftools --version | head -n 1)
bcftools_version_string="${bcftools_version_string##bcftools }"

if echo "$sentieon_version_string" | grep -q "$VERSION_PATTERN"; then
    cmp_vers "$SENTIEON_MIN_VERSION" "$sentieon_version_string"
    if [ "$?" -eq "2" ]; then
        echo "Error: the pipeline requires sentieon version '$SENTIEON_MIN_VERSION' o
r later but sentieon '$sentieon_version_string' was found in the PATH"
        exit 2
    fi
else
    echo "Warning: unable to check sentieon version string"
fi

if echo "$bcftools_version_string" | grep -q "$VERSION_PATTERN"; then
    cmp_vers "$BCFTOOLS_MIN_VERSION" "$bcftools_version_string"
    if [ "$?" -eq "2" ]; then
        echo "Error: the pipeline requires bcftools version '$BCFTOOLS_MIN_VERSION' o
r later but bcftools '$bcftools_version_string' was found in the PATH"
        exit 2
    fi
else
    echo "Warning: unable to check bcftools version string"
fi

#### Check that all input files can be found and are not empty
if [ ! -f "$_arg_reference_fasta" ]; then
    echo "Error: cannot find the input reference fasta file, '$_arg_reference_fasta'"
    exit 2
elif [ ! -s "$_arg_reference_fasta" ]; then
    echo "Error: the input reference fasta file is empty, '$_arg_reference_fasta'"
    exit 2
fi

if [ ! -f "$_arg_sample_input" ]; then
    echo "Error: cannot find the input BAM file, '$_arg_sample_input'"
    exit 2
elif [ ! -s "$_arg_sample_input" ]; then
    echo "Error: the input BAM file is empty, '$_arg_sample_input'"
    exit 2
fi

if [ ! -f "$_arg_model_bundle" ]; then
    echo "Error: cannot find the input model bundle file, '$_arg_model_bundle'"
```

```
    exit 2
elif [ ! -s "$_arg_model_bundle" ]; then
    echo "Error: the input model bundle file is empty, '$_arg_model_bundle'"
    exit 2
fi

if [ -n "$_arg_dbsnp" ]; then
    if [ ! -f "$_arg_dbsnp" ]; then
        echo "Error: cannot find the optional dbSNP VCF file, '$_arg_dbsnp'"
        exit 2
    elif [ ! -s "$_arg_dbsnp" ]; then
        echo "Error: the optional dbSNP VCF file is empty, '$_arg_dbsnp'"
        exit 2
    fi
fi

if [ -n "$_arg_regions" ]; then
    if [ ! -f "$_arg_regions" ]; then
        echo "Error: cannot find the optional regions BED file, '$_arg_regions'"
        exit 2
    elif [ ! -s "$_arg_regions" ]; then
        echo "Error: the optional regions BED file is empty, '$_arg_regions'"
        exit 2
    fi
fi

#### Temporary directory ####
tmp_base="/tmp"
if [ -n "$SENTIEON_TMPDIR" ]; then
    tmp_base="$SENTIEON_TMPDIR"
elif [ -n "$TMPDIR" ]; then
    tmp_base="$TMPDIR"
fi
tmp_base="$tmp_base/$$"

tmp_dir="$tmp_base"
while [ -e "$tmp_dir" ]; do
    rand_num=$(echo "" | awk 'BEGIN {srand()} {print int(rand() * 65535)}')
    tmp_dir="${tmp_base}"_"$rand_num"
done
mkdir -p "$tmp_dir"


#### Output files ####
```

```
OUTPUT_BASENAME=$(basename "${_arg_output_vcf%%.vcf.gz}")
TMP_BASE="$tmp_dir"/"$OUTPUT_BASENAME"

#### - Diploid
DIPLOID_TMP_OUT="${TMP_BASE}_diploid_tmp.vcf.gz"
DIPLOID_OUT="${TMP_BASE}_diploid.vcf.gz"
DIPLOID_GVCF="${TMP_BASE}_diploid.g.vcf.gz"

#### - Phasing
PHASED_VCF="${TMP_BASE}_diploid_phased.vcf.gz"
PHASED_BED="${TMP_BASE}_diploid_phased.bed"
PHASED_EXT="${TMP_BASE}_diploid_phased.ext.vcf.gz"
REF_BED="${TMP_BASE}_reference.bed"
UNPHASED_BED="${TMP_BASE}_diploid_unphased.bed"
PHASED_UNPHASED_VCF="${TMP_BASE}_diploid_phased_unphased.vcf.gz"
REPEAT_MODEL="${TMP_BASE}_repeat.model"

#### - Pass2 - haploid
HAP_ONE_TMP="${TMP_BASE}_hap1_tmp.vcf.gz"
HAP_ONE_STD_TMP="${TMP_BASE}_hap1_nohp_tmp.vcf.gz"
HAP_ONE_PATCH="${TMP_BASE}_hap1_patch.vcf.gz"
HAP_ONE_OUT="${TMP_BASE}_hap1.vcf.gz"
HAP_TWO_TMP="${TMP_BASE}_hap2_tmp.vcf.gz"
HAP_TWO_STD_TMP="${TMP_BASE}_hap2_nohp_tmp.vcf.gz"
HAP_TWO_PATCH="${TMP_BASE}_hap2_patch.vcf.gz"
HAP_TWO_OUT="${TMP_BASE}_hap2.vcf.gz"

#### - Pass2 - diploid
DIPLOID_UNPHASED_HP="${TMP_BASE}_diploid_unphased_hp.vcf.gz"
DIPLOID_UNPHASED_PATCH="${TMP_BASE}_diploid_unphased_patch.vcf.gz"
DIPLOID_UNPHASED="${TMP_BASE}_diploid_unphased.vcf.gz"

#### - Merge generated VCF files
OUTPUT_GVCF="${_arg_output_vcf%%.vcf.gz}.g.vcf.gz"



#### Supporting scripts ####
DIR=$(CDPATH='' cd -- "$(dirname -- "$0")" && pwd)
VCF_MOD_PY="${DIR}"/vcf_mod.py
GVCF_COMBINE="${DIR}"/gvcf_combine.py



set -e
#  Un-comment this line for more log information in the Bash shell
```

```
#set -exvuo pipefail


#### Pipeline fuctions ####
dnascope_hp()
{
    bam="$1" model="$2" repeat_model="$3" vcf="$4"
    bed="$5" read_filter="${6:-}" ds_model="${7:-}"
    ds_vcf="${8:-}"

    sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
        --interval "$bed" -i "$bam" ${read_filter:+--read_filter $read_filter} \
        ${ds_vcf:+--algo DNAscope ${_arg_dbsnp:+--dbsnp "$_arg_dbsnp"} --model "$ds_
model" "$ds_vcf"} \
        --algo DNAscopeHP ${_arg_dbsnp:+--dbsnp "$_arg_dbsnp"} \
        --model "$model" --pcr_indel_model "$repeat_model" \
        --min_repeat_count 6 "$vcf"
}

model_apply()
{
    input_vcf="$1"
    output_vcf="$2"
    model="$3"

    sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
        --algo DNAModelApply --model "$model" -v "$input_vcf" "$output_vcf"
}

#### Pipeline ####
# Pass 1
## Call variants
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    ${_arg_regions:+--interval "$_arg_regions"} -i "$_arg_sample_input" \
    ${_arg_gvcf:+--algo DNAscope --model "$_arg_model_bundle"/gvcf_model --emit_mod
e gvcf "$DIPLOID_GVCF"} \
    --algo DNAscope ${_arg_dbsnp:+--dbsnp "$_arg_dbsnp"} \
    --model "$_arg_model_bundle"/diploid_model "$DIPLOID_TMP_OUT"

model_apply "$DIPLOID_TMP_OUT" "$DIPLOID_OUT" "$_arg_model_bundle"/diploid_model

# Phasing
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    -i "$_arg_sample_input" --algo VariantPhaser -v "$DIPLOID_OUT" \
```

```
    --max_depth 1000 --out_bed "$PHASED_BED" --out_ext "$PHASED_EXT" \
    "$PHASED_VCF"
if [ -n "$_arg_regions" ]; then
    bedtools subtract -a "$_arg_regions" -b "$PHASED_BED" > "$UNPHASED_BED" &
else
    (cat "$_arg_reference_fasta".fai | awk -v OFS='\t' '{print $1,0,$2}' > "$REF_BED"; \
        bedtools subtract -a "$REF_BED" -b "$PHASED_BED" > "$UNPHASED_BED") &
fi

## Create the repeat model
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    -i "$_arg_sample_input" --interval "$PHASED_BED" \
    --read_filter PhasedReadFilter,phased_vcf="$PHASED_EXT",phase_select=tag \
    --algo RepeatModel --phased --min_map_qual 1 --min_group_count 10000 \
    --read_flag_mask drop=supplementary --repeat_extension 5 \
    --max_repeat_unit_size 2 --min_repeat_count 6 "$REPEAT_MODEL"

wait
bcftools view -T "$UNPHASED_BED" "$PHASED_VCF" | \
    sentieon util vcfconvert - "$PHASED_UNPHASED_VCF" &

# Pass 2 - call variants on the phased haploid chromosomes
## Call variants
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/haploid_hp_model \
    "$REPEAT_MODEL" "$HAP_ONE_TMP" "$PHASED_BED" \
    "PhasedReadFilter,phased_vcf=$PHASED_EXT,phase_select=1" \
    "$_arg_model_bundle"/haploid_model "$HAP_ONE_STD_TMP"
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/haploid_hp_model \
    "$REPEAT_MODEL" "$HAP_TWO_TMP" "$PHASED_BED" \
    "PhasedReadFilter,phased_vcf=$PHASED_EXT,phase_select=2" \
    "$_arg_model_bundle"/haploid_model "$HAP_TWO_STD_TMP"

## Merge DNAscope and DNAscopeHP VCFs
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" haploid_patch \
    --patch1 "$HAP_ONE_PATCH" --patch2 "$HAP_TWO_PATCH" \
    --hap1 "$HAP_ONE_STD_TMP" --hap2 "$HAP_TWO_STD_TMP" \
    --hap1_hp "$HAP_ONE_TMP" --hap2_hp "$HAP_TWO_TMP"

## Apply the trained model to the patched VCFs
model_apply "$HAP_ONE_PATCH" "$HAP_ONE_OUT" "$_arg_model_bundle"/haploid_model
model_apply "$HAP_TWO_PATCH" "$HAP_TWO_OUT" "$_arg_model_bundle"/haploid_mode
l

# Pass 2 - call variant on the unphased regions
```

```
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/diploid_hp_model \
    "$REPEAT_MODEL" "$DIPLOID_UNPHASED_HP" "$UNPHASED_BED"

## Merge the DNAscope and DNAscopeHP VCFs
wait
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" patch \
    --vcf "$PHASED_UNPHASED_VCF" --vcf_hp "$DIPLOID_UNPHASED_HP" \
    "$DIPLOID_UNPHASED_PATCH"
model_apply "$DIPLOID_UNPHASED_PATCH" "$DIPLOID_UNPHASED" \
    "$_arg_model_bundle"/diploid_model_unphased

## Merge the calls to create the output
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" merge \
    --hap1 "$HAP_ONE_OUT" --hap2 "$HAP_TWO_OUT" --unphased "$DIPLOID_UNPHASE
D" \
    --phased "$PHASED_VCF" --bed "$PHASED_BED" "${_arg_output_vcf}"

if [ "${_arg_gvcf}" = "on" ]; then
    sentieon pyexec "$GVCF_COMBINE" -t "$_arg_threads" "$DIPLOID_GVCF" "${_arg_out
put_vcf}" - | \
        sentieon util vcfconvert - "$OUTPUT_GVCF"
fi

rm -r "$tmp_dir"
```

4.6.7 DNAscope LongRead ONT 分析脚本

以下是 Sentieon DNAscope LongRead 对 Oxford Nanopore Technologies (ONT) 数据进行全面的变异检测，包括 SNPs 和 INDELs 的分析脚本示例：

```
#!/bin/sh
#
# The script performs germline variant calling from ONT long reads with Sentieon DNAs
cope

#### Argument Parsing ####
die()
{
    _ret="${2:-1}"
    test "${_PRINT_HELP:-no}" = yes && print_help >&2
    echo "$1" >&2
    exit "${_ret}"
}


# THE DEFAULTS INITIALIZATION - POSITIONALS
# THE DEFAULTS INITIALIZATION - OPTIONALS
_arg_reference_fasta=
_arg_sample_input=
_arg_model_bundle=
_arg_dbsnp=
_arg_regions=
_arg_threads=
_arg_gvcf="off"


print_help()
{
    printf '%s\n' "Call small variants from ONT long reads with Sentieon DNAscope"
    printf 'Usage: %s -r <reference> -i <sample-input> -m <model-file> [-d <dbSNP>] [-b
 <regions>] [-t <threads>] [-g] [-h] [--] <output-vcf>\n' "$0"
    printf '\t%s\n' "<output-vcf>: The output VCF file."
    printf '\t%s\n' "-r: The reference FASTA file. (required)"
    printf '\t%s\n' "-i: The aligned BAM or CRAM file. (required)"
    printf '\t%s\n' "-m: The model bundle file. (required)"
    printf '\t%s\n' "-d: dbSNP VCF file. Supplying this file will annotate variants with th
eir dbSNP refSNP ID numbers. (no default)"
    printf '\t%s\n' "-b: Region BED file. Supplying this file will limit variant calling to th
```

```
e intervals inside the BED file. (no default)"
    printf '\t%s\n' "-t: Number of threads/processes to use. (number of available cores)
"
    printf '\t%s\n' "-g: Generate a gVCF output file along with the VCF. (default genera
tes only the VCF)"
    printf '\t%s\n' "-h: Print this help message"
}


parse_commandline()
{
    while getopts 'r:i:m:d:b:t:gh' _key
    do
        case "$_key" in
            r)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_reference_fasta="$OPTARG"
                ;;
            i)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_sample_input="$OPTARG"
                ;;
            m)
                test "x$OPTARG" = x && die "Missing value for the argument '-$_key'.
" 1
                _arg_model_bundle="$OPTARG"
                ;;
            d)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_dbsnp="$OPTARG"
                ;;
            b)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_regions="$OPTARG"
                ;;
            t)
                test "x$OPTARG" = x && die "Missing value for the optional argumen
t '-$_key'." 1
                _arg_threads="$OPTARG"
                ;;
```

```
                g)
                    _arg_gvcf="on"
                    ;;
                h)
                    print_help
                    exit 0
                    ;;
                *)
                    _PRINT_HELP=yes die "FATAL ERROR: Got an unexpected option '-${_key}'" 1
                    ;;
            esac
        done
}


handle_passed_args_count()
{
    _required_args_string="'output-vcf'"
    test "${_positionals_count}" -ge 1 || _PRINT_HELP=yes die "ERROR: Not enough positional arguments - we require exactly 1 (namely: $_required_args_string), but got only ${_positionals_count}." 1
    test "${_positionals_count}" -le 1 || _PRINT_HELP=yes die "ERROR: There were spurious positional arguments --- we expect exactly 1 (namely: $_required_args_string), but got ${_positionals_count} (the last one was: '${_last_positional}')." 1
}


assign_positional_args()
{
    _shift_for=$1
    _positional_names="_arg_output_vcf "

    shift "$_shift_for"
    for _positional_name in ${_positional_names}
    do
        test $# -gt 0 || break
        eval "$_positional_name=\${1}" || die "Error during argument parsing, possibly an Argbash bug." 1
        shift
    done
}

parse_commandline "$@"
```

```
_positionals_count=$(($# - OPTIND + 1)); _last_positional=$(eval "printf '%s' \"\$$#\""); h
andle_passed_args_count
assign_positional_args "$OPTIND" "$@"
_positionals_count=$(($# - OPTIND + 1)); _last_positional=$(eval "printf '%s' \"\$$#\""); h
andle_passed_args_count

case "$_arg_output_vcf" in
     *.vcf.gz)
          ;;
     *)
          die "The output VCF file is expected to have a '.vcf.gz' suffix" 1
          ;;
esac

if [ -z "$_arg_threads" ]; then
     _arg_threads=$(nproc)
fi

if [ -z "$_arg_reference_fasta" ]; then
  _PRINT_HELP=yes die "Error: the '-r' argument is required" 1
fi

if [ -z "$_arg_sample_input" ]; then
  _PRINT_HELP=yes die "Error: the '-i' argument is required" 1
fi

if [ -z "$_arg_model_bundle" ]; then
  _PRINT_HELP=yes die "Error: the '-m' argument is required" 1
fi

#### Check the PATH for the required executable files ####
for exec_file in bcftools bedtools sentieon; do
     if ! which $exec_file 1>/dev/null 2>/dev/null; then
          echo "Error: no '$exec_file'  executable found in the PATH"
          exit 2
     fi
done

#### Check executables have the required versions ####
VERSION_PATTERN='^v\{0,1\}\([[:digit:]]\{1,\}\).\{0,1\}\([[:digit:]]\{1,\}\)\{0,1\}.\{0,1\}\([[:digit:]]
\{1,\}\)\{0,1\}$'

cmp() {
     num_a="$1"
```

```
    num_b="$2"
    if [ "$num_a" -lt "$num_b" ]; then
        return 1
    elif [ "$num_a" -gt "$num_b" ]; then
        return 2
    fi
    return 0
}

cmp_vers() {
    if [ "$1" = "$2" ];
    then
        return 0
    fi

    a_major=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\1/')
    a_minor=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\2/')
    [ -n "$a_minor" ] || a_minor=0
    a_patch=$(echo $1 | sed -e 's/"$VERSION_PATTERN"'/\3/')
    [ -n "$a_patch" ] || a_patch=0
    b_major=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\1/')
    b_minor=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\2/')
    [ -n "$b_minor" ] || b_minor=0
    b_patch=$(echo $2 | sed -e 's/"$VERSION_PATTERN"'/\3/')
    [ -n "$b_patch" ] || b_patch=0

    if cmp "$a_major" "$b_major"; then
        if cmp "$a_minor" "$b_minor"; then
            cmp "$a_patch" "$b_patch"
            return $?
        else
            cmp "$a_minor" "$b_minor"
            return $?
        fi
    else
        cmp "$a_major" "$b_major"
        return $?
    fi
}

SENTIEON_MIN_VERSION=202308
BCFTOOLS_MIN_VERSION=1.10
sentieon_version_string=$(sentieon driver --version)
sentieon_version_string="${sentieon_version_string##sentieon-genomics-}"
```

```
bcftools_version_string=$(bcftools --version | head -n 1)
bcftools_version_string="${bcftools_version_string##bcftools }"

if echo "$sentieon_version_string" | grep -q "$VERSION_PATTERN"; then
    cmp_vers "$SENTIEON_MIN_VERSION" "$sentieon_version_string"
    if [ "$?" -eq "2" ]; then
        echo "Error: the pipeline requires sentieon version '$SENTIEON_MIN_VERSION' o
r later but sentieon '$sentieon_version_string' was found in the PATH"
        exit 2
    fi
else
    echo "Warning: unable to check sentieon version string"
fi

if echo "$bcftools_version_string" | grep -q "$VERSION_PATTERN"; then
    cmp_vers "$BCFTOOLS_MIN_VERSION" "$bcftools_version_string"
    if [ "$?" -eq "2" ]; then
        echo "Error: the pipeline requires bcftools version '$BCFTOOLS_MIN_VERSION' o
r later but bcftools '$bcftools_version_string' was found in the PATH"
        exit 2
    fi
else
    echo "Warning: unable to check bcftools version string"
fi

#### Check that all input files can be found and are not empty
if [ ! -f "$_arg_reference_fasta" ]; then
    echo "Error: cannot find the input reference fasta file, '$_arg_reference_fasta'"
    exit 2
elif [ ! -s "$_arg_reference_fasta" ]; then
    echo "Error: the input reference fasta file is empty, '$_arg_reference_fasta'"
    exit 2
fi

if [ ! -f "$_arg_sample_input" ]; then
    echo "Error: cannot find the input BAM file, '$_arg_sample_input'"
    exit 2
elif [ ! -s "$_arg_sample_input" ]; then
    echo "Error: the input BAM file is empty, '$_arg_sample_input'"
    exit 2
fi

if [ ! -f "$_arg_model_bundle" ]; then
    echo "Error: cannot find the input model bundle file, '$_arg_model_bundle'"
```

```
    exit 2
elif [ ! -s "$_arg_model_bundle" ]; then
    echo "Error: the input model bundle file is empty, '$_arg_model_bundle'"
    exit 2
fi

if [ -n "$_arg_dbsnp" ]; then
    if [ ! -f "$_arg_dbsnp" ]; then
        echo "Error: cannot find the optional dbSNP VCF file, '$_arg_dbsnp'"
        exit 2
    elif [ ! -s "$_arg_dbsnp" ]; then
        echo "Error: the optional dbSNP VCF file is empty, '$_arg_dbsnp'"
        exit 2
    fi
fi

if [ -n "$_arg_regions" ]; then
    if [ ! -f "$_arg_regions" ]; then
        echo "Error: cannot find the optional regions BED file, '$_arg_regions'"
        exit 2
    elif [ ! -s "$_arg_regions" ]; then
        echo "Error: the optional regions BED file is empty, '$_arg_regions'"
        exit 2
    fi
fi

#### Temporary directory ####
tmp_base="/tmp"
if [ -n "$SENTIEON_TMPDIR" ]; then
    tmp_base="$SENTIEON_TMPDIR"
elif [ -n "$TMPDIR" ]; then
    tmp_base="$TMPDIR"
fi
tmp_base="$tmp_base/$$"

tmp_dir="$tmp_base"
while [ -e "$tmp_dir" ]; do
    rand_num=$(echo "" | awk 'BEGIN {srand()} {print int(rand() * 65535)}')
    tmp_dir="${tmp_base}"_"$rand_num"
done
mkdir -p "$tmp_dir"


#### Output files ####
```

```
OUTPUT_BASENAME=$(basename "${_arg_output_vcf%%.vcf.gz}")
TMP_BASE="$tmp_dir"/"$OUTPUT_BASENAME"

#### - Diploid
DIPLOID_TMP_OUT="${TMP_BASE}_diploid_tmp.vcf.gz"
DIPLOID_OUT="${TMP_BASE}_diploid.vcf.gz"
DIPLOID_GVCF="${TMP_BASE}_diploid.g.vcf.gz"

#### - Phasing
PHASED_VCF="${TMP_BASE}_diploid_phased.vcf.gz"
PHASED_BED="${TMP_BASE}_diploid_phased.bed"
PHASED_EXT="${TMP_BASE}_diploid_phased.ext.vcf.gz"
REF_BED="${TMP_BASE}_reference.bed"
UNPHASED_BED="${TMP_BASE}_diploid_unphased.bed"
PHASED_PHASED_VCF="${TMP_BASE}_diploid_phased_phased.vcf.gz"
PHASED_UNPHASED_VCF="${TMP_BASE}_diploid_phased_unphased.vcf.gz"
REPEAT_MODEL="${TMP_BASE}_repeat.model"

#### - Pass2 - haploid
HAP_ONE_TMP="${TMP_BASE}_hap1_tmp.vcf.gz"
HAP_ONE_PATCH="${TMP_BASE}_hap1_patch.vcf.gz"
HAP_ONE_OUT="${TMP_BASE}_hap1.vcf.gz"
HAP_TWO_TMP="${TMP_BASE}_hap2_tmp.vcf.gz"
HAP_TWO_PATCH="${TMP_BASE}_hap2_patch.vcf.gz"
HAP_TWO_OUT="${TMP_BASE}_hap2.vcf.gz"

#### - Pass2 - diploid
DIPLOID_UNPHASED_HP="${TMP_BASE}_diploid_unphased_hp.vcf.gz"
DIPLOID_UNPHASED_PATCH="${TMP_BASE}_diploid_unphased_patch.vcf.gz"
DIPLOID_UNPHASED="${TMP_BASE}_diploid_unphased.vcf.gz"

#### - Merge generated VCF files
OUTPUT_GVCF="${_arg_output_vcf%%.vcf.gz}.g.vcf.gz"



#### Supporting scripts ####
DIR=$(CDPATH='' cd -- "$(dirname -- "$0")" && pwd)
VCF_MOD_PY="${DIR}"/vcf_mod.py
GVCF_COMBINE="${DIR}"/gvcf_combine.py



set -e
#  Un-comment this line for more log information in the Bash shell
#set -exvuo pipefail
```

```
#### Pipeline fuctions ####
dnascope_hp()
{
    bam="$1" model="$2" repeat_model="$3" vcf="$4"
    bed="$5" read_filter="${6:-}"

    sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
        --interval "$bed" -i "$bam" ${read_filter:+--read_filter $read_filter} \
        --algo DNAscopeHP ${_arg_dbsnp:+--dbsnp "$_arg_dbsnp"} \
        --model "$model" --pcr_indel_model "$repeat_model" \
        --min_repeat_count 6 "$vcf"
}

model_apply()
{
    input_vcf="$1"
    output_vcf="$2"
    model="$3"

    sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
        --algo DNAModelApply --model "$model" -v "$input_vcf" "$output_vcf"
}

#### Pipeline ####
# Pass 1
## Call variants
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    ${_arg_regions:+--interval "$_arg_regions"} -i "$_arg_sample_input" \
    ${_arg_gvcf:+--algo DNAscope --model "$_arg_model_bundle"/gvcf_model --emit_mod
e gvcf "$DIPLOID_GVCF"} \
    --algo DNAscope ${_arg_dbsnp:+--dbsnp "$_arg_dbsnp"} \
    --model "$_arg_model_bundle"/diploid_model "$DIPLOID_TMP_OUT"

model_apply "$DIPLOID_TMP_OUT" "$DIPLOID_OUT" "$_arg_model_bundle"/diploid_model

# Phasing
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    -i "$_arg_sample_input" --algo VariantPhaser -v "$DIPLOID_OUT" \
    --max_depth 1000 --out_bed "$PHASED_BED" --out_ext "$PHASED_EXT" \
    "$PHASED_VCF"
if [ -n "$_arg_regions" ]; then
    bedtools subtract -a "$_arg_regions" -b "$PHASED_BED" > "$UNPHASED_BED" &
```

```
else
    (cat "$_arg_reference_fasta".fai | awk -v OFS='\t' '{print $1,0,$2}' > "$REF_BED"; \
        bedtools subtract -a "$REF_BED" -b "$PHASED_BED" > "$UNPHASED_BED") &
fi
bcftools view -T "$PHASED_BED" "$PHASED_VCF" | \
    sentieon util vcfconvert - "$PHASED_PHASED_VCF" &

## Create the repeat model
sentieon driver -t "$_arg_threads" -r "$_arg_reference_fasta" \
    -i "$_arg_sample_input" --interval "$PHASED_BED" \
    --read_filter PhasedReadFilter,phased_vcf="$PHASED_EXT",phase_select=tag \
    --algo RepeatModel --phased --min_map_qual 1 --min_group_count 10000 \
    --read_flag_mask drop=supplementary --repeat_extension 5 \
    --max_repeat_unit_size 2 --min_repeat_count 6 "$REPEAT_MODEL"

wait
bcftools view -T "$UNPHASED_BED" "$PHASED_VCF" | \
    sentieon util vcfconvert - "$PHASED_UNPHASED_VCF" &

# Pass 2 - call variants on the phased haploid chromosomes
## Call variants
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/haploid_hp_model \
    "$REPEAT_MODEL" "$HAP_ONE_TMP" "$PHASED_BED" \
    "PhasedReadFilter,phased_vcf=$PHASED_EXT,phase_select=1"
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/haploid_hp_model \
    "$REPEAT_MODEL" "$HAP_TWO_TMP" "$PHASED_BED" \
    "PhasedReadFilter,phased_vcf=$PHASED_EXT,phase_select=2"

## Merge DNAscope and DNAscopeHP VCFs
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" haploid_patch \
    --patch1 "$HAP_ONE_PATCH" --patch2 "$HAP_TWO_PATCH" --phased "$PHASED_PHAS
ED_VCF" \
    --hap1_hp "$HAP_ONE_TMP" --hap2_hp "$HAP_TWO_TMP"

## Apply the trained model to the patched VCFs
model_apply "$HAP_ONE_PATCH" "$HAP_ONE_OUT" "$_arg_model_bundle"/haploid_model
model_apply "$HAP_TWO_PATCH" "$HAP_TWO_OUT" "$_arg_model_bundle"/haploid_mode
l

# Pass 2 - call variant on the unphased regions
dnascope_hp "$_arg_sample_input" "$_arg_model_bundle"/diploid_hp_model \
    "$REPEAT_MODEL" "$DIPLOID_UNPHASED_HP" "$UNPHASED_BED"

## Merge the DNAscope and DNAscopeHP VCFs
```

```
wait
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" patch \
    --vcf "$PHASED_UNPHASED_VCF" --vcf_hp "$DIPLOID_UNPHASED_HP" \
    "$DIPLOID_UNPHASED_PATCH"
model_apply "$DIPLOID_UNPHASED_PATCH" "$DIPLOID_UNPHASED" \
    "$_arg_model_bundle"/diploid_model_unphased

## Merge the calls to create the output
sentieon pyexec "$VCF_MOD_PY" -t "$_arg_threads" merge \
    --hap1 "$HAP_ONE_OUT" --hap2 "$HAP_TWO_OUT" --unphased "$DIPLOID_UNPHASE
D" \
    --phased "$PHASED_VCF" --bed "$PHASED_BED" "${_arg_output_vcf}"

if [ "${_arg_gvcf}" = "on" ]; then
    sentieon pyexec "$GVCF_COMBINE" -t "$_arg_threads" "$DIPLOID_GVCF" "${_arg_out
put_vcf}" - | \
        sentieon util vcfconvert - "$OUTPUT_GVCF"
fi

rm -r "$tmp_dir"
```

4.7 LongReads SV

4.7.1 概述

本节描述了使用 Sentieon® LongReads SV 从 PacBio HiFi 和 Oxford Nanopore 长读序列中调用结构变异(SV)。准确的长读测序技术能够准确发现之前短读方法无法获得的大型 SV 事件。

Sentieon® LongReads SV 能够利用更长的读长，对 PacBio HiFi 和 Oxford Nanopore 长读输入进行快速准确的大型 SV 事件检测和基因型分析。

Sentieon® LongReads SV 以比对的 BAM 或 CRAM 文件作为输入，并以 VCF 格式输出变异。我们建议使用 Sentieon®加速的 Minimap2 和排序来执行高效准确的比对。

目前，Sentieon® LongReads SV 可检测和分型大型 INDEL 事件，并输出高置信度的一致 INDEL 碱基序列。未来版本将添加对其他 SV 事件类型的支持。

如有任何其他问题，请联系 Sentieon® Inc.的技术持：support@insvast.com。

4.7.2 使用 LongReads SV

要运行 Sentieon LongReads SV，请运行以下命令：

```
sentieon driver -t NUMBER_THREADS -r REFERENCE -i INPUT_BAM \
  --algo LongReadSV [--min_sv_size MIN_SV_SIZE] [--min_map_qual MIN_MAP_QUAL]\
  --model MODEL/longreadsv.model  OUT_SV_VCF
```

该命令需要以下参数：

- **-t NUMBER_THREADS**：计算中将使用的计算机线程数。我们建议该数字不要超过系统中可用的计算核心数。
- **-r REFERENCE**：参考 FASTA 文件的位置。您应确保参考与映射阶段使用的

参考相同。

- **-i INPUT_BAM**：输入比对文件应为使用 minimap2 或 pbmm2 比对的 PacBio HiFi 或 Oxford Nanopore 读数的索引 BAM 或 CRAM 文件。
- **--model MODEL**：包含针对 PacBio HiFi 或 Oxford Nanopore 长读预设配置编码的输入模型包文件。您需要选择与输入比对文件平台匹配的模型文件。请参考 Sentieon 的 *https://github.com/Sentieon/sentieon-models* 页面下载最新的 SV 模型。

该命令需要以下位置参数：

- **OUT_SV_VCF**：SV 调用输出的位置和文件名，扩展名为.vcf 或.vcf.gz。将创建相应的索引文件。对于.gz 扩展名，该工具将输出压缩文件。

以下参数是可选的：

- **--min_sv_size MIN_SV_SIZE**：输出的最小 SV 大小(以碱基对为单位)(默认值：40)
- **--min_map_qual MIN_MAP_QUAL**：最小读数映射质量(默认值：20)

4.7.3 评估 LongReads SV 结果

Sentieon®建议使用 Sentieon® hap-eval 基于组装的单倍型对 SV 调用结果进行准确评估和比较。Hap-eval 是一个开源的 VCF 比较引擎，用于结构变异基准测试。可在 *https：//github.com/Sentieon/hap-eval* 上下载。

4.8 Sentieon CLI

sentieon-cli 通过单条命令即可实现 Sentieon® 流程的完整运行。设计初衷是显著提升复杂的多阶段比对及变异检测流程的易用性。

4.8.1 Sentieon cli 安装说明

（1）从 sdist 安装（推荐）

从 GitHub 发布页面 *https://github.com/sentieon/sentieon-cli/releases/* 下载最新的 tar.gz 文件，并使用 pip 安装包:

```
curl -LO https://github.com/sentieon/sentieon-cli/releases/download/v1.1.0/sentieon_cli-1.1.0.tar.gz
pip install sentieon_cli-1.1.0.tar.gz
```

（2）使用 Poetry 安装

如果需要，为项目创建一个新的 Python 虚拟环境：

```
# Create a new venv, if needed
python3 -m venv /path/to/new/virtual/environment/sentieon_cli

# Activate the venv
source /path/to/new/virtual/environment/sentieon_cli/bin/activate
```

sentieon-cli 使用 poetry 进行打包和依赖管理。首先，你需要安装 poetry：

```
pip install poetry
```

克隆此仓库并进入根目录：

```
git clone https://github.com/sentieon/sentieon-cli.git
cd sentieon-cli
```

使用 poetry 将 sentieon-cli 安装到虚拟环境中：

```
poetry install
```

然后你可以从虚拟环境中运行命令：

```
sentieon-cli ...
```

（3）全局参数

sentieon-cli 支持以下全局参数：

- **--verbose**：详细日志记录。
- **--debug**：调试模式，用于更详细的日志记录。

（4）支持的流程

- DNAscope - 用于从短读段数据进行胚系 SNV 和 indel 调用的 DNAscope 流程实现。
- DNAscope LongRead - 用于三代长读段数据进行胚系 SNV 和 indel 调用的 DNAscope LongRead 流程实现。
- DNAscope Hybrid - DNAscope 长短读段混合（Hybrid）流程。
- Pangenome - 利用泛基因组图谱进行短读段比对及变异检测的 DNAscope 流程。

4.8.2 DNAscope

Sentieon® DNAscope 是一个专门针对短读段 DNA 测序数据设计的完整流程，涵盖了从序列比对到胚系细胞变异检测（包括 SNV、SV、Indel 和 CNV）的全过程。通过 sentieon-cli 工具，用户仅需一条命令即可运行完整的 DNAscope 流程。该流程在设计上兼顾了高准确性、高效率以及易用性。

DNAscope 结合了贝叶斯统计方法与机器学习算法，以实现卓越的变异检测精度。它不仅支持全基因组测序，也支持靶向捕获（如全外显子组 WES）文库制备样本。

流程支持输入已比对的 BAM/CRAM，或未比对的 FASTQ、uBAM、uCRAM

格式；流程最终输出 VCF（或 gVCF）格式的变异结果，以及 BAM/CRAM 格式的比对文件。

4.8.2.1 环境配置

（1）必备条件

- Sentieon® 软件包：202308 或更高版本。
- samtools：1.16 或更高版本（用于处理 uBAM/uCRAM 或重新比对）。
- MultiQC：1.18 或更高版本（用于生成质控报告）。
- 上述可执行文件需添加至系统的 PATH 环境变量中。

（2）参考基因组

DNAscope 将相对于高质量的参考基因组序列来检测样本中存在的变异。除了参考基因组文件外，还需要提供 samtools fasta 索引文件（.fai）。序列比对还需要 bwa 索引文件。

我们建议比对到不含备选序列的参考基因组。如果基因组中存在备选序列，请同时提供一个“.alt”文件，以激活 bwa 中的 alt 敏感比对。

4.8.2.2 使用说明

（1）从 FASTQ 进行比对和变异检测

只需运行单条命令，即可完成从 FASTQ 文件开始的比对、预处理以及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope [-h] \
  -r REFERENCE \
```

```
--r1_fastq R1_FASTQ ... \
--r2_fastq R2_FASTQ ... \
--readgroups READGROUPS ... \
-m MODEL_BUNDLE \
[-d DBSNP] \
[-b INTERVAL_FILE] \
[--interval_padding INTERVAL_PADDING] \
[-t NUMBER_THREADS] \
[--pcr_free] \
[-g] \
[--duplicate_marking DUP_MARKING] \
[--assay ASSAY] \
[--consensus] \
[--dry_run] \
[--bam_format] \
SAMPLE_VCF
```

使用 FASTQ 输入时，DNAscope 流程需要以下必选参数：

- **-r REFERENCE**：参考基因组 FASTA 文件的位置。同时需要参考基因组索引文件（“.fai”）以及 bwa 比对索引文件。
- **--r1_fastq R1_FASTQ**：输入的 R1 端 FASTQ 文件。可以多次使用。若没有对应的 R2_FASTQ，则默认为单端数据。请注意，该流程为单样本处理，所有 FASTQ 文件应来自同一个样本。
- **--r2_fastq R2_FASTQ**：输入的 R2 端 FASTQ 文件。可以多次使用。
- **--readgroups READGROUPS**：每个 FASTQ 的读段组信息。流程要求 --r1_fastq 和 --readgroups 参数的数量必须一致。
  - 示例参数：

    --readgroups"@RG\tID:HG002-1\tSM:HG002\tLB:HG002-LB-1\tPL:ILLUMINA"
- **-m MODEL_BUNDLE**：模型包的位置。模型文件可在 Sentieon 官方仓库中找到。

- **SAMPLE_VCF**: 用于存储 SNV 和 Indel 检测结果的输出 VCF 文件位置。流程要求输出文件名必须以 .vcf.gz 为后缀。不带后缀的文件路径将用作其他输出文件的基准名称。

DNAscope 流程支持以下可选参数：

- **-d DBSNP**: 单核苷酸多态性数据库（dbSNP）的位置，用于在 VCF 中标记已知变异。支持 VCF 或 bgzip 压缩的 VCF 格式。仅支持输入一个文件。提供此文件将通过 dbSNP 的 refSNP ID 编号对变异进行注释。需要提供 VCF 索引文件。
- **-b INTERVAL_FILE**: 指定参考基因组上的区间以限制变异检测范围，格式为 BED。提供此文件后，变异检测将仅限于 BED 文件内的区间。若未提供，软件将处理全基因组。
- **--interval_padding INTERVAL_PADDING**: 为输入区间的边缘添加指定碱基数的缓冲。默认值为 0。
- **-t NUMBER_THREADS**: 软件运行并行任务时使用的计算线程数。此参数可选；若省略，流程将使用服务器现有的全部线程。
- **--pcr_free**: 使用 --pcr_indel_model NONE 模式进行变异检测，适用于 PCR-free（无 PCR 扩增）文库。系统仍会进行去重复以识别光学重复。
- **-g**: 在输出 VCF 文件的同时，输出 gVCF 格式的变异结果。工具将输出一个带有相应索引文件的 bgzip 压缩 gVCF。
- **--duplicate_marking DUP_MARKING**: 重复序列标记设置。markdup 为标记重复读段；rmdup 为移除重复读段；none 为跳过重复标记。默认设置为 markdup。

- **--assay ASSAY**：用于指标收集的实验类型设置：WGS（全基因组）或 WES（全外显子组）。默认设置为 WGS。
- **--consensus**：在标记重复过程中生成共识读段。
- **-h**：打印命令行帮助信息并退出。
- **--dry_run**：打印流程命令，但不实际执行。
- **--bam_format**：输出比对文件时使用 BAM 格式而非 CRAM 格式。

（2）从 uBAM 或 uCRAM 进行比对和变异检测

只需运行单条命令，即可完成从 uBAM 或 uCRAM 文件开始的比对、预处理以及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope [-h] \
  -r REFERENCE \
  -i SAMPLE_INPUT ... \
  --align \
  [--input_ref INPUT_REF] \
  -m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b BED] \
  [--interval_padding INTERVAL_PADDING] \
  [-t NUMBER_THREADS] \
  [--pcr_free] \
  [-g] \
  [--duplicate_marking DUP_MARKING] \
  [--assay ASSAY] \
  [--consensus] \
  [--dry_run] \
  [--bam_format] \
  SAMPLE_VCF
```

使用 uBAM 或 uCRAM 作为输入时，DNAscope 流程需要以下新增参数：

- **-i SAMPLE_INPUT**：uBAM 或 uCRAM 格式的输入样本文件。在 -i 参数后依次列出多个文件即可输入一个或多个文件。
- **--align**：指示流程对比对输入读段进行比对。

DNAscope 流程支持以下新增可选参数：

- **--input_ref INPUT_REF**：用于解码输入文件的参考基因组 FASTA 文件。在使用 uCRAM 输入时必须提供此参数。该文件可以与 -r 参数指定的参考基因组不同。

（3）从坐标排序的 BAM 或 CRAM 进行比对和变异检测

只需运行单条命令，即可完成从（已完成坐标排序的）BAM 或 CRAM 文件开始的比对、预处理以及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope [-h] \
  -r REFERENCE \
  -i SAMPLE_INPUT ... \
  --collate_align \
  [--input_ref INPUT_REF] \-m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b BED] \
  [--interval_padding INTERVAL_PADDING] \
  [-t NUMBER_THREADS] \
  [--pcr_free] \
  [-g] \
  [--duplicate_marking DUP_MARKING] \
  [--assay ASSAY] \
  [--consensus] \
  [--dry_run] \
  [--bam_format] \
  SAMPLE_VCF
```

使用 BAM 或 CRAM 作为输入时，DNAscope 流程需要以下新增参数：

- **--collate_align**：指示流程先对输入读段进行整理，然后再进行比对。

（4）从已排序的 BAM 或 CRAM 进行变异检测

只需运行单条命令，即可完成从（已排序的）BAM 或 CRAM 文件开始的预处理，以及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope [-h] \
  -r REFERENCE \
  -i SAMPLE_INPUT ... \
```

```
-m MODEL_BUNDLE \
[-d DBSNP] \
[-b BED] \
[--interval_padding INTERVAL_PADDING] \
[-t NUMBER_THREADS] \
[--pcr_free] \
[-g] \
[--duplicate_marking DUP_MARKING] \
[--assay ASSAY] \
[--consensus] \
[--dry_run] \
[--bam_format] \
SAMPLE_VCF
```

如果不提供 --align 和 --collate_align 参数，流程将指示系统直接从输入的读段中检测变异。

4.8.2.3 流程输出

（1）输出文件列表

当使用默认参数处理全基因组 FASTQ 数据且指定的输出文件名为 sample.vcf.gz 时，会生成以下文件：

- **sample.vcf.gz**：在 -b 参数指定的 BED 文件区域（或全基因组）内检测到的 SNV 和 Indel 变异结果。
- **sample_deduped.cram** 或 **sample_deduped.bam**：由输入 FASTQ 生成的经比对、坐标排序及去重标记后的读段数据。
- **sample_svs.vcf.gz**：由 DNAscope 和 SVSolver 检测到的 SV 结果。
- **sample_metrics**：包含该样本所有 QC 指标的目录。
- **sample_metrics/coverage***：处理样本的覆盖度指标（仅适用于 WGS 样本）。

- **sample_metrics/{sample}.txt.alignment_stat.txt**：来自 AlignmentStat 算法的比对统计指标。
- **sample_metrics/{sample}.txt.base_distribution_by_cycle.txt**：来自 BaseDistributionByCycle 算法的逐循环碱基分布指标。
- **sample_metrics/{sample}.txt.dedup_metrics.txt**：来自 Dedup 算法的去重统计指标。
- **sample_metrics/{sample}.txt.gc_bias***：来自 GCBias 算法的 GC 偏好指标（仅适用于 WGS 样本）。
- **sample_metrics/{sample}.txt.insert_size.txt**：来自 InsertSizeMetricAlgo 算法的文库插入片段大小指标。
- **sample_metrics/{sample}.txt.mean_qual_by_cycle.txt**：来自 MeanQualityByCycle 算法的逐循环平均质量值指标。
- **sample_metrics/{sample}.txt.qual_distribution.txt**：来自 QualDistribution 算法的质量值分布指标。
- **sample_metrics/{sample}.txt.wgs.txt**：来自 WgsMetricsAlgo 算法的全基因组统计指标（仅适用于 WGS 样本）。
- **sample_metrics/{sample}.txt.hybrid-selection.txt**：来自 HsMetricAlgo 算法的杂交捕获（如全外显子组）统计指标。
- **sample_metrics/multiqc_report.html**：由 MultiQC 汇总并生成的交互式 QC 指标报告。

4.8.3 DNAscope LongRead

Sentieon® DNAscope LongRead 是一套专门用于长读段测序数据的比对及胚系变异检测（包括 SNV、SV、CNV 和 Indel）的分析流程。DNAscope LongRead 流程利用长读段的长度优势，配合经过专门校准的机器学习模型，能够实现快速且准确的变异检测。sentieon-cli 通过单条命令提供了 DNAscope LongRead 流程的完整实现。该流程旨在提供卓越的准确性、高效性以及易用性。

该流程支持的输入格式包括：已比对的 BAM 或 CRAM 文件，或者未比对的 FASTQ、uBAM 或 uCRAM 文件。流程将输出 VCF（或 gVCF）格式的变异结果，以及 BAM 或 CRAM 格式的比对文件。

4.8.3.1 环境配置

（1）必备条件

- Sentieon® 软件包：202308.01 或更高版本。
- Python：3.8 或更高版本。
- bcftools：1.10 或更高版本（用于 SNV 和 Indel 检测流程）。
- bedtools：用于 SNV 和 Indel 检测流程。
- samtools：1.16 或更高版本（用于 uBAM/uCRAM 格式的比对或对已比对数据的重新比对）。
- mosdepth：0.2.6 或更高版本（用于长读段数据的覆盖度指标统计）。
- hificnv：1.0.0 或更高版本（用于 CNV 检测）。

上述可执行文件均需添加至用户的 PATH 环境变量中，以便程序调用。

（2）参考基因组

DNAscope LongRead 将基于高质量的参考基因组序列来检测样本中的变异。除参考基因组文件外，还需提供配套的 samtools fasta 索引文件（.fai）和序列字典文件（.dict）。

我们建议对比对到不含替代 Contig 的参考基因组上。

4.8.3.2 使用说明

（1）从 FASTQ 进行比对和变异检测

只需运行单条命令，即可完成对 FASTQ 格式的 PacBio HiFi 或 ONT 读段的比对，以及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope-longread [-h] \
  -r REFERENCE \
  --fastq INPUT_FASTQ ... \
  --readgroups READGROUP ... \
  -m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b DIPLOID_BED] \
  [-t NUMBER_THREADS] \
  [-g] \--tech HiFi|ONT \
  [--haploid_bed HAPLOID_BED] \
  [--cnv_excluded_regions CNV_EXCLUDE_BED] \
  SAMPLE_VCF
```

使用 FASTQ 作为输入时，DNAscope LongRead 流程需要以下必选参数：

- **-r REFERENCE**：参考基因组 FASTA 文件的路径。同时需要配套的参考基因组索引文件（“.fai”）。
- **--fastq INPUT_FASTQ**：FASTQ 格式的输入样本文件。在 --fastq 参数后可以传入多个文件路径以同时输入一个或多个文件。
- **--readgroups READGROUP**：读段数据的读段组信息。若输入数据为

FASTQ 格式，则必须提供此参数。该参数需要完整的 Readgroup 字符串，这些字符串将通过 -R 参数传递给 minimap2 比对软件。

- 参数示例：--readgroups '@RG\tID:foo\tSM:bar'。

- **-m MODEL_BUNDLE**：模型包的路径。模型文件可在 Sentieon 官方模型仓库中找到。
- **--tech HiFi|ONT**：生成读段时所使用的测序技术。支持的参数值为 ONT 或 HiFi。
- **SAMPLE_VCF**：用于存储 SNV 和 Indel 检测结果的输出 VCF 文件路径。流程要求输出文件名必须以 ".vcf.gz" 为后缀。不带后缀的文件路径将作为其他输出文件的基准名称。

Sentieon® LongRead 流程支持以下可选参数：

- **-d DBSNP**：单核苷酸多态性数据库（dbSNP）的路径，用于在 VCF（.vcf 或 .vcf.gz 格式）中标记已知变异。仅支持单个文件。提供此文件后，变异结果将标注其对应的 dbSNP refSNP ID 编号。需要提供 VCF 索引文件。
- **-b DIPLOID_BED**：参考基因组上的特定区间（BED 格式），用于限制二倍体变异检测的范围。提供此文件将使二倍体变异检测仅在 BED 记录的区间内进行。
- **--haploid_bed HAPLOID_BED**：参考基因组上的特定区间（BED 格式），用于限制单倍体变异检测的范围（常用于 X/Y 性染色体或线粒体）。
- **--cnv_excluded_regions**：需在 CNV 检测中排除的区域 BED 文件，该文件将传递给 hificnv。更多详情请参考 hificnv 官方文档。
- **-t NUMBER_THREADS**：软件运行并行任务时使用的计算线程数。此参数为

可选；若省略，流程将默认使用服务器现有的全部可用线程。

- **-g**：除主 VCF 输出文件外，同时输出 gVCF 格式的变异结果。工具将生成一个经过 bgzip 压缩的 gVCF 文件及其对应的索引文件。
- **-h**：打印命令行帮助信息并退出。
- **--dry_run**：打印流程实际会运行的指令，但不实际执行。

（2）从 uBAM、uCRAM、BAM 或 CRAM 进行比对和变异检测

只需运行单条命令，即可对 uBAM、uCRAM、BAM 或 CRAM 格式的 PacBio HiFi 或 ONT 读段进行比对及 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope-longread [-h] \
  -r REFERENCE \
  -i SAMPLE_INPUT ... \
  --align \-m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b DIPLOID_BED] \
  [-t NUMBER_THREADS] \
  [-g] \--tech HiFi|ONT \
  [--haploid_bed HAPLOID_BED] \
  [--cnv_excluded_regions CNV_EXCLUDE_BED] \
  [--input_ref INPUT_REF] \
  SAMPLE_VCF
```

使用 uBAM、uCRAM、BAM 或 CRAM 作为输入时，DNAscope LongRead 流程需要以下新增参数：

- **-i SAMPLE_INPUT**：uBAM 或 uCRAM 格式的输入样本文件。在 -i 参数后可以传入多个文件路径以同时输入一个或多个文件。
- **--align**：使用 Sentieon® 版本的 minimap2 将输入读段重新比对到参考基因组上。

DNAscope LongRead 流程支持以下新增可选参数：

- **--input_ref INPUT_REF**：用于解码（还原）输入文件的参考基因组 FASTA

文件。在使用 uCRAM 或 CRAM 输入时必须提供此参数。该文件可以与 -r 参数指定的参考基因组不同。

(3) 从 BAM 或 CRAM 进行变异检测

只需运行单条命令，即可对 BAM 或 CRAM 格式的 PacBio HiFi 或 ONT 读段进行 SNV、Indel 和 SV 的检测：

```
sentieon-cli dnascope-longread [-h] \
  -r REFERENCE \
  -i SAMPLE_INPUT ... \
  -m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b DIPLOID_BED] \
  [-t NUMBER_THREADS] \
  [-g] \--tech HiFi|ONT \
  [--haploid_bed HAPLOID_BED] \
  [--cnv_excluded_regions CNV_EXCLUDE_BED] \
  SAMPLE_VCF
```

注意：若不提供 --align 参数，流程将指示系统直接基于输入的读段（即跳过比对步骤）进行变异检测。

4.8.3.3 流程输出

在处理 FASTQ 数据，或带有 --align 参数处理 uBAM、uCRAM、BAM、CRAM 文件时，会生成以下文件：

- **sample.vcf.gz**：在 -b DIPLOID_BED 文件定义的基因组区域内检测到的 SNV 和 Indel 变异结果。
- **sample.sv.vcf.gz**：由 Sentieon® LongReadSV 工具检测到的结构变异（SV）结果。
- **sample_mm2_sorted_fq_*.cram**：由输入 FASTQ 文件生成的经比对及坐

标排序后的读段数据。

- **sample_mm2_sorted_*.cram**：由输入的 uBAM、uCRAM、BAM 或 CRAM 文件生成的经比对及坐标排序后的读段数据。
- **sample.hificnv**：HiFiCNV 输出文件的基准名称（仅在使用 HiFi 数据输入时生成）。

4.8.3.4 其他注意事项

（1）二倍体与单倍体变异检测

默认流程推荐用于二倍体生物样本。对于同时包含二倍体和单倍体染色体的样本（如人类男性样本），可以使用 -b DIPLOID_BED 选项将二倍体检测限制在常染色体，并使用 --haploid_bed HAPLOID_BED 参数在单倍体染色体上执行变异检测。

本仓库的 /data 文件夹中提供了人类 hg38 参考基因组（针对男性样本）的二倍体和单倍体 BED 文件示例。

（2）脚本修改说明

本仓库中的脚本遵循 BSD 2-Clause 开源协议。

位于 sentieon_cli/scripts 文件夹下的 Python 脚本会对流程生成的中间 gVCF 和 VCF 文件进行底层处理。由于这些脚本涉及复杂的数据底层操作，不建议用户自行修改这些脚本文件。

4.8.4 DNAscope Hybrid

Sentieon® DNAscope Hybrid 是一套利用同一样本的短读段与长读段组合数据进行胚系变异检测的分析流程。DNAscope Hybrid 能够结合短读段（高准确度）与长读段（长跨度、高结构解析力）技术的各自优势，在处理复杂基因组区域（如高重复序列、伪基因区）时，混合流程能显著降低假阴性并提升变异检测的完整性。sentieon-cli 通过单条命令提供了 DNAscope Hybrid 流程的完整实现。

该流程要求同时输入短读段和长读段数据，支持以下格式：

- 未比对的短读段数据：Gzip 压缩格式的 FASTQ 文件。
- 已比对的短读段数据：BAM 或 CRAM 格式文件。
- 未比对的长读段数据：uBAM 或 uCRAM 格式文件。
- 已比对的长读段数据：BAM 或 CRAM 格式文件。

默认情况下，该流程将生成以下输出文件：

- 小变异（SNV 和 Indel）：VCF 格式。
- 结构变异（SV）：VCF 格式。
- 拷贝数变异（CNV）：VCF 格式。

注：若使用未比对的读段作为输入，流程还会输出比对后的 BAM 或 CRAM 文件。

4.8.4.1 环境配置

（1）必备条件

- Sentieon® 软件包：202503.01 或更高版本。

- Python：3.8 或更高版本。
- bcftools：1.10 或更高版本。
- Bedtools
- MultiQC：1.18 或更高版本（用于生成指标报告）。
- samtools：1.16 或更高版本。
- mosdepth：0.2.6 或更高版本（用于收集长读段数据的覆盖度指标）。

上述可执行文件均需添加至用户的 PATH 环境变量中，以便系统调用。

（2）参考基因组

DNAscope Hybrid 将基于 FASTA 格式的高质量参考基因组序列进行变异检测。除参考基因组外，还需提供配套的 samtools 索引文件（.fai）。短读段比对还需要 BWA 索引文件。

我们建议对比对到不含替代 Contig 的参考基因组上。如果参考基因组中包含替代 Contig 且流程需要执行短读段比对，请务必提供 “.alt” 文件以激活 BWA 的 Alt-aware（替代序列感知）比对模式。

4.8.4.2 使用说明

（1）从已比对的短读段与长读段数据进行胚系变异检测

只需运行单条命令，即可从已比对的短、长读段中检测 SNV、Indel、SV 和 CNV：

```
sentieon-cli dnascope-hybrid \
  -r REFERENCE \
  --sr_aln SR_ALN [SR_ALN ...] \
  --lr_aln LR_ALN [LR_ALN ...] \
  -m MODEL_BUNDLE \
  [-d DBSNP] \
```

```
[-b DIPLOID_BED] \
[-t NUMBER_THREADS] \
sample.vcf.gz
```

DNAscope Hybrid 流程需要以下必选参数：

- **-r REFERENCE**：参考基因组 FASTA 文件的路径。同时需要配套的参考基因组索引文件（“.fai”）。
- **--sr_aln**：BAM 或 CRAM 格式的输入短读段数据。在 --sr_aln 参数后传入多个文件路径即可同时输入一个或多个文件。
- **--lr_aln**：BAM 或 CRAM 格式的输入长读段数据。在 --lr_aln 参数后传入多个文件路径即可同时输入一个或多个文件。
- **-m MODEL_BUNDLE**：模型包的路径。模型文件可在 Sentieon 官方模型仓库（models29）中找到。
- **sample.vcf.gz**：用于存储 SNV 和 Indel 检测结果的输出 VCF 文件路径。流程要求输出文件名必须以 “.vcf.gz” 为后缀。

DNAscope Hybrid 流程支持以下可选参数：

- **-d DBSNP**：单核苷酸多态性数据库（dbSNP）的路径，用于在 VCF（.vcf 或 .vcf.gz 格式）中标记已知变异。仅支持单个文件。提供此文件后，变异结果将标注其对应的 dbSNP refSNP ID 编号。需要提供 VCF 索引文件。
- **-b DIPLOID_BED**：参考基因组上的特定区间（BED 格式），用于限制二倍体变异检测的范围。提供此文件将使二倍体变异检测仅在 BED 记录的区间内进行。
- **-t NUMBER_THREADS**：软件运行并行任务时使用的计算线程数。此参数为可选；若省略，流程将默认使用服务器现有的全部可用线程。

- **-h**：打印命令行帮助信息并退出。
- **--dry_run**：打印流程实际会运行的指令，但不实际执行。

（2）从未比对的短读段与长读段数据进行胚系变异检测

只需运行单条命令，即可从未比对的短读段和长读段中检测 SNV、Indel、SV 和 CNV：

```
sentieon-cli dnascope-hybrid \
  -r REFERENCE \
  --sr_r1_fastq SR_R1_FQ [SR_R1_FQ ...] \
  --sr_r2_fastq SR_R2_FQ [SR_R2_FQ ...] \
  --sr_readgroups SR_READGROUP [SR_READGROUP ...] \
  --lr_aln LR_ALN [LR_ALN ...] \
  --lr_align_input \
  -m MODEL_BUNDLE \
  [-d DBSNP] \
  [-b DIPLOID_BED] \
  [-t NUMBER_THREADS] \
  sample.vcf.gz
```

DNAscope Hybrid 流程需要以下必选参数：

- **--sr_r1_fastq**：Gzip 压缩格式的输入 R1 端短读段数据。在参数后传入多个文件路径即可同时输入一个或多个文件。
- **--sr_r2_fastq**：Gzip 压缩格式的输入 R2 端短读段数据。数量需与 R1 端对应。
- **--sr_readgroups**：每个短读段 FASTQ 的读段组信息。流程要求此参数的数量必须与 --sr_r1_fastq 一致。
  - 示例：--sr_readgroups "@RG\tID:HG002-1\tSM:HG002\tLB:HG002-LB-1\tPL:ILLUMINA"
- **--lr_aln**：uBAM 或 uCRAM 格式的输入长读段数据。可传入一个或多个文件。

- **--lr_align_input**：指示流程对比对输入的长读段进行比对。

DNAscope Hybrid 流程支持以下可选参数：

- **--sr_duplicate_marking**：短读段的重复序列标记设置。markdup 为标记重复；rmdup 为移除重复；none 为跳过。默认设置为 markdup。
- **--lr_input_ref**：用于解码输入长读段文件的参考 FASTA 文件。在使用长读段 uCRAM 或 CRAM 输入时必须提供。该文件可与 -r 参数指定的参考基因组不同。
- **--bam_format**：输出的比对文件使用 BAM 格式而非 CRAM 格式。

4.8.4.3 流程输出

（1）输出文件列表

DNAscope Hybrid 流程会输出以下文件：

- sample.vcf.gz：在 -b DIPLOID_BED 定义的基因组区域内的 SNV 和 Indel 变异结果。
- sample.sv.vcf.gz：来自 Sentieon® LongReadSV 工具的结构变异结果。
- sample.cnv.vcf.gz：来自 Sentieon® CNVscope 工具的拷贝数变异结果。
- sample_deduped.cram：由输入 FASTQ 生成的经比对、坐标排序及去重后的短读段数据。
- sample_mm2_sorted_*.cram：由输入长读段文件生成的经比对及坐标排序后的长读段数据。
- sample_metrics：包含该样本所有 QC 指标的目录。

4.8.4.4 故障排除（Troubleshooting）

流程报错：输入文件具有不同的 RG-SM 标签（different RG-SM tag）

当流程检测到输入文件 (短读段和长读段) 中包含不一致的 Readgroup SM (样本名） 标签时，会触发此错误。 要修正这个错误请使用 --rgsm 参数。该参数会在变异检测过程中统一调整输入文件的 SM 标签。需要注意，使用此参数后，输入文件中的所有读段都将被视为同一来源参与变异检测。

4.9 去重复与 UMI 处理

本章节将详细介绍如何使用 Sentieon® Genomics 工具移除 PCR 重复序列。该处理流程通过两个独立的指令协同完成，首先由系统收集读段信息，随后再执行具体的去重复操作。在此过程中，LocusCollector 模块提供的 --consensus 选项起到了关键作用，它负责控制系统是否输出 PCR 重复序列的共识序列。此外，针对适用唯一分子标识符（UMI）标签的数据，用户可以通过为 LocusCollector 开启 --umi_tag 选项，从而激活具有条形码感知能力的去重复功能。

4.9.1 非共识去重复

在“非共识去重复”模式下，系统会从一组重复读段中筛选出一个最具代表性的读段作为主要读段。

4.9.1.1 不带 UMI 的非共识去重复

此工作流的运行结果与 Picard MarkDuplicates 的默认输出一致。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz \
  --metrics DEDUP_METRIC_TXT DEDUPED_BAM
```

此外，还有一种特殊的三步走去重复流程，可用于同时标记主要读段和非主要读段。但请注意，该工作流仅适用于不带 UMI 的非共识去重复模式。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
--algo LocusCollector --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
--algo Dedup --score_info SCORE.gz --output_dup_read_name \
--metrics DEDUP_METRIC_TXT TMP_DUP_QNAME.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
--algo Dedup --dup_read_name TMP_DUP_QNAME.gz \
```

```
DEDUPED_BAM
```

4.9.1.2 带 UMI 的非共识去重复

该工作流除了利用读段或读段对的 5' 端位置信息外，还结合了 UMI 信息来判定 PCR 重复序列。在操作中，需通过 LocusCollector 的 --umi_tag 选项来指定逻辑 UMI 标签。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --umi_tag XR --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz \
  --metrics DEDUP_METRIC_TXT DEDUPED_BAM
```

4.9.2 基于共识去重复

在基于共识的去重复模式下，系统会从一组重复读段中生成一条单一的共识读段。这一过程能够纠正 PCR 扩增和测序过程中引入的错误，并重新估算每个位点的碱基质量得分，以反映共识读段中碱基判定错误的最新概率。需要注意的是，在完成基于共识的去重复后，不应再进行额外的碱基质量校准。

若要开启此功能，需在 LocusCollector 中设置 --consensus 选项。此外，在执行后续的 Dedup 指令时，现在必须提供参考基因组的 FASTA 文件。

注意：在基于共识的去重复处理后，共识读段的配对信息可能会出现偏差。虽然 Sentieon® 变异检测软件并不依赖此类配对信息，但其他生物信息学软件可能会用到。如果您的后续分析需要准确的配对信息，可以在完成去重复后使用 samtools fixmate 工具对读段的配对信息进行更新。

```
samtools collate -@ NUMBER_THREADS -Ou DEDUPED_BAM \
  | samtools fixmate --reference REFERENCE -@ NUMBER_THREADS - - \
  | sentieon util sort -r REFERENCE -o FIXMATE_BAM -t NUMBER_THREADS -i -
```

4.9.2.1 不带 UMI 的基于共识去重复

在不使用 UMI 的情况下，该工作流仅依据比对后的基因组坐标对测序读段进行聚类分析。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --consensus --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz \
  --metrics DEDUP_METRIC_TXT DEDUPED_BAM
```

4.9.2.2 带 UMI 的基于共识去重复

在包含 UMI 的情况下，该工作流将同时结合比对坐标及其对应的 UMI 标签对测序读段进行聚类。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --consensus --umi_tag XR --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz \
  --metrics DEDUP_METRIC_TXT DEDUPED_BAM
```

（1）UMI 条形码纠错

系统会根据 UMI 条形码之间的编辑距离自动纠正其中的条形码错误。如果需要禁用此功能，可以在 LocusCollector 中使用 --umi_ecc_dist 0 选项。

（2）RNA 测序数据

当处理使用 STAR 软件比对的 RNA 测序数据时，需在 LocusCollector 中设置 --rna 选项。

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --rna [--consensus] [--umi_tag XR] --fun score_info SCORE.gz
sentieon driver -t NUMBER_THREADS -r REFERENCE -i SORTED_BAM \
  --algo Dedup [--rmdup] --score_info SCORE.gz DEDUPED_BAM
```

4.9.3 附录

4.9.3.1 确定读段结构并提取 UMI 条形码序列

作为分析的第一步，您需要从输入读段中提取条形码序列。该操作通过执行 sentieon umi extract 命令完成，该命令会从读段中提取条形码序列信息并将其添加到读段描述信息中。在进行 UMI 标签提取之前，应先从输入读段中去除接头序列，这一步骤可由其他第三方工具完成。

umi extract 命令的输出结果为 FASTQ 格式，并采用 R1 和 R2 读段交错的方式排列。默认情况下，提取命令的输出将发送至标准输出，除非使用 -o 选项另行指定输出路径。

umi extract 命令的具体语法如下：

```
sentieon umi extract [options] read_structure fastq1 [fastq2] [fastq3]

Options:
  -o        Output file (default: stdout)
  -d        Turn on duplex mode
  --umi_tag        Logic umi tag (default 'XR')
  --output_format          Output format FASTQ or SAM (default 'FASTQ')
```

umi extract 命令的第一个参数用于定义读段结构。对于双端测序数据，需要指定两个由逗号“,”分隔的读段结构。

读段结构由“数字+操作符”的组合进行定义。其中数字可以是任何数字，或者使用“+”来表示读段的末尾。可选的操作符包括：

- T：模板序列（即目标插入片段）。
- B：细胞条形码序列。
- M：分子条形码序列。
- S：应忽略的碱基序列。

-d 选项用于提取双链 UMI 并标记其来源链。双链 UMI 提取要求两条链具有完全一致的读段结构。

在下方的命令中，将演示如何从双端测序读段中提取单端 UMI 的操作。在本例中，配对读段的 R1 端包含一个 8bp 的分子条形码，随后是一个 12bp 的间隔序列，最后是模板序列；而 R2 端仅包含模板序列。配对后的读段将以交错格式写入输出文件。请注意，在该示例中，输出结果通过管道传输给了 gzip 以生成压缩的 FASTQ 文件。通常情况下，我们建议将输出直接通过管道传输给下一步操作（即 Sentieon® bwa mem 比对）。

```
sentieon umi extract "8M12S+T,+T" \
  sample_R1.fastq.gz \
  sample_R2.fastq.gz | \
  gzip -c \
  > sample_extracted_pair.fastq.gz
```

下方的命令演示了双链 UMI 的提取过程，其中双端读段均包含 4bp 的分子条形码，随后紧跟模板序列。

```
sentieon umi extract \
  -d \
  "4M+T,4M+T" \
  sample_R1.fastq.gz \
  sample_R2.fastq.gz | \
  gzip -c \
  > sample_extracted_pair.fastq.gz
```

紧接其后的代码示例则完整展示了从 UMI 提取到参考基因组比对的全过程。

```
sentieon umi extract \
  8M12S+T,+T \
  sample_R1.fastq.gz \
  sample_R2.fastq.gz | \
sentieon bwa mem \
  -R "@RG\tID:$GROUP\tSM:$SAMPLE\tLB:$LIBRARY\tPL:$PLATFORM" \
  -t $NT \
  -K $BWA_K_SIZE \
  -p \
  -C \
```

```
  $REF \
  - | \
sentieon util sort \
  -i - \
  -o sorted.bam \
  --sam2bam
```

此外，文档还提供了一个针对 UMI 序列已存储在独立 FASTQ 文件（如 sample_I1.fastq.gz）中的应用案例。在此模式下运行分析时，仅允许额外输入一个 UMI 索引读段，且该读段中不得包含任何模板序列。需要注意的是，此模式不支持双链 UMI 的提取。

```
sentieon umi extract \
"+M,+T,+T" \
sample_I1.fastq.gz \
sample_R1.fastq.gz \
sample_R2.fastq.gz | \
gzip -c \
> sample_extracted_pair.fastq.gz
```

下方的指令将生成 SAM 格式的输出结果，该格式在配合使用 RNA-seq 比对软件 STAR 时非常实用。在该示例中，配对读段的 R1 端包含 16bp 的细胞条形码和 10bp 的分子条形码，而 R2 端仅包含模板序列。其最终输出结果为 SAM 格式的单端读段。

```
sentieon umi extract \
  --output_format SAM \
  "16B10M+S,+T" \
  sample_R1.fastq.gz \
  sample_R2.fastq.gz \
  > sample_extracted.sam
```

umi extract 命令的输出包含额外的标签信息。在默认设置下，输出会包含一个用于记录细胞或样本条形码的 CR 标签，以及一个用于记录 UMI 序列的 XR 标签，后者将供后续的 umi consensus 步骤使用。

表 4-2 由 umi extract 生成的附加标签

标签	含义
RX	提取出的 UMI 序列碱基
XR	用于 umi consensus 步骤中进行分子分组的 UMI 标签
CR	细胞条形码

4.10 Joint Calling

4.10.1 Joint calling 分析脚本

以下是一个完整的 DNA 测序数据特别是针对多个样本的联合变异检测（joint calling）的分析脚本示例：

```
#!/bin/sh

# Copyright (c) 2016-2020 Sentieon Inc. All rights reserved

# ****************************************
# Script to perform joint calling on 3 samples
# with fastq files named sample<i>_1.fastq.gz
# and sample<i>_2.fastq.gz
# ****************************************

set -eu

# Update with the fullpath location of your sample fastq
SM_LIST="sample1 sample2 sample3" # list of sample names
PL="ILLUMINA" #or other sequencing platform
FASTQ_FOLDER="/home/pipeline/samples"
FASTQ_1_SUFFIX="1.fastq.gz"
FASTQ_2_SUFFIX="2.fastq.gz" #If using Illumina paired data

# Update with the location of the reference data files
FASTA_DIR="/home/regression/references/b37/"
FASTA="$FASTA_DIR/human_g1k_v37_decoy.fasta"
KNOWN_DBSNP="$FASTA_DIR/dbsnp_138.b37.vcf.gz"
KNOWN_INDELS="$FASTA_DIR/1000G_phase1.indels.b37.vcf.gz"
KNOWN_MILLS="$FASTA_DIR/Mills_and_1000G_gold_standard.indels.b37.vcf.gz"

# Update with the location of the Sentieon software package and license file
SENTIEON_INSTALL_DIR=/home/release/sentieon-genomics-|release_version|
export SENTIEON_LICENSE=/home/Licenses/Sentieon.lic #or using licsrvr: c1n11.sentieon.com:5443

# Other settings
NT=$(nproc) #number of threads to use in computation, set to number of cores in the server
START_DIR="$PWD/test/DNAseq_joint" #Determine where the output files will be stored
```

```
# You do not need to modify any of the lines below unless you want to tweak the pi
peline

# ********************************************************************************
********************************************************************************
**************

# ****************************************
# 0. Setup
# ****************************************
WORKDIR="$START_DIR/joint_call"
mkdir -p $WORKDIR
LOGFILE=$WORKDIR/run.log
exec >$LOGFILE 2>&1

# ****************************************
# 0. Process all samples independently
# ****************************************
GVCF_INPUTS=""
for SM in $SM_LIST; do
  RGID="rg_$SM"
  mkdir $WORKDIR/$SM
  cd $WORKDIR/$SM

  # ****************************************
  # 1. Mapping reads with BWA-MEM, sorting
  # ****************************************
  ( $SENTIEON_INSTALL_DIR/bin/sentieon bwa mem -R "@RG\tID:$RGID\tSM:$SM\tPL:$PL"
 \
       -t $NT -K 10000000 $FASTA $FASTQ_FOLDER/${SM}_$FASTQ_1_SUFFIX \
       $FASTQ_FOLDER/${SM}_$FASTQ_2_SUFFIX || { echo -n 'bwa error'; exit 1; } ) | \
       $SENTIEON_INSTALL_DIR/bin/sentieon util sort -r $FASTA -o sorted.bam -t $NT --sa
m2bam -i - || \
       { echo "Alignment failed"; exit 1; }

  # ****************************************
  # 2. Metrics
  # ****************************************
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i sorted.bam \
       --algo MeanQualityByCycle mq_metrics.txt --algo QualDistribution qd_metrics.txt \
```

```
        --algo GCBias --summary gc_summary.txt gc_metrics.txt --algo AlignmentStat \
        --adapter_seq '' aln_metrics.txt --algo InsertSizeMetricAlgo is_metrics.txt || \
        { echo "Metrics failed"; exit 1; }

  $SENTIEON_INSTALL_DIR/bin/sentieon plot GCBias -o gc-report.pdf gc_metrics.txt
  $SENTIEON_INSTALL_DIR/bin/sentieon plot QualDistribution -o qd-report.pdf qd_metrics.
txt
  $SENTIEON_INSTALL_DIR/bin/sentieon plot MeanQualityByCycle -o mq-report.pdf mq_me
trics.txt
  $SENTIEON_INSTALL_DIR/bin/sentieon plot InsertSizeMetricAlgo -o is-report.pdf is_metric
s.txt

  # ****************************************
  # 3. Remove Duplicate Reads. It is possible
  # to remove instead of mark duplicates
  # by adding the --rmdup option in Dedup
  # ****************************************
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo LocusCollector
\
        --fun score_info score.txt || { echo "LocusCollector failed"; exit 1; }

  $SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT -i sorted.bam --algo Dedup \
        --score_info score.txt --metrics dedup_metrics.txt deduped.bam || \
        { echo "Dedup failed"; exit 1; }

  # ****************************************
  # 2a. Coverage metrics
  # ****************************************
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam \
        --algo CoverageMetrics coverage_metrics || { echo "CoverageMetrics failed"; exit
1; }

  # ****************************************
  # 5. Base recalibration
  # ****************************************
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam --algo Q
ualCal \
        -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_data.table
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_
data.table \
        --algo QualCal -k $KNOWN_DBSNP -k $KNOWN_MILLS -k $KNOWN_INDELS recal_d
ata.table.post
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -t $NT --algo QualCal --plot \
        --before recal_data.table --after recal_data.table.post recal.csv
```

```
  $SENTIEON_INSTALL_DIR/bin/sentieon plot QualCal -o recal_plots.pdf recal.csv

  # ****************************************
  # 6b. HC Variant caller for GVCF
  # ****************************************
  $SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA -t $NT -i deduped.bam -q recal_
data.table \
       --algo Haplotyper -d $KNOWN_DBSNP --emit_conf=30 --call_conf=30 --emit_mode
gvcf \
       output-hc.g.vcf.gz || { echo "Haplotyper failed"; exit 1; }

  GVCF_INPUTS="$GVCF_INPUTS -v $WORKDIR/$SM/output-hc.g.vcf.gz"
done

# ****************************************
# Perform the joint calling
# ****************************************
$SENTIEON_INSTALL_DIR/bin/sentieon driver -r $FASTA --algo GVCFtyper $GVCF_INPUTS \
    -d $KNOWN_DBSNP $WORKDIR/output-jc.vcf.gz || { echo echo "GVCFtyper failed"; e
xit 1; }
```

5 工具详细使用说明

5.1 DRIVER 二进制程序

DRIVER 是用于执行生物信息学流程所有阶段的二进制程序。对 driver 可执行文件的单次调用可以运行多个算法；例如，指标计算阶段就是通过调用 driver 运行多个算法来实现的。

5.1.1 DRIVER 语法

DRIVER 二进制文件的一般语法是：

```
sentieon driver OPTIONS--algo ALGORITHM ALGO_OPTION OUTPUT \
 [--algo ALGORITHM2 ALGO_OPTION2 OUTPUT2]
```

在同一驱动程序调用中同时运行多个算法的情况下，所有的选项配置将在这些算法之间共享。

该命令的参数(OPTIONS)包括：

- **-t NUMBER_THREADS**：软件用于运行并行进程的计算线程数。此参数是可选的；如果省略，driver 工具将使用服务器拥有的所有线程。
- **-r REFERENCE**：参考基因组 FASTA 文件的位置。除了 LocusCollector 和不带 consensus 的 Dedup 外，此参数在所有算法中都是必需的。
- **-i INPUT_FILE**：BAM 输入文件的位置。可以多次使用此参数使软件使用多个输入文件，其结果将与将输入的 BAM 文件合并成一个 BAM 文件相同。Sentieon®仅支持按坐标排序的 bam 文件，必须存在带有 SO:coordinate 属性的@HD 行。Sentieon®还要求输入的 bam 文件在比对部分有 RG 标签，并在头部定义有相应匹配 ID 标签的@RG 行。除了 CollectVCMetrics、

GVCFtyper、DNAModelApply、SVSolver、TNModelApply、VarCal、ApplyVarCal 和 TNfilter 外，此参数在所有算法中都是必需的。

- **-q QUALITY_RECALIBRATION_TABLE**：来自 BQSR 阶段的质量重校准表的位置，该表将在 BQSR 阶段和变异检测阶段作为输入使用。可以多次使用此参数使软件使用多个输入校准表；每个校准表将通过匹配 reads 的读组对 BAM 文件进行重校准。
- **--interval INTERVAL**：软件将使用的参考基因组中的区间。可以多次使用此参数以在所有区间的并集上执行计算。INTERVAL 可以指定为：
  - **CONTIG[:START-END]**：仅在相应的 contig 上进行计算。START 和 END 是可选的数字，用于进一步缩小区间范围；两者都是以 1 为基的。您可以输入以逗号分隔的多个 contigs 列表。
  - **BED_FILE**：包含区间的 BED 文件的位置。提供空文件将导致不进行处理，就像区间长度为 0 一样。也支持压缩的 BGZF 文件。
  - **PICARD_INTERVAL_FILE**：包含区间的文件位置，遵循 Picard 区间标准。提供空文件将导致不进行处理，就像区间长度为 0 一样。也支持压缩的 BGZF 文件。
  - **VCF_FILE**：包含变异记录的 VCF 文件位置，其基因组坐标将用作区间。也支持压缩的 VCF.gz 文件。
- **--interval_padding PADDING_SIZE**：为输入区间的边缘添加 PADDING_SIZE 个碱基的填充。默认值为 0。
- **--help**：显示帮助的选项。该选项可以与--algo ALGORITHM 一起使用，以显示特定算法的帮助。

- **--read_filter FILTER,OPTION=VALUE,OPTION=VALUE**：在应用算法之前执行 reads 的过滤或转换。请参考 *第 5.1.3 节* 获取有关可用过滤器及其功能的其他信息。
- **--temp_dir DIRECTORY**：确定临时文件的存储位置。默认为命令运行的文件夹($PWD)。
- **--skip_no_coor**：确定是否跳过未映射的 reads。
- **--cram_read_options decode_md=0**：CRAM 输入选项，用于关闭输入 CRAM 中的 NM/MD 标签。
- **--replace_rg ORIG_RG="NEW_RG_STRING"**：修改下一个 BAM 输入文件的@RG 读组标签，使用新信息更新 ORIG_RG；NEW_RG_STRING 需要是有效且完整的字符串。此参数可以多次使用，每次都会影响下一个-i BAM 输入。您可以查看 *第 6.6 节* 获取详细的使用示例。

该命令支持的算法包括：

- **LocusCollector、Dedup**：用于去除重复序列阶段。
- **Realigner**：用于插入缺失重比对阶段。
- **QualCal**：用于碱基质量重校准阶段。
- **MeanQualityByCycle、QualDistribution、GCBias、AlignmentStat、InsertSizeMetricAlgo、HsMetricAlgo、CoverageMetrics、BaseDistributionByCycle、QualityYield、WgsMetricsAlgo、SequenceArtifactMetricsAlgo**：用于计算 QC 指标。
- **Genotyper、Haplotyper**：用于胚系变异检测分析。
- **DNAscope、DNAModelApply**：用于 DNAscope 胚系变异检测分析。

- **DNAscope、SVsolver**：用于 DNAscope 胚系结构变异分析。
- **DNAscope、VariantPhaser**：用于 PacBio® HiFi® reads 的 DNAscope 胚系变异检测分析。
- **ReadWriter**：用于输出经过碱基质量重校准后的 BAM 文件或合并多个 BAM 文件。
- **VarCal、ApplyVarCal**：用于 VQSR 阶段。
- **GVCFtyper**：用于胚系联合变异检测分析。
- **TNsnv、TNhaplotyper、TNhaplotyper2、ContaminationModel、OrientationBias、TNfilter**：用于体细胞瘤瘤-正常或仅瘤样本分析。
- **TNscope、TNModelApply**：用于体细胞瘤瘤-正常体细胞和结构变异分析。
- **RNASplitReadsAtJunction**：用于在结点处分割 RNA reads 的阶段。
- **ContaminationAssessment**：用于从 BAM 文件中识别样本间污染。
- **CollectVCMetrics**：用于计算变异检测后的指标。

5.1.2 DRIVER 算法

（1）LocusCollector

LocusCollector 算法收集将用于删除重复读段的读段信息。

LocusCollector 算法的输入是 BAM 文件；其输出是指示哪些读段可能是重复的评分文件。

LocusCollector 算法需要以下 ALGO_OPTION：

- **--fun SCORE**：要使用的评分函数。SCORE 的可能值为：
  - **score_info**：计算 Dedup 算法的读段对评分。这是默认的评分函数。

LocusCollector 算法接受以下可选的 ALGO_OPTION：

- **--consensus**：设置此选项以启用共识去重。
- **--umi_tag TAG**：用于 UMI 条形码感知去重的逻辑 UMI 标签。默认值为 None。
- **--umi_ecc_dist DISTANCE**：UMI 条形码错误纠正距离。将其设置为 0 以关闭条形码错误纠正。默认值为 1。
- **--rna**：为使用 STAR 对齐的 RNA 序列数据设置此选项。

(2) Dedup

Dedup 算法执行重复读段的标记/删除。

Dedup 算法的输入是 BAM 文件；其输出是删除重复读段后的 BAM 文件。

Dedup 算法需要以下 ALGO_OPTION：

- **--score_info LOCUS_COLLECTOR_OUTPUT**：LocusCollector 命令检测的输出文件位置。

Dedup 算法接受以下可选的 ALGO_OPTION：

- **--rmdup**：设置此选项以在输出 BAM 文件中删除重复的读段。如果未设置此选项，重复的读段将被标记，但不会从 BAM 文件中删除。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]**：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。
- **--metrics METRICS_FILE**：包含去重阶段指标数据的输出文件的位置和文件名。
- **--optical_dup_pix_dist DISTANCE**：确定两个重复读段被视为光学重复的最大距离。默认值为 100。
- **--bam_compression COMPRESSION_LEVEL[0-9]**：输出 BAM 文件的 gzip 压缩级别。默认值为 6。
- **--output_dup_read_name**：使用此选项时，命令的输出将不是带有标记/删除重复的 BAM 文件，而是被标记为重复的读段名称列表。
- **--dup_read_name DUPLICATE_READ_NAME_FILE**：使用此输入时，DUPLICATE_READ_NAME_FILE 中包含的所有读段将被标记为重复，无论它们是

主要还是非主要读段。

（3）Realigner

Realigner 算法执行插入缺失（indel）重新比对。

Realigner 算法的输入是 BAM 文件；其输出是重新比对后的 BAM 文件。

Realigner 算法接受以下可选的 ALGO_OPTION：

- **-k KNOWN_SITES**：用作已知位点集的 VCF 文件位置。已知位点将用于帮助识别可能需要重新比对的位点；只会使用文件中的 indel 变异。您可以通过指定多个文件并重复 -k KNOWN_SITES 选项来包含多个已知位点集合。
- **--interval_list INTERVAL**：在计算重新比对目标时将使用的参考序列中的区间。只会考虑单个输入 INTERVAL：如果您重复 --interval_list 选项，只会考虑最后一个 INTERVAL。INTERVAL 可以指定为：
  - **BED_FILE**： 包含区间的 BED 文件的位置。
  - **PICARD_INTERVAL_FILE**： 包含区间的文件的位置，遵循 Picard 区间标准。
- **--bam_compression COMPRESSION_LEVEL[0-9]**：输出 BAM 文件的 gzip 压缩级别。默认值为 6。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]**：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。

(4) QualCal

QualCal 算法计算执行 BQSR（碱基质量分数重校准）所需的重校准表。QualCal 算法还应用重校准来计算创建报告所需的数据，并创建生成报告所需的数据。

重校准的计算依赖于 ReadGroup 的平台（PL）标签；QualCal 算法支持以下平台：ILLUMINA、ION_TORRENT、LS454、PACBIO、COMPLETE_GENOMICS、DNBSEQ。目前尚未实现对 SOLID 平台测序数据的支持。

QualCal 算法的输入是 BAM 文件；其输出是重校准表或包含创建报告所需数据的 csv 文件。

QualCal 算法接受以下可选的 ALGO_OPTION：

- **-k KNOWN_SITES**：用作已知位点集的 VCF 文件位置。已知位点将用于确保已知位置不会因误将真实变异识别为测序方法错误而获得人为低质量分数。您可以通过指定多个文件并重复该选项来包含多个已知位点集合。我们强烈建议使用尽可能多的已知位点，否则重校准将把变异位点视为测序错误。
- **--plot**：指示该命令是否用于生成创建报告所需的数据。
- **--cycle_val_max**：循环协变量允许的最大循环值。
- **--before RECAL_TABLE**：先前计算的重校准表的位置；它将用于应用重校准。
- **--after RECAL_TABLE.POST**：先前计算的应用重校准表结果的位置；它将用于计算创建报告所需的数据。

(5) MeanQualityByCycle

MeanQualityByCycle 算法用于计算每个测序循环的平均碱基质量分数。

MeanQualityByCycle 算法的输入是一个 BAM 文件；其输出是指标数据。

MeanQualityByCycle 算法不接受任何 ALGO_OPTION。

(6) QualDistribution

QualDistribution 算法用于计算具有特定碱基质量分数的碱基数量。

QualDistribution 算法的输入是一个 BAM 文件；其输出是指标数据。

QualDistribution 算法不接受任何 ALGO_OPTION。

(7) GCBias

GCBias 算法用于计算参考基因组和样本中的 GC 偏差。

GCBias 算法的输入是一个 BAM 文件；其输出是指标数据。

GCBias 算法接受以下可选的 ALGO_OPTION：

- **--summary SUMMARY_FILE**：输出文件的位置和文件名，用于汇总 GC 偏差指标。
- **--accum_level LEVEL**：确定累积级别。LEVEL 的可能值包括 ALL_READS、SAMPLE、LIBRARY、READ_GROUP。默认值是 ALL_READS。
- **--also_ignore_duplicates**：决定是否使用唯一的非重复读段来计算输出指标。
- **--is_bisulfite_seq**：设置此选项表明输入的 BAM 文件包含亚硫酸氢盐测序的 reads。

(8) HsMetricAlgo

HsMetricAlgo 算法用于计算样本的杂交选择特定指标以及参考基因组的 AT/GC 脱落指标。

HsMetricAlgo 算法的输入是一个 BAM 文件；其输出是指标数据。

HsMetricAlgo 算法需要以下 ALGO_OPTION：

- **--targets_list TARGETS_FILE**：包含目标位置的间隔列表输入文件的位置和文件名。
- **--baits_list TARGETS_FILE**：包含所使用诱饵位置的间隔列表输入文件的位置和文件名。
- **--clip_overlapping_reads**：确定在计算过程中是否剪切重叠的读段。
- **--min_map_qual QUALITY**：确定所使用读段的过滤质量。任何映射质量低于 QUALITY 的读段将被过滤掉。
- **--min_base_qual QUALITY**：确定变异检测中使用的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。
- **--coverage_cap COVERAGE**：确定直方图中使用的最大覆盖度限制。

(9) AlignmentStat

AlignmentStat 算法用于计算读段比对的统计数据。

AlignmentStat 算法的输入是一个 BAM 文件；其输出是度量数据。

AlignmentStat 算法接受以下可选的 ALGO_OPTION：

- **--adapter_seq SEQUENCE_LIST**：测序中使用的接头序列，以逗号分隔的列表形式提供。默认值是默认的 Illumina 接头列表。
- **--is_bisulfite_seq**：设置此选项表明输入的 BAM 文件包含亚硫酸氢盐测序的 reads。

（10）InsertSizeMetricAlgo

InsertSizeMetricAlgo 算法用于计算插入片段大小的统计分布。

InsertSizeMetricAlgo 算法的输入是一个 BAM 文件；其输出是指标数据。

InsertSizeMetricAlgo 算法接受以下可选的 ALGO_OPTION：

- **--deviation DEVIATION**：在直方图被裁剪之前的最大标准偏差倍数。默认值为 10.0。
- **--hist_width HISTOGRAM_WIDTH**：设置 HISTOGRAM_WIDTH 以覆盖直方图尾部的自动截断。默认值为 0。
- **--min_read_ratio RATIO**：一个读段类别被包含在直方图中的最小读段比率。默认值为 0.05。
- **--se_tag TAG**：在直方图中包含带有指定标签（TAG）的单端读段。默认值为 MI。

（11）CoverageMetrics

CoverageMetrics 算法用于计算 BAM 文件的深度覆盖率。如果在驱动程序级别包含了该选项，则覆盖率会按区间聚合；如果未包含选项，则会对参考基因组中所有碱基按每个重叠群进行区间聚合，每个位点的覆盖率输出文件将约为 60GB。如果包含 RefSeq 文件，则覆盖率会按基因聚合。

CoverageMetrics 算法的输入是 BAM 文件，建议使用去重复后的 BAM 文件进行此分析；其输出是包含按分区、聚合和输出组织的指标数据的文件。可能的输出包括：

- **summary**：包含深度数据。
- **statistics**：包含特定深度位点的直方图。
- **cumulative_coverage_counts**：包含深度大于 x 的位点直方图。
- **cumulative_coverage_proportions**：包含深度大于 x 的位点的归一化直方图。

当输出名称为 OUTPUT 时，输出文件示例：

- **OUTPUT**：无分区的每个位点覆盖率。
- **OUTPUT.sample_summary**：PARTITION_GROUP 样本的汇总，聚合所有碱基。
- **OUTPUT.library_interval_statistics**：PARTITION_GROUP 文库的统计数据，按区间聚合。

CoverageMetrics 算法接受以下可选的 ALGO_OPTION：

- **--partition PARTITION_GROUP**：确定如何分区数据。可能的值是 readgroup 或 RG 属性的逗号分隔组合，即 sample、platform、library、

center。默认值是 "sample"。可以通过重复该选项包含多个分区组，每个输出文件将为每个选项创建一次。

- **--gene_list REFSEQ_FILE**：用于将 CoverageMetrics 算法的结果聚合到基因级别的 RefSeq 文件位置。
- 过滤选项：
  - **--min_map_qual MAP_QUALITY**：确定使用的读段质量过滤。任何映射质量低于 QUALITY 的读段将被过滤掉。
  - **--max_map_qual MAP_QUALITY**：确定使用的读段质量过滤。任何映射质量高于 QUALITY 的读段将被过滤掉。
  - **--min_base_qual QUALITY**：确定使用的碱基质量过滤。任何质量低于 QUALITY 的碱基将被忽略。
  - **--max_base_qual QUALITY**：确定使用的碱基质量过滤。任何质量高于 QUALITY 的碱基将被忽略。
  - **--cov_thresh THRESHOLD**：在聚合中添加覆盖率大于阈值的碱基百分比。可以通过重复参数包含多个阈值。
- 省略输出选项：
  - **--omit_base_output**：跳过无分区的每个位点覆盖率输出。当不使用区间以节省空间时，可以使用此选项。
  - **--omit_sample_stat**：跳过聚合所有碱基的汇总结果输出（_summary）。
  - **--omit_locus_stat**：跳过所有直方图文件的输出（_cumulative_coverage_counts 和 _cumulative_coverage_proportions）。

- **--omit_interval_stat**：跳过所有区间统计文件的输出（_interval_statistics）。

- **--count_type TYPE**：确定如何处理来自同一片段的重叠读段。可能的选项有：
  - 0：计算重叠读段，即使它们来自同一片段。这是默认值。
  - 1：计算重叠读段。
  - 2：仅当片段中的读段具有一致的碱基时才计算重叠读段。
- **--print_base_counts**：在无分区的每个位点覆盖率输出中包含 "AGCTND" 的数量。
- **--include_ref_N**：包括参考基因组设置为 N 的位点的覆盖率数据。
- **--ignore_del_sites**：忽略有删除的位点的覆盖率数据。
- **--include_del**：此参数将与其他参数交互如下：
  - 如果 ignore_del_sites 关闭，将删除计为深度
  - 如果 print_base_counts 打开，包括 'D' 的数量
- **--histogram_scale [log/linear] --histogram_low MIN_DEPTH --histogram_high MAX_DEPTH --histogram_bin_count NUM_BINS**：确定直方图的比例类型、bin 和大小。默认值为 log、1、500、499。

（12）CollectVCMetrics

CollectVCMetrics 算法收集与输入 VCF 文件中存在的变异相关的度量指标。

CollectVCMetrics 算法的输入是一个 VCF 文件和一个 DBSNP 文件；其输出是一对包含 VCF 文件中变异信息的文件。

CollectVCMetrics 算法需要以下 ALGO_OPTION：

- **-d dbSNP_FILE**：单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **-v INPUT**：将计算度量指标的 VCF 文件的位置。仅支持一个文件。

（13）BaseDistributionByCycle

BaseDistributionByCycle（按循环的碱基分布）

BaseDistributionByCycle 算法计算每个测序循环的核苷酸分布。

BaseDistributionByCycle 算法的输入是一个 BAM 文件；其输出是度量数据。

BaseDistributionByCycle 算法接受以下可选的 ALGO_OPTION：

- **--aligned_reads_only**：决定是否仅计算已比对读段的碱基分布。
- **--pf_reads_only**：决定是否仅计算通过过滤（PF）读段的碱基分布。

(14) QualityYield

QualityYield 算法收集与通过质量阈值和 Illumina 特定过滤器的读序相关的指标。

QualityYield 算法的输入是一个 BAM 文件；其输出是指标数据。

QualityYield 算法接受以下可选的 ALGO_OPTION：

- **--include_secondary**：决定是否在计算中包括来自次级比对的碱基。
- **--include_supplementary**：决定是否在计算中包括来自补充比对的碱基。

（15）WgsMetricsAlgo

WgsMetricsAlgo 算法收集与全基因组测序（WGS）实验的覆盖度和性能相关的指标。

WgsMetricsAlgo 算法的输入是一个 BAM 文件；其输出是指标数据。

WgsMetricsAlgo 算法接受以下可选的 ALGO_OPTION：

- **--min_map_qual QUALITY**：决定用于计算的读序的过滤质量。任何质量低于 QUALITY 的读序将被忽略。默认值为 20。
- **--min_base_qual QUALITY**：决定用于计算的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 20。
- **--coverage_cap COVERAGE**：决定直方图的最大覆盖度限制。任何覆盖度高于 COVERAGE 的位置，其覆盖度将被设置为 COVERAGE。
- **--sample_size SIZE**：决定用于理论杂合敏感性采样的样本大小。默认值为 10000。
- **--include_unpaired**：决定是否计算未配对的读序和一端未映射的配对读序。
- **--base_qual_histogram**：决定是否报告碱基质量直方图。

(16) SequenceArtifactMetricsAlgo

SequenceArtifactMetricsAlgo 算法收集量化单碱基测序伪影和 OxoG 伪影的指标。

SequenceArtifactMetricsAlgo 算法的输入是一个 BAM 文件；其输出是指标数据。

SequenceArtifactMetricsAlgo 算法接受以下可选的 ALGO_OPTION：

- **--dbsnp FILE**：用于排除已知多态性周围区域的单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **--min_map_qual QUALITY**：决定用于计算的读序的过滤质量。任何质量低于 QUALITY 的读序将被忽略。默认值为 30。
- **--min_base_qual QUALITY**：决定用于计算的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 20。
- **--include_unpaired**：决定是否计算未配对的读序和一端未映射的配对读序。
- **--include_duplicates**：决定是否计算重复的读序。
- **--include_non_pf_reads**：决定是否计算非 PF 读序。
- **--min_insert_size ISIZE**：决定用于计算的读序的过滤插入大小。任何插入大小小于 ISIZE 的读序将被忽略。默认值为 60。
- **--max_insert_size ISIZE**：决定用于计算的读序的过滤插入大小。任何插入大小大于 ISIZE 的读序将被忽略。默认值为 600。
- **--tandem_reads**：决定配对读序是否从同一条链上测序。
- **--context_size SIZE**：决定在每侧包含的上下文碱基数量。默认值为 1。

（17）Genotyper

Genotyper 算法执行统一基因型变异检测。

Genotyper 算法的输入是 BAM 文件；输出是 VCF 文件。

Genotyper 算法接受以下可选的 ALGO_OPTION：

- **--annotation 'ANNOTATION_LIST'**：确定将添加到输出 VCF 的其他注释。使用逗号分隔的列表来启用或禁用注释。包括'none'以删除默认注释；在注释前加感叹号（!）以禁用特定注释。有关支持的注释，请参见 *第 6.2 节*。
- **-d dbSNP_FILE**：用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **--var_type VARIANT_TYPE**：确定将检测哪些变异类型；VARIANT_TYPE 的可能值为：
  - SNP 仅检测单核苷酸多态性。这是默认行为。
  - INDEL 仅检测插入-缺失。
  - both 同时检测 SNP 和 INDEL。
- **--call_conf CONFIDENCE**：确定检测变异的变异质量阈值。质量低于 CONFIDENCE 的变异将被删除。
- **--emit_conf CONFIDENCE**：确定发射变异的变异质量阈值。质量低于 CONFIDENCE 的变异将不会添加到输出 VCF 文件中。
- **--emit_mode MODE**：确定将发射哪些检测。MODE 的可能值为：
  - variant：仅在置信的变异位点发射检测。这是默认行为。
  - confident：在置信的变异位点或置信的参考位点发射检测。
  - all：发射所有检测，无论其置信度如何。

- **--min_base_qual QUALITY**：确定用于变异检测的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 17。
- **--ploidy PLOIDY**：确定正在处理的样本的倍性数。默认值为 2。
- **--given GIVEN_VCF**：仅使用 GIVEN_VCF 中提供的变异执行变异检测。检测将仅评估文件中提供的位点和等位基因，并且仅当变异的 FILTER 列为 PASS 或.时。
- **--genotype_model MODEL**：确定用于基因型分型和 QUAL 计算的模型。MODEL 可以是 coalescent 使用基于群体遗传学中的合并理论的"所谓精确模型"，或 multinomial 使用假设变异是独立的简化模型；默认值是 coalescent。

（18）Haplotyper

Haplotyper 算法执行单倍型变异检测。

Haplotyper 算法的输入是 BAM 文件；输出是 VCF 文件。

Haplotyper 算法接受以下可选的 ALGO_OPTION：

- **--annotation 'ANNOTATION_LIST'**：确定将添加到输出 VCF 的其他注释。使用逗号分隔的列表来启用或禁用注释。包括'none'以删除默认注释；在注释前加感叹号（!）以禁用特定注释。有关支持的注释，请参见 *第 6.2 节*。
- **-d dbSNP_FILE**：用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **--call_conf CONFIDENCE**：确定检测变异的变异质量阈值。质量低于 CONFIDENCE 的变异将被删除。当--emit_mode 为 gvcf 时，此选项将被忽略。
- **--emit_conf CONFIDENCE**：确定发射变异的变异质量阈值。质量低于 CONFIDENCE 的变异将不会添加到输出 VCF 文件中。当--emit_mode 为 gvcf 时，此选项将被忽略。
- **--emit_mode MODE**：确定将发射哪些检测。MODE 的可能值为：
  - variant：仅在置信的变异位点发射检测。这是默认行为。
  - confident：在置信的变异位点或置信的参考位点发射检测。
  - all：发射所有检测，无论其置信度如何。
  - gvcf：发射联合检测所需的附加信息。如果要使用 GVCFtyper 算法执行联合检测，则需要此选项。
- **--gq_bands LIST_OF_BANDS**：确定用于压缩具有相似基因型质量（GQ）

的变异的带，这些变异将在 GVCF 输出文件中作为单个 VCF 记录发射。LIST_OF_BANDS 是一个逗号分隔的带列表，其中每个带由 START-END/STEP 定义。默认值为 1-60，60-99/10，99。

- **--min_base_qual QUALITY**：确定用于变异检测的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 10。
- **--pcr_indel_model MODEL**：用于或多或少积极地剔除假阳性插入缺失的 PCR 插入缺失模型。可能的 MODEL 有：NONE（用于无 PCR 样本），以及 HOSTILE、AGGRESSIVE 和 CONSERVATIVE，按降低积极性的顺序排列。默认值为 CONSERVATIVE。
- **--phasing [1/0]**：在使用 emit_mode GVCF 时启用或禁用输出中的相位的标志。默认值为 1（开启），此标志在使用 GVCF 以外的 emit_mode 时无影响。相位仅为二倍体样本计算。
- **--ploidy PLOIDY**：确定正在处理的样本的倍性数。默认值为 2。
- **--prune_factor FACTOR**：局部组装中的最小修剪因子；支持 k-mer 少于 FACTOR 的路径将从图中修剪。默认值为 2。
- **--trim_soft_clip**：确定是否应从变异检测中排除读段中的软剪切碱基。仅建议处理 RNA 读段时使用此参数。
- **--given GIVEN_VCF**：仅使用 GIVEN_VCF 中提供的变异执行变异检测。检测将仅评估文件中提供的位点和等位基因，并且仅当变异的 FILTER 列为 PASS 或.时。此选项不能与--emit_mode gvcf 一起使用。
- **--bam_output OUTPUT_BAM**：输出包含变异检测所做的局部重组后修改读段的 BAM 文件。此选项仅应与小型 bed 文件一起用于故障排除目的。

- **--genotype_model MODEL**：确定用于基因型分型和 QUAL 计算的模型。MODEL 可以是 coalescent 使用基于群体遗传学中的合并理论的"所谓精确模型"，或 multinomial 使用假设变异是独立的简化模型；默认值是 coalescent。

（19）ReadWrite

ReadWriter 算法将碱基质量分数重校准（BQSR）的应用结果输出到文件。

ReadWriter 算法还可以合并 BAM 文件，和/或将它们转换为 CRAM 文件。

ReadWriter 算法的输入是一个或多个 BAM 文件和一个或多个重校准表；其输出是重校准后的 BAM 文件。如果输出文件扩展名为 CRAM，将创建 CRAM 文件。如果使用了多个输入文件，输出文件将是合并所有文件的结果。

ReadWriter 算法接受以下可选的 ALGO_OPTION：

- **--bam_compression COMPRESSION_LEVEL[0-9]**：输出 BAM 文件的 gzip 压缩级别。默认值为 6。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]**：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。
- **--output_mapq_filter min_map_qual=MIN_QUAL, max_map_qual=MAX_QUAL**：基于 reads 的映射质量进行过滤；ReadWriter 只会输出映射质量大于等于 MIN_QUAL 且小于等于 MAX_QUAL 的 reads。
- **--output_flag_filter MASK：VALUE[，VALUE，VALUE...]**：基于标志过滤 reads。MASK 是应考虑的位集，VALUE 是以逗号分隔的 FLAG 列表，表示保留 read 所需设置的标志，如果 MASK 中的所有位都需要未设置，则为 0。您可以包含此选项的多个实例，它们将按命令行中的顺序依次应用。一些示例：
  - 要只输出 PROPER_PAIR 的 reads，您可以使用 --output_flag_filter PROPER_PAIR：PROPER_PAIR 或 --output_flag_filter 0x2：0x2

- 要输出除了设置了 UNMAP 标志的未映射 reads 之外的所有 reads，您可以使用 --output_flag_filter UNMAP：0 或 --output_flag_filter 0x4：0
- 要写入一个既不包含重复 reads 也不包含未映射 reads 的 BAM 文件，您可以使用两个选项，它们将依次应用：首先只允许 UNMAP 标志未设置的 reads，然后只允许 DUP 标志未设置的 reads。--output_flag_filter UNMAP：0 --output_flag_filter DUP：0
- 要写入一个不包含两个配对 reads 都未映射的 BAM 文件，您可以使用 --output_flag_filter UNMAP+MUNMAP：0，UNMAP，MUNMAP，这将允许 reads，只要 UNMAP 和 MUNMAP 不同时设置。
- 更复杂的情况可以借助真值表构建，其中 MASK 是您想考虑的所有位的总和，VALUE 是每种不同条件下设置的那些位的总和。例如，要查找两个非二次和非补充 reads 具有相同方向（F1F2 或 R1R2）的 read 对（暗示结构变异），您可以创建如下表格：

Flag	#	F1F2	R1R2
PAIRED	0x001	x	x
REVERSE	0x010	0	x
MREVERSE	0x020	0	x
SECONDARY	0x100	0	0
SUPPLEMENTARY	0x800	0	0
	0x931	0x001	0x031

选项将是 **--output_flag_filter 0x931：0x001，0x031**

或更复杂的形式 **--output_flag_filter PAIRED+SECONDARY+**

SUPPLEMENTARY+DUP+REVERSE+MREVERSE：PAIRED+REVERSE+MREVERSE，PAIRED+REVERSE+MREVERSE

下表是标志定义及其命名约定的提醒：

Flag#	Flag name	Description	Decimal
0x1	PAIRED	该读段是双端测序的一部分	1
0x2	PROPER_PAIR	根据比对器，读段以正确的方向和距离成对比对	2
0x4	UNMAP	片段未映射	4
0x8	MUNMAP	模板中的下一个片段未映射	8
0x10	REVERSE	读段比对到反向链	16
0x20	MREVERSE	模板中下一个片段的 SEQ 是反向的	32
0x40	READ1	双端测序中的 Read 1	64
0x80	READ2	双端测序中的 Read 2	128
0x100	SECONDARY	二次比对	256
0x200	QCFAIL	未通过质量控制	512
0x400	DUP	PCR 或光学重复	1024
0x800	SUPPLEMENTARY	补充比对	2048

我们建议在执行变异检测命令的同时运行 ReadWriter 算法，以减少开销。

（20）GVCFtyper

GVCFtyper 算法执行多个样本的联合变异检测，前提是每个单个样本之前已使用 Haplotyper 算法和--emit_mode gvcf 选项进行处理。

GVCFtyper 算法在驱动程序级别除参考文件外没有输入，其输出是包含所有样本联合检测变异的 VCF 文件。

GVCFtyper 算法需要以下 ALGO_OPTION：

- **-v INPUT**：使用额外选项--emit_mode gvcf 对单个样本执行变异检测算法所得的 GVCF 文件的位置。您可以通过指定多个文件并重复该选项来包含来自多个样本的 GVCF 文件。您可以使用用 bgzip 压缩和索引的 VCF 文件。
- 或者，您可以在命令末尾的输出文件之后输入 GVCF 文件列表。因此，以下两个命令是可互换的：

```
sentieon driver -r REFERENCE --algo GVCFtyper \
  -v s1_VARIANT_GVCF -v s2_VARIANT_GVCF -v s3_VARIANT_GVCF VARIANT_VCF
sentieon driver -r REFERENCE --algo GVCFtyper \
  VARIANT_VCF s1_VARIANT_GVCF s2_VARIANT_GVCF s3_VARIANT_GVCF
```

- 您可以使用以下命令从文本文件读取 GVCF 文件列表：

```
gvcf_argument=""
while read -r line; do
 gvcf_argument=$gvcf_argument" -v $line"
done < "list_of_gvcfs"
sentieon driver -r REFERENCE --algo GVCFtyper $gvcf_argument output-joint.vcf
```

- 您还可以使用以下利用 stdin 管道（-）的命令从文本文件读取 GVCF 文件列表：

```
cat list_of_gvcfs | sentieon driver -r REFERENCE --algo GVCFtyper output-joint.vcf -
```

- 您可以使用以下命令输入特定文件夹中的所有文件：

```
sentieon driver -r REFERENCE --algo GVCFtyper output-joint.vcf sample*.g.vcf
```

GVCFtyper 算法接受以下可选的 ALGO_OPTION：

- **--annotation 'ANNOTATION_LIST'**：确定将添加到输出 VCF 的其他注释。使用逗号分隔的列表来启用或禁用注释。包括'none'以删除默认注释；在注释前加感叹号（!）以禁用特定注释。有关支持的注释，请参见 *第 6.2 节*。
- **-d dbSNP_FILE**：用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **--call_conf CONFIDENCE**：确定检测变异的变异质量阈值。质量低于 CONFIDENCE 的变异将被删除。默认值为 30。
- **--emit_conf CONFIDENCE**：确定发射变异的变异质量阈值。质量低于 CONFIDENCE 的变异将不会添加到输出 VCF 文件中。默认值为 30。
- **--emit_mode MODE**：确定将发射哪些检测。MODE 的可能值为：
  - variant：仅在置信的变异位点发射检测。这是默认行为。
  - confident：在置信的变异位点或置信的参考位点发射检测。
  - all：发射所有检测，无论其置信度如何。
- **--max_alt_alleles NUMBER** ：最大替代等位基因数。默认值为 100。
- **--genotype_model MODEL**：确定用于基因型分型和 QUAL 计算的模型。MODEL 可以是 coalescent 使用基于群体遗传学中的合并理论的"所谓精确模型"，或 multinomial 使用假设变异是独立的简化模型；默认值是 coalescent。

(21) VarCal

VarCal 算法计算变异质量分数重校准（VQSR）。VQSR 为单个变异检测分配一个经过良好校准的概率分数，以便在确定最可能的变异时能够更准确地控制。为此，VQSR 使用高度可信的已知位点来构建重校准模型，并确定检测位点为真的概率。有关该算法的更多信息，您可以查看 *https://gatk.broadinstitute.org/hc/en-us/community/topics*。有关 VQSR 中推荐使用的资源信息，您可以查看 *https://gatk.broadinstitute.org/hc/en-us*。

VarCal 算法在驱动程序级别除参考文件外没有输入，其输出是包含与 VQSR 相关的额外注释的重校准文件。

VarCal 算法需要以下 ALGO_OPTION：

- **-v INPUT**：变异检测算法生成的 VCF 文件的位置；您可以使用用 bgzip 压缩和索引的 VCF 文件。
- **--tranches_file TRANCHES_FILE**：包含检测集划分为质量等级的文件的位置和文件名。
- **--annotation ANNOTATION**：确定在重校准期间将使用的注释。您可以通过重复该选项在优化中包含多个注释。您可以使用原始变异检测文件中存在的所有注释。
- **--resource RESOURCE_FILE --resource_param PARAM**：用作 VQSR 中训练/真实资源的 VCF 文件的位置，后跟确定如何使用该文件的参数。您可以通过指定多个文件并重复该选项来包含多个集合。PARAM 参数遵循以下语法：

```
LABEL,known=IS_KNOWN,training=IS_TRAIN,truth=IS_TRUTH,prior=PRIOR
```

- LABEL 是资源的描述性名称。
- IS_KNOWN 可以是 true 或 false，确定资源中包含的位点是否将用于分层输出指标。
- IS_TRAIN 可以是 true 或 false，确定资源中包含的位点是否将用于训练重校准模型。
- IS_TRUTH 可以是 true 或 false，确定资源中包含的位点是否将被视为真实位点。
- PRIOR 是一个值，反映了您对该资源作为真实集的可靠性的信心。

VarCal 算法接受以下可选的 ALGO_OPTION：

- **--srand RANDOM_SEED**：确定用于随机数生成的种子。您可以将 RANDOM_SEED 设置为 0，软件将使用您计算机的随机种子。为了生成确定性结果，您应使用非零 RANDOM_SEED 并将 NUMBER_THREADS 设置为 1。
- **--var_type VARIANT_TYPE**：确定将重校准哪些变异类型；VARIANT_TYPE 的可能值为：
  - SNP：仅重校准单核苷酸多态性。这是默认行为。
  - INDEL：仅重校准插入-缺失。
  - BOTH：重校准 SNP 和 INDEL。不应使用此设置，因为应该独立对 SNP 和 INDEL 执行 VQSR。
- **--tranche TRANCH_THRESHOLD**：每个等级的归一化质量阈值；TRANCH_THRESHOLD 数字是 0 到 100 之间的数字。允许多个选项实例，这将创建与阈值一样多的等级。默认值为 90、99、99.9 和 100。

- **--max_gaussians MAX_GAUSS**：确定将用于正重校准模型的高斯分布的最大数量。SNP 的默认值为 8，INDEL 的默认值为 4。
- **--max_neg_gaussians MAX_GAUSS**：确定将用于负重校准模型的高斯分布的最大数量。默认值为 2。
- **--max_iter MAX_ITERATIONS**：确定期望最大化（EM）优化的最大迭代次数。默认值为 150。
- **--max_mq MAPQ**： 指示数据中的最大 MQ，将用于执行 MQ 的 logit 抖动变换，使分布更接近高斯分布。
- **--aggregate_data AGREGATE_VCF**：包含从其他类似样本检测的变异的额外 VCF 文件的位置；这些额外数据将增加统计模型校准的有效样本量。允许多个选项实例。
- **--plot_file PLOT_FILE**：包含生成 VarCal 算法报告所需数据的临时文件的位置。

（22）ApplyVarCal

ApplyVarCal 算法将 VQSR 的输出信息与原始变异信息结合。

ApplyVarCal 算法在驱动程序级别除参考文件外没有输入；其输出是原始 VCF 的副本，包含来自 VQSR 的额外注释。

ApplyVarCal 算法需要以下 ALGO_OPTION：

- **-v INPUT**：变异检测算法生成的 VCF 文件的位置。它应该与 VarCal 算法中使用的文件相同；您可以使用用 bgzip 压缩和索引的 VCF 文件。
- **--recal VARIANT_RECAL_DATA**：VarCal 算法输出的 VCF 文件的位置。
- **--tranches_file TRANCHES_FILE**：VarCal 算法输出的 tranches 文件的位置。
- **--var_type VARIANT_TYPE**：确定将重校准哪些变异类型。此选项应与 VarCal 算法中使用的选项一致。

或者，您可以使用 --vqsr_model 选项，以逗号分隔列表的形式输入多个 VQSR 模型所需的信息。该选项允许您在单条命令行中同时应用 SNP 和 INDEL 检测模式。该选项的语法如下：

```
--vqsr_model  var_type=VARIANT_TYPE,\
                    recal=VARIANT_RECAL_DATA,\
                    tranches_file=TRANCHES_FILE,\
                    sensitivity=SENSITIVITY
```

ApplyVarCal 算法接受以下可选的 ALGO_OPTION：

- **--sensitivity SENSITIVITY**：确定对可用真实位点的敏感性；只有阈值大于敏感性的等级才会包含在重校准中。我们建议您使用包含在 VarCal 算法的等级阈值列表中的敏感性数字；这将减少舍入问题。默认值为 NULL，因此不应用等级过滤，只应用 LOW_VQSLOD 过滤器。

(23) TNsnv

TNsnv 算法使用基因型分析算法对肿瘤-正常配对样本或肿瘤和正常样本面板数据执行体细胞变异检测。

TNsnv 算法的输入是 BAM 文件；其输出是 VCF 文件。

TNsnv 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

根据运行模式，TNsnv 算法可能需要以下 ALGO_OPTION：

- **--normal_sample SAMPLE_NAME**：BAM 文件中正常样本的 SM 标签名称。
- **--detect_pon**：表示您正在使用 TNsnv 算法创建将成为正常样本面板一部分的 VCF 文件。
- **--cosmic COSMIC_VCF**：用于创建正常样本面板文件的体细胞突变目录（COSMIC）VCF 文件的位置。仅支持一个文件。
- **--pon PANEL_OF_NORMAL_VCF**：包含在正常样本面板分析中检测到的变异的文件位置，用于去除假阳性。仅支持一个文件。

TNsnv 算法接受以下可选的 ALGO_OPTION：

- **--dbsnp dbSNP_FILE**：单核苷酸多态性数据库（dbSNP）的位置。dbSNP 中的变异更有可能被标记为胚系变异，因为它们需要更多证据证明在正常样本中不存在。仅支持一个文件。
- **--call_stats_out CALL_STATS_FILE**：包含体细胞变异检测检测统计信息的文件的位置和文件名。
- **--stdcov_out COVERAGE_FILE**：包含标准覆盖度的 wiggle 文件的位置和

文件名。

- **--tumor_depth_out TUMOR_DEPTH_FILE**：包含肿瘤样本读段深度的 wiggle 文件的位置和文件名。
- **--normal_depth_out NORMAL_DEPTH_FILE**：包含正常样本读段深度的 wiggle 文件的位置和文件名。
- **--power_out POWER_FILE**：功效文件的位置和文件名。
- **--min_base_qual QUALITY**：确定用于变异检测的碱基质量过滤。任何质量低于 QUALITY 的碱基将被忽略。默认值为 5。
- **--min_init_tumor_lod NUMBER**：初始通过检测变异时的最小肿瘤对数比值。默认值为 4。
- **--min_tumor_lod NUMBER**：最终变异检测时的最小肿瘤对数比值。默认值为 6.3。
- **--min_normal_lod NUMBER**：用于检查肿瘤变异不是正常变异的最小正常对数比值。默认值为 2.2。
- **--contamination_frac NUMBER**：来自其他样本的污染比例估计。默认值为 0.02。
- **--min_cell_mutation_frac NUMBER**：具有突变的细胞的最小比例。默认值为 0。
- **--min_strand_bias_lod NUMBER**：检测链偏好的最小对数比值。默认值为 2。
- **--min_strand_bias_power NUMBER**：检测链偏好的最小功效。默认值为 0.9。

- **--min_dbsnp_normal_lod NUMBER**：在 dbSNP 位点检测正常非变异的最小对数比值。默认值为 5.5。
- **--min_normal_allele_frac NUMBER**：在正常样本中考虑的最小等位基因比例；当正常样本被肿瘤样本污染时，此参数很有用。默认值为 0。
- **--min_tumor_allele_frac NUMBER**：肿瘤样本中的最小等位基因比例。默认值为 0.005。
- **--max_indel NUMBER**：允许的最大附近插入/缺失事件数。默认值为 3。
- **--max_read_clip_frac NUMBER**：读段中软/硬裁剪碱基的最大比例。默认值为 0.3。
- **--max_mapq0_frac NUMBER**： 用于确定映射质量差的区域的 mapq 为 0 的读段的最大比例。默认值为 0.5。
- **--min_pir_median NUMBER**：最小读段位置中位数。默认值为 10。
- **--min_pir_mad NUMBER**：最小读段位置中位数绝对偏差。默认值为 3。
- **--max_alt_mapq NUMBER**：替代等位基因映射质量得分的最大值。默认值为 20。
- **--max_normal_alt_cnt NUMBER**：正常堆积中替代等位基因计数的最大值。默认值为 2。
- **--max_normal_alt_qsum NUMBER**：正常堆积中替代等位基因质量得分总和的最大值。默认值为 20。
- **--max_normal_alt_frac NUMBER**：正常堆积中替代等位基因的最大比例。默认值为 0.03。
- **--power_allele_frac NUMBER**：用于功效计算的等位基因比例。默认值为 0.3。

(24) TNhaplotyper

TNhaplotyper 算法使用单倍型分析算法对肿瘤-正常配对样本或肿瘤和正常样本面板数据执行体细胞变异检测。

TNhaplotyper 算法的输入是 BAM 文件；其输出是 VCF 文件。

TNhaplotyper 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

根据运行模式，TNhaplotyper 算法可能需要以下 ALGO_OPTION：

- **--normal_sample SAMPLE_NAME**：BAM 文件中正常样本的 SM 标签名称。
- **--detect_pon**：表示您正在使用 TNhaplotyper 算法创建将成为正常样本面板一部分的 VCF 文件。
- **--cosmic COSMIC_VCF**：用于创建正常样本面板文件的体细胞突变目录（COSMIC）VCF 文件的位置。仅支持一个文件。
- **--pon PANEL_OF_NORMAL_VCF**：包含在正常样本面板分析中检测到的变异的文件位置，用于去除假阳性。仅支持一个文件。

TNhaplotyper 算法接受以下可选的 ALGO_OPTION：

- **--dbsnp dbSNP_FILE**：单核苷酸多态性数据库（dbSNP）的位置。dbSNP 中的变异更有可能被标记为胚系变异，因为它们需要更多证据证明在正常样本中不存在。仅支持一个文件。
- **--min_base_qual QUALITY**：确定用于变异检测的碱基质量过滤。任何质量低于 QUALITY 的碱基将被忽略。默认值为 10。
- **--prune_factor FACTOR**：局部组装中的最小修剪因子；支持 k-mer 数少于 FACTOR 的路径将从图中修剪。默认值为 2。

- **--pcr_indel_model MODEL**: 用于或多或少积极地剔除假阳性插入/缺失的 PCR 插入/缺失模型。可能的模式有： NONE（用于无 PCR 样本），HOSTILE、AGGRESSIVE 和 CONSERVATIVE，按积极程度递减排序。默认值为 HOSTILE。
- **--phasing [1/0]**：启用或禁用输出中的相位信息的标志。
- **--min_init_tumor_lod NUMBER**：初始通过检测变异时的最小肿瘤对数比值。默认值为 4。
- **--min_init_normal_lod NUMBER**：初始通过检测变异时的最小正常对数比值。默认值为 0.5。
- **--min_tumor_lod NUMBER**：最终变异检测时的最小肿瘤对数比值。默认值为 6.3。
- **--min_normal_lod NUMBER**:用于检查肿瘤变异不是正常变异的最小正常对数比值。默认值为 2.2。
- **--min_strand_bias_lod NUMBER**：检测链偏好的最小对数比值。默认值为 2。
- **--min_strand_bias_power NUMBER**：检测链偏好的最小功效。默认值为 0.9。
- **--min_pir_median NUMBER**：最小读段位置中位数。默认值为 10。
- **--min_pir_mad NUMBER**：最小读段位置中位数绝对偏差。默认值为 3。
- **--max_normal_alt_cnt NUMBER**：正常堆积中替代等位基因计数的最大值。默认值为 2。
- **--max_normal_alt_qsum NUMBER**：正常堆积中替代等位基因质量得分总和的最大值。默认值为 20。
- **--max_normal_alt_frac NUMBER**：正常堆积中替代等位基因的最大比例。默认值为 0.03。

- **--tumor_contamination_frac NUMBER**：肿瘤样本来自其他样本的污染比例估计。默认值为 0。
- **--normal_contamination_frac NUMBER**：正常样本来自其他样本的污染比例估计。默认值为 0。
- **--filter_clustered_read_position**：过滤在测序读段开始或结束处聚集的变异。
- **--filter_strand_bias**：过滤显示链偏好证据的变异。
- **--bam_output OUTPUT_BAM**：输出包含变异检测后局部重组修改读段的 BAM 文件。此选项应仅与小型 bed 文件一起用于故障排除目的。
- **--trim_soft_clip**：确定是否应从变异检测中排除读段中的软裁剪碱基。

（25）TNhaplotyper2

TNhaplotyper2 算法使用基于单倍型的算法对肿瘤-正常配对样本或肿瘤和正常样本面板数据执行体细胞变异检测。

TNhaplotyper2 算法的输入是一个或多个 BAM 文件；其输出是一个 VCF 文件，将用于在 TNfilter 中过滤结果；TNhaplotyper2 还会输出一个与输出文件同名但扩展名为.stats 的额外文件，其中包含可帮助过滤的统计信息。

TNhaplotyper2 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

根据运行模式，TNhaplotyper2 算法可能需要以下 ALGO_OPTION：

- **--normal_sample SAMPLE_NAME**：BAM 文件中正常样本的 SM 标签名称。
- **--pon PANEL_OF_NORMAL_VCF**：包含在正常样本面板分析中检测到的变异的文件位置，用于去除假阳性。仅支持一个文件。

TNhaplotyper2 算法接受以下可选的 ALGO_OPTION：

- **--min_base_qual QUALITY**： 确定用于变异检测的碱基质量过滤。任何质量低于 QUALITY 的碱基将被忽略。默认值为 10。
- **--prune_factor FACTOR**：局部组装中的最小修剪因子；支持 k-mer 数少于 FACTOR 的路径将从图中修剪。将修剪因子设置为 0 将启用自适应修剪。默认值为 0。
- **--pcr_indel_model MODEL**: 用于或多或少积极地剔除假阳性插入/缺失的 PCR 插入/缺失模型。可能的模式有： NONE（用于无 PCR 样本），HOSTILE、AGGRESSIVE 和 CONSERVATIVE，按积极程度递减排序。默认值为 CONSERVATIVE。

- **--min_init_tumor_lod NUMBER**：初始通过检测变异时的最小肿瘤对数比值。默认值为 2.0。
- **--min_tumor_lod NUMBER**：最终变异检测时的最小肿瘤对数比值。默认值为 3.0。
- **--min_normal_lod NUMBER**：用于检查肿瘤变异不是正常变异的最小正常对数比值。默认值为 2.2。
- **--germline_vcf VCF**：包含群体等位基因频率的 VCF 文件位置。
- **--default_af AF**：确定在胚系 VCF 中未找到的等位基因的等位基因频率值。在运行肿瘤-正常模式时默认值为 1E-6，在不使用配对正常样本的仅肿瘤模式下运行时默认值为 5E-8。
- **--max_germline_af AF**：确定仅肿瘤模式下的最大胚系等位基因频率。默认值为 0.01。
- **--call_pon_sites**：确定是否检测在正常样本面板输入中存在的候选变异。
- **--callable_depth DEPTH**： 确定一个位点被视为可检测的最小深度，用于包含可帮助过滤的统计信息的额外.stats 文件。
- **--given GIVEN_VCF**：使用 GIVEN_VCF 中提供的变异执行变异检测。除了在发现模式下寻找变异外，还将评估文件中提供的位点和等位基因，但仅当变异的 FILTER 列为 PASS 或.时。
- **--bam_output OUTPUT_BAM**：输出包含变异检测后局部重组修改读段的 BAM 文件。此选项应仅与小型 bed 文件一起用于故障排除目的。
- **--trim_soft_clip**：确定是否应从变异检测中排除读段中的软裁剪碱基。
- **--call_germline_sites**：确定是否检测在胚系位点中存在的候选变异。

(26) TNfilter

TNfilter 算法对 TNhaplotyper2 的输出执行过滤。

TNfilter 算法的输入是包含待过滤候选变异的 VCF 文件；其输出是包含已过滤变异的 VCF 文件；TNfilter 还会输出一个与输出文件同名但扩展名为.stats 的额外文件，其中包含有关过滤的统计信息。

TNfilter 算法需要以下 ALGO_OPTION：

- **-v CANDIDATE_VCF**：包含 TNhaplotyper2 产生的未过滤变异的文件位置和文件名。您可以使用 bgzip 压缩和索引的 VCF 文件。除此文件外，二进制文件还需要 TNhaplotyper2 创建的名为 CANDIDATE_VCF.stats 的统计文件。
- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

根据运行模式，TNfilter 算法可能需要以下 ALGO_OPTION：

- **--normal_sample SAMPLE_NAME**：BAM 文件中正常样本的 SM 标签名称。

TNfilter 算法接受以下可选的 ALGO_OPTION：

- **--orientation_priors PRIORS**：包含 OrientationBias 生成的方向偏差信息的文件位置和文件名。此选项影响方向偏差过滤器。
- **--contamination FILE**：包含 ContaminationModel 生成的污染信息的文件位置和文件名。此选项影响污染过滤器。
- **--tumor_segments SEGMENTS**：包含 ContaminationModel 生成的肿瘤片段信息的文件位置和文件名。此选项影响胚系风险伪影过滤器。
- **--threshold_strategy STRATEGY**：确定应用于优化后验概率阈值的策略。可能的值有：f_score 优先考虑 F 分数，precision 优先考虑减少假阳性，和 constant。默认值为 f_score。

- **--f_score_beta SCORE**：使用--threshold_strategy f_score 时，确定召回率相对于精确度的权重，这将是过滤的目标。默认值为 1，表示精确度和召回率权重相等。
- **--max_fp_rate FP_RATE**：使用--threshold_strategy precision 时，确定预期的最大假阳性率。默认值为 0.05。
- **--threshold THRESHOLD**：使用--threshold_strategy constant 时，确定后验概率阈值。
- **--min_median_base_qual QUALITY**：确定 alt 读段的最小中位碱基质量。此选项影响中位碱基质量过滤器。默认值为 20。
- **--max_event_count COUNT**：确定活跃区域允许的最大事件数。如果活跃区域中的事件超过 COUNT，变异将被聚集事件过滤器过滤。默认值为 2。
- **--unique_alt_reads COUNT**：确定需要支持变异的最小唯一 ALT 读段数。此选项影响重复读段过滤器。默认值为 0。
- **--max_mfrl_diff VALUE**：确定 ALT 和 REF 片段中位数之间的最大差异。此选项影响中位片段长度差异过滤器。默认值为 10000。
- **--max_haplotype_distance DISTANCE**：确定同一单倍型内两个相位变异的最大距离。如果一个变异被过滤，DISTANCE 内的所有相位变异也将被过滤掉，但超出 DISTANCE 的变异不会受到影响。此选项影响单倍型过滤器。默认值为 100。
- **--min_tumor_af AF**：确定最小肿瘤等位基因频率。此选项影响低等位基因频率过滤器。默认值为 0。
- **--min_median_map_qual QUALITY**：确定支持变异的读段的最小中位映射质量。此参数影响中位映射质量过滤器。默认值为 30。

- **--long_indel_length LENGTH**：确定 INDEL 是否使用参考映射质量；长度超过 LENGTH 的 INDEL 将使用参考映射质量。此参数影响中位映射质量过滤器。默认值为 5。
- **--max_alt_count COUNT**：确定一个位点的最大 ALT 等位基因数。此选项影响多等位基因过滤器。默认值为 1。
- **--max_n_ratio RATIO**：确定 N 碱基与 ALT 碱基的最大比率。此选项影响 N 碱基比率过滤器。默认值为 1。
- **--normal_p_value VALUE**：：确定检测正常伪影的 P 值阈值。此选项影响正常伪影过滤器。默认值为 0.001。
- **--min_median_pos DISTANCE**：确定从变异到读段末端的最小中位距离。此选项影响中位读段位置过滤器。默认值为 1。
- **--min_slippage_length LENGTH**：确定 STR 中 REF 碱基的最小长度，以便可能发生聚合酶滑移。此选项影响聚合酶滑移过滤器。默认值为 8。
- **--slippage_rate RATE**：确定可能发生聚合酶滑移区域的聚合酶滑移频率。此选项影响聚合酶滑移过滤器。默认值为 0.1。
- **--min_alt_reads_per_strand COUNT**：确定每条链上所需的最小 ALT 读段数。此选项影响硬链偏差过滤器。默认值为 0。

除上述过滤器外，TNfilter 还可能应用以下过滤器：

- 基于模型的链偏差伪影过滤器，用于可能是链偏差伪影的位点。
- 正常样本面板过滤器，用于存在于正常样本面板中的位点。
- 肿瘤证据过滤器，用于变异证据较弱的位点。

(27) OrientationBias

OrientationBias 算法估计测序数据中可能存在的任何方向偏差。

OrientationBias 算法的输入是一个或多个 BAM 文件；其输出是一个包含方向偏差信息的文件，该文件将在 TNfilter 中过滤 TNhaplotyper2 的结果时使用。

OrientationBias 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

OrientationBias 算法接受以下可选的 ALGO_OPTION：

- **--min_base_qual QUALITY**：决定用于变异调用的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 20。
- **--min_median_map_qual QUALITY**：决定最小中位数映射质量。读序的中位数映射质量低于 QUALITY 的位点将被忽略。默认值为 50。
- **--max_depth DEPTH**：决定分组的深度阈值。深度高于 DEPTH 的位点将被分在一起。默认值为 200。

(28) ContaminationModel

ContaminationModel 算法估计在 TNseq 流程中使用的跨样本污染和肿瘤分段。

ContaminationModel 算法的输入是一个或多个 BAM 文件；其输出是一个包含污染信息的文件，该文件可在 TNfilter 中过滤 TNhaplotyper2 的结果时使用。此外，该工具可能会输出一个 .segments 文件，其中包含可帮助过滤的肿瘤分段信息。

ContaminationModel 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。
- **--vcf VCF**：包含群体等位基因频率的 VCF 文件位置。

根据运行模式的不同，ContaminationModel 算法可能需要以下 ALGO_OPTION：

- **--normal_sample SAMPLE_NAME**：BAM 文件中正常样本的 SM 标签名称。

ContaminationModel 算法接受以下可选的 ALGO_OPTION：

- **--tumor_segments CONTAMINATION.segments**：可选输出文件的位置和文件名，该文件包含肿瘤分段信息；此信息将在 TNfilter 中用于过滤可能是胚系风险伪影的调用。
- **--min_map_qual QUALITY**：决定使用的读序的过滤质量。任何映射质量低于 QUALITY 的读序将被过滤掉。默认值为 50。
- **--min_af AF**：决定群体等位基因频率的最小值。默认值为 0.01。

- **--max_af AF**：决定群体等位基因频率的最大值。默认值为 0.2。

（29）TNscope

TNscope 算法使用单倍型算法对肿瘤-正常配对样本或仅肿瘤数据执行体细胞变异检测。

TNscope 算法的输入是 BAM 文件；其输出是 VCF 文件。

TNscope 算法需要以下 ALGO_OPTION：

- **--tumor_sample SAMPLE_NAME**：BAM 文件中肿瘤样本的 SM 标签名称。

TNscope 算法接受以下可选的 ALGO_OPTION：

- **--annotation 'ANNOTATION'**：确定将添加到输出 VCF 文件中的额外注释。目前支持 FAD 注释。在使用--annotation FAD 选项时，将添加三个额外的注释：FAD、F1R2 和 F2R1。
- **--normal_sample SAMPLE_NAME**： BAM 文件中正常样本的 SM 标签名称。在仅肿瘤体细胞变异检测时，不需要此参数。
- **--cosmic COSMIC_VCF**：用于创建正常样本面板文件的癌症体细胞突变目录（COSMIC）VCF 文件的位置。仅支持一个文件。
- **--pon PANEL_OF_NORMAL_VCF**：包含在正常样本面板分析中检测到的变异的文件位置，用于去除假阳性。仅支持一个文件。此文件与 TNhaplotyper 使用的文件相同。
- **--dbsnp dbSNP_FILE**：单核苷酸多态性数据库（dbSNP）的位置。dbSNP 中的变异更可能被标记为胚系，因为它们需要更多证据证明在正常样本中不存在。仅支持一个文件。
- **--min_base_qual QUALITY**：确定用于变异检测的碱基质量过滤。任何质量低于 QUALITY 的碱基将被忽略。默认值为 15。

- **--prune_factor FACTOR**：局部组装中的最小修剪因子；支持 k-mer 数少于 FACTOR 的路径将从图中修剪。将修剪因子设置为 0 将启用自适应修剪。默认值为 2。
- **--pcr_indel_model MODEL**：用于或多或少积极地剔除假阳性插入/缺失的 PCR 插入/缺失模型。可能的模式有： NONE (用于无 PCR 样本)，HOSTILE、AGGRESSIVE 和 CONSERVATIVE，按积极程度递减排序。默认值为 CONSERVATIVE。
- **--phasing [1/0]**：启用或禁用输出中的相位信息的标志。默认值为 1 (开启)。
- **--min_init_tumor_lod NUMBER**：初始通过检测变异时的最小肿瘤对数比值。默认值为 4。
- **--min_init_normal_lod NUMBER**：初始通过检测变异时的最小正常对数比值。默认值为 0.5。
- **--min_tumor_lod NUMBER**：最终变异检测时的最小肿瘤对数比值。默认值为 6.3。
- **--min_normal_lod NUMBER**：用于检查肿瘤变异不是正常变异的最小正常对数比值。默认值为 2.2。
- **--min_dbsnp_normal_lod NUMBER**：在 dbSNP 位点检测正常非变异的最小对数比值。默认值为 5.5。
- **--tumor_contamination_frac NUMBER**：来自其他样本的肿瘤样本污染比例估计。默认值为 0。
- **--normal_contamination_frac NUMBER**：来自其他样本的正常样本污染比例估计。默认值为 0。

- **--given GIVEN_VCF**：仅使用 GIVEN_VCF 中提供的变异执行变异检测。仅评估文件中提供的位点和等位基因，且仅当变异的 FILTER 列为 PASS 或.时。
- **--bam_output OUTPUT_BAM**：输出包含变异检测后局部重组修改读段的 BAM 文件。此选项应仅与小型 bed 文件一起用于故障排除目的。
- **--disable_detector DETECTOR**：禁用特定检测器的变异检测：使用'sv'作为 DETECTOR 以防止检测结构变异，使用'snv_indel'作为 DETECTOR 以防止检测小变异。
- **--trim_soft_clip**：确定是否应从变异检测中排除读段中的软裁剪碱基。
- **--trim_primer AMPLICON_TABLE**：确定是否根据 AMPLICON_TABLE 中提供的信息修剪引物。AMPLICON_TABLE 是一个 BED 文件，包含扩增子测序中引物位置的信息，是一个包含八列的制表符分隔文件：contig、amplicon_start、amplicon_end、name（忽略）、score（忽略）、strand（忽略）、insert_start、insert_end；该工具将修剪 amplicon_start 和 insert_start 之间以及 amplicon_end 和 insert_end 之间的扩增子读段碱基。
- **--trim_adaptor=[0|1]**：确定是否从输入读段中修剪接头碱基。默认是修剪接头碱基。设置为 0 以关闭。

（30）RNASplitReadsAtJunction

RNASplitReadsAtJunction 算法通过去除 N 但保留分组信息来将读段分割成外显子片段，并对任何延伸到内含子区域的序列进行硬修剪。

RNASplitReadsAtJunction 算法的输入是 BAM 文件；其输出也是 BAM 文件。

RNASplitReadsAtJunction 算法需要以下 ALGO_OPTION：

- **--reassign_mapq IN_QUAL：OUT_QUAL**：算法将从 IN_QUAL 重新分配映射质量到 OUT_QUAL。此参数是必需的，因为 STAR 为良好的比对分配 255 的质量，而不是预期的默认分数 60。

RNASplitReadsAtJunction 算法接受以下可选的 ALGO_OPTION：

- **--ignore_overhang**：确定是否忽略且不修复读段的悬挂部分。
- **--overhang_max_bases NUMBER**：确定硬修剪悬挂部分允许的最大碱基数，如果悬挂部分的碱基数超过此值，则不会进行硬修剪。默认值为 40。
- **--overhang_max_mismatches NUMBER**：确定非硬修剪悬挂部分允许的最大错配数，如果错配数太高，则会对整个悬挂部分进行硬修剪。默认值为 1。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]**：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。

（31）ContaminationAssessment

ContaminationAssessment 算法评估样本 BAM 文件中存在的污染；此算法的输出可用作 TNseq 和 TNscope 工具中 contamination_frac、normal_contamination_frac 和 tumor_contamination_frac 参数的值。ContaminationAssessment 算法的输入是一个 BAM 文件；其输出是一个文本文件。

ContaminationAssessment 算法需要以下 ALGO_OPTION：

- **--pop_vcf VCF_FILE**：包含样本特定人群等位基因频率信息的 VCF 文件位置。
- **--genotype_vcf VCF_FILE**：包含为个体报告的 DNAseq 变异的 VCF 文件位置；要计算肿瘤样本中的污染，应使用为正常样本报告的 DNAseq 变异。您可以通过在样本 bam 上使用 Haplotyper 或 Genotyper 创建此文件。

ContaminationAssessment 算法接受以下可选的 ALGO_OPTION：

- **--type ASSESS_TYPE**：确定估计的类型。可能的值为 SAMPLE、READGROUP 和 META，分别用于按样本、按测序通道或跨所有读段进行污染评估。允许多个选项实例，以在多个层面评估污染。默认值为 META。
- **--min_base_qual QUALITY**：确定用于污染评估的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 20。
- **--min_map_qual QUALITY**：确定用于污染评估的读段的过滤质量。任何质量低于 QUALITY 的读段将被忽略。默认值为 20。
- **--min_basecount NUMBER**：确定在评估污染之前在一个位点需要存在的最小碱基数。默认值为 500。

- **--trim_thresh NUMBER**：用于修剪位点的阈值；如果污染比率大于 0.5 的概率大于阈值，则该位点不会包含在污染评估中。默认值为 0.95。
- **--trim_frac NUMBER**：确定基于修剪阈值可能被修剪的最大位点比例。默认值为 0.01。
- **--precision NUMBER**：确定输出百分比数字的精度。默认值为 0.1。
- **--base_report FILE**：包含处理数据扩展报告的输出文件的位置和文件名。
- **--population POPULATION_NAME**：用于确定样本基线等位基因频率的人群。默认值为 CEU。

（32）TNModelApply

TNModelApply 算法将机器学习模型应用于 TNscope 的结果，以帮助进行变异过滤。此算法仅在 Sentieon Genomics 软件的 Linux 版本中支持。

TNModelApply 算法在驱动程序级别没有输入，其输出是一个 VCF 文件。

TNModelApply 算法需要以下 ALGO_OPTION：

- **-v INPUT**：来自 TNscope 变异调用的 VCF 文件位置；您可以使用经 bgzip 压缩并已索引的 VCF 文件。
- **-m MODEL_FILE**：包含机器学习模型的文件位置。

TNModelApply 算法通过向变异添加 MLrejected FILTER 来修改输入的 VCF 文件；由于添加了 FILTER，您可能希望删除输入 VCF 中已存在的任何 FILTER，因为它们可能不再相关。您可以使用 bcftools（*https://samtools.github.io/bcftools/bcftools.html*）来实现这一目的：

```
$BCF/bcftools annotate -x "^FILTER/MLrejected,FILTER/PASS" -O z \
  -o $OUTPUT.vcf.gz $INPUT.vcf.gz
```

(33) DNAscope

DNAscope 算法执行改进版的单倍型变异检测。

DNAscope 算法的输入是 BAM 文件；输出是 VCF 文件。

DNAscope 算法接受以下可选的 ALGO_OPTION：

- **--annotation 'ANNOTATION_LIST'**：确定将添加到输出 VCF 的其他注释。使用逗号分隔的列表来启用或禁用注释。包括'none'以删除默认注释；在注释前加感叹号（!）以禁用特定注释。有关支持的注释，请参见 *第 6.2 节*。
- **-d dbSNP_FILE**：用于标记已知变异的单核苷酸多态性数据库（dbSNP）的位置。仅支持一个文件。
- **--var_type VARIANT_TYPE**：确定将检测哪些变异类型；VARIANT_TYPE 是以下可能值的逗号分隔列表：
  - SNP 检测单核苷酸多态性。这包含在默认行为中。
  - INDEL 检测插入-缺失。这包含在默认行为中。
  - BND 检测结构变异检测器所需的断点信息。
- **--call_conf CONFIDENCE**：确定检测变异的变异质量阈值。质量低于 CONFIDENCE 的变异将被删除。
- **--emit_conf CONFIDENCE**：确定发射变异的变异质量阈值。质量低于 CONFIDENCE 的变异将不会添加到输出 VCF 文件中。
- **--emit_mode MODE**：确定将发射哪些检测。MODE 的可能值为：
  - variant：仅在置信的变异位点发射检测。这是默认行为。
  - confident：在置信的变异位点或置信的参考位点发射检测。
  - all：发射所有检测，无论其置信度如何。

- gvcf：发射联合检测所需的附加信息。如果要使用 GVCFtyper 算法执行联合检测，则需要此选项。

- **--gq_bands LIST_OF_BANDS**：确定用于压缩具有相似基因型质量（GQ）的变异的带，这些变异将在 GVCF 输出文件中作为单个 VCF 记录发射。LIST_OF_BANDS 是一个逗号分隔的带列表，其中每个带由 START-END/STEP 定义。默认值为 1-60，60-99/10，99。
- **--min_base_qual QUALITY**：确定用于变异检测的碱基的过滤质量。任何质量低于 QUALITY 的碱基将被忽略。默认值为 10。
- **--pcr_indel_model MODEL**：用于或多或少积极地剔除假阳性插入缺失的 PCR 插入缺失模型。可能的 MODEL 有：NONE（用于无 PCR 样本），以及 HOSTILE、AGGRESSIVE 和 CONSERVATIVE，按降低积极性的顺序排列。默认值为 CONSERVATIVE。
- **--phasing [1/0]**：启用或禁用输出中的相位的标志。默认值为 1（开启）。相位仅为二倍体样本计算。
- **--ploidy PLOIDY**：确定正在处理的样本的倍性数。默认值为 2。
- **--prune_factor FACTOR**：局部组装中的最小修剪因子；支持 k-mer 少于 FACTOR 的路径将从图中修剪。默认值为 2。
- **--trim_soft_clip**：确定是否应从变异检测中排除读段中的软剪切碱基。仅建议处理 RNA 读段时使用此参数。
- **--given GIVEN_VCF**：仅使用 GIVEN_VCF 中提供的变异执行变异检测。检测将仅评估文件中提供的位点和等位基因，并且仅当变异的 FILTER 列为 PASS 或.时。此选项不能与--emit_mode gvcf 一起使用。

- **--bam_output OUTPUT_BAM**：输出包含变异检测所做的局部重组后修改读段的 BAM 文件。此选项仅应与小型 bed 文件一起用于故障排除目的。
- **--filter_chimeric_reads**：确定在检测变异时是否使用嵌合读段。默认情况下，仅当设置了 var_type BND 时才包括嵌合读段。
- **--trim_adaptor**：确定是否从输入读段中修剪适配器碱基。默认是不修剪适配器碱基，因为这可能会影响 SV 周围的区域。
- **--model MODEL_FILE**：将与 DNAModelApply 工具一起使用的机器学习模型文件的位置；该模型将用于确定变异检测中使用的设置。

(34) DNAModelApply

DNAModelApply 算法执行使用 DNAscope 进行变异调用的第二步。此算法仅在 Sentieon Genomics 软件的 Linux 版本中支持。

DNAModelApply 算法在驱动程序级别没有输入，其输出是一个包含变异的 VCF 文件。

DNAModelApply 算法需要以下 ALGO_OPTION：

- **-v INPUT**：来自 DNAscope 变异调用算法的 VCF 文件位置，该算法使用输入模型文件确定正确的设置执行。您可以使用经 bgzip 压缩并已索引的 VCF 文件。
- **-m MODEL_FILE**：机器学习模型文件的位置；此文件应与用于生成输入 VCF 的 DNAscope 命令中使用的文件相同。

DNAModelApply 算法不支持任何可选的 ALGO_OPTION。

DNAModelApply 算法通过向变异添加 MLrejected FILTER 来修改输入的 VCF 文件；由于添加了 FILTER，您可能希望删除输入 VCF 中已存在的任何 FILTER，因为它们可能不再相关。您可以使用 bcftools (*https://samtools.github.io/bcftools/bcftools.html*) 来实现这一目的：

```
$BCF/bcftools annotate -x "^FILTER/MLrejected,FILTER/PASS" -O z \
  -o $OUTPUT.vcf.gz $INPUT.vcf.gz
```

(35) SVSolver

SVSolver 算法执行样本的结构变异检测，前提是该样本之前已使用 DNAscope 算法进行处理，并使用了 --var_type bnd 选项。

SVSolver 算法在驱动程序级别没有输入，其输出是一个包含检测到的结构变异的 VCF 文件。

SVSolver 算法需要以下 ALGO_OPTION：

- **-v INPUT**: 使用 --var_type bnd 选项对样本执行 DNAscope 变异检测算法所得到的 VCF 文件的位置。您可以使用 bgzip 压缩并索引的 VCF 文件。

SVSolver 算法不支持任何可选的 ALGO_OPTION。

(36) VariantPhaser

VariantPhaser 算法通过使用包含长读序的 BAM 文件中的读序信息，对 VCF 文件中的变异进行基于读序的相位分析。

VariantPhaser 算法的输入是一个 BAM 文件和一个 VCF 文件；其输出是相位分析后的 VCF 文件。

VariantPhaser 算法需要以下 ALGO_OPTION：

- **-v INPUT**：包含需要进行相位分析的变异的 VCF 文件的位置。您可以使用 bgzip 压缩并索引的 VCF 文件。

5.1.3 DRIVER read_filter 选项

DRIVER 二进制程序的--read_filter 参数允许在执行计算之前过滤或转换输入 BAM 文件中的 reads。可以在同一命令行中使用多个--read_filter 参数，它们将按顺序执行，因此顺序很重要。

参数语法为：--read_filter FILTER,OPTION=VALUE,OPTION=VALUE,...

（1）QualCalFilter read_filter

QualCalFilter read 过滤器用于转换 reads 并执行碱基质量分数重校准，同时修改重校准表中包含的信息。

QualCalFilter read 过滤器需要以下选项之一：

- **table=TABLE_FILEPATH**：用作执行碱基质量分数重校准基础的重校准表的位置。
- **use_oq=[true/false]**：确定是否使用 BAM 文件中 OQ 标签包含的原始碱基质量分数。此选项不能与 table 选项一起使用，它用于通过将输出的碱基质量分数设置为输入 BAM 文件中 OQ 标签中包含的分数来撤销碱基质量分数重校准。通常，此选项将在第二个 QualCalFilter read 过滤器之前使用，以首先撤销输入 BAM 文件上可能进行的重校准。

QualCalFilter read 过滤器接受以下可选选项：

- **prior=PRIOR**：确定所有碱基质量分数的全局偏差。
- **min_qual=QUAL**：确定执行重校准的质量阈值；质量分数小于 QUAL 的碱基将不会被重校准。
- **levels=LEVEL1/LEVEL2/...**：确定碱基质量分数的静态量化级别。
- **indel=[true/false]**：确定是否将 INDEL 的碱基质量分数添加到 BAM 标签中。
- **keep_oq=[true/false]**：确定是否通过使用 BAM 文件中的 OQ 标签来保留重校准前的原始碱基质量分数。

(2) OverclippingFilter read_filter

OverclippingFilter read 过滤器用于根据 reads 的软裁剪特征进行过滤。

OverclippingFilter read 过滤器接受以下可选选项：

- **min_align_len=LENGTH**：过滤非软裁剪碱基数少于 LENGTH 的 reads。
- **count_both_ends=[true/false]**：如果设置为 true，则仅过滤 read 两端都有软裁剪的 reads，这样只有一端有软裁剪的 reads 将不会被过滤，无论其非软裁剪长度如何。默认值为 true。

(3) SimplifyCigarTransform read_filter

SimplifyCigarTransform read 过滤器用于按以下规则简化和标准化 CIGAR：

- 长度为零的操作符：删除。
- EQUAL 操作符(=)转换为 MATCH 操作符(M)。
- DIFF 操作符(X)转换为 MATCH 操作符(M)。
- 相邻的 INS(I)和 DEL(D)操作符合并为 MATCH(M)。
- 尾部 DEL 操作符(D)：删除。

5.2 BWA 二进制程序

BWA 二进制程序用于执行 DNA-seq 数据的比对。

在 Sentieon®版本 202112 中，修复了原始 bwa mem 中导致少量比对的 MAPQ 不正确的 bug。提供了一个兼容性环境变量 ksw_compat=1，设置此环境变量将使 Sentieon® bwa mem 的行为与 *http://bio-bwa.sourceforge.net/bwa.shtml* 中描述的相同，为某些比对提供不正确的 MAPQ。默认行为为所有比对实现了正确的 MAPQ 计算，并在一些较新的硬件上提供了速度提升。

BWA 二进制程序有两种重要模式："mem"模式用于将 FASTQ 文件与参考 FASTA 文件进行比对，以及"shm"模式用于将 FASTA 索引文件加载到内存中，以便在同一服务器上运行的多个 BWA 进程之间共享。

5.2.1 BWA mem 语法

您可以运行以下命令将单端 FASTQ1 文件或一对双端 FASTQ 文件与 FASTA 参考进行比对，这将产生映射的读段到标准输出，以便通过管道传输到 util sort：

```
<SENTIEON_FOLDER>/bin/sentieon bwa mem OPTIONS FASTA FASTQ1 [FASTQ2]
```

此命令的参数（OPTIONS）包括：

- **-t NUMBER_THREADS**：软件将用于运行并行进程的计算线程数。此参数是可选的；如果省略，bwa 二进制工具将使用 1 个线程。
- **-p**：确定第一个输入 FASTQ 文件是否包含交错的双端读段。如果使用此参数，则只使用单个 FASTQ 输入，因为第二个 FASTQ2 文件将被忽略。
- **-M**：确定是否将分裂读段标记为次要。

- **-R READGROUP_STRING**：所有读段将附加的读组头行。推荐的是 @RG\tID：$readgroup\tSM：$sample\tPL：$platform\tPU：$platform_unit
  - $readgroup 是标识读段的唯一 ID。
  - $sample 是读段所属的样本的名称。
  - $platform 是测序技术，通常是 ILLUMINA。
  - $platform_unit 是执行测序的测序元素。
- **-K CHUNK_SIZE**：确定将同时映射的读段组的大小。如果未设置此参数，结果将取决于使用的线程数。

5.2.2 BWA shm 语法

您可以运行以下命令将 FASTA 索引文件加载到内存中：

```
<SENTIEON_FOLDER>/bin/sentieon bwa shm FASTA
```

您可以运行以下命令列出存储在内存中的 FASTA 索引文件：

```
<SENTIEON_FOLDER>/bin/sentieon bwa shm -l
```

您可以运行以下命令删除存储在内存中的所有 FASTA 索引文件，从而在不再需要时释放内存：

```
<SENTIEON_FOLDER>/bin/sentieon bwa shm -d
```

此命令的参数（OPTIONS）包括：

- **-t NUMBER_THREADS**：软件将用于运行并行进程的计算线程数。此参数是可选的；如果省略，bwa 二进制工具将使用 1 个线程。
- **-f FILE**：将用于减少峰值内存使用的临时文件的位置。

5.2.3 控制 BWA 中的内存使用

默认情况下，BWA 在 Linux 系统中将使用约 24 GB，在 Mac 系统中将使用 8 GB。您可以通过 bwt_max_mem 环境变量控制内存使用，这可以通过使用更多内存来提高速度性能，或者以牺牲速度性能为代价限制内存使用。例如，通过在脚本中添加以下内容，您将获得更快的比对：

```
export bwt_max_mem=50G
```

请记住，您在环境变量中使用的数字不是硬限制，而是 BWA 使用内存的估计值；因此，如果 BWA 内存使用不超过某个特定值以下的任何值，这意味着它是特定参考所需的最小内存，设置为小于最小所需内存的值不会改变 BWA mem 作业的内存使用。

5.2.4 使用现有 BAM 文件作为输入

如果您无法访问 FASTQ 输入，而只有已比对和排序的 BAM 文件，您可以通过运行 samtools 将其用作输入并重新进行比对：

```
samtools collate -@ 32 -Ou INPUT_BAM tmp- | samtools fastq -@ 32 -s \
/dev/null -0 /dev/null - | <SENTIEON_FOLDER>/bin/sentieon bwa mem -t 32 -R \
'@RG\tID:id\tLB:lib\tSM:sample\tPL:ILLUMINA' -M -K 1000000 -p $ref /dev/stdin \
| <SENTIEON_FOLDER>/bin/sentieon util sort -t 32 -o OUTPUT_BAM --sam2bam -
```

或者，您可以先创建 FASTQ 文件，然后像通常那样处理它们：

```
samtools collate -n -@ 32 -uO INPUT_BAM tmp- | samtools fastq -@ 32 \
-s >(gzip -c > single.fastq.gz) -0 >(gzip -c > unpaired.fastq.gz) \
-1 >(gzip -c > output_1.fastq.gz) -2 >(gzip -c > output_2.fastq.gz) -
```

如果您这样做，可能会遇到 BWA 中异常的内存使用；如果出现这种情况，您可以按照 *第 7.2.5 使用从 BAM 文件创建的 FASTQ 文件时使用异常数量的内存* 中的说明进行操作。

5.3 STAR 二进制程序

STAR 二进制程序用于执行 RNA-seq 数据的比对，其行为与 *https://github.com/alexdobin/STAR* 中描述的工具相同。

5.3.1 使用压缩的 FASTQ.gz 输入文件

使用压缩的 FASTQ 输入文件需要使用 STAR 选项 --readFilesCommand COMMAND 来确定将使用哪个外部程序来解压输入文件。当使用 Sentieon® 实现的 STAR 与 Sentieon® util sort 一起使用时，高效的输入处理可能会导致解压成为瓶颈，因此使用快速解压方法非常重要。

5.4 Minimap2 二进制程序

minimap2 二进制程序用于执行 PacBio 或 Oxford Nanopore 基因组测序数据的比对，其行为与 *https://github.com/lh3/minimap2* 中描述的工具相同。

5.5 UTIL 二进制程序

UTIL 是用于运行一些实用功能的二进制程序。这个二进制程序主要用于处理来自 BWA 的原始读段输出。

5.5.1 UTIL 语法

UTIL 二进制程序的一般语法是：

```
sentieon util MODE [OPTIONS]
```

该命令支持的模式（MODE）包括：

- **index**：为 BAM 文件建立索引。以下命令将在输入文件相同位置生成 bai BAM 索引文件：

```
sentieon util index INPUT.bam
```

- **vcfindex**：为 VCF 文件建立索引。以下命令将在输入文件相同位置生成 idx VCF 索引文件：

```
sentieon util vcfindex INPUT.vcf
```

- **sort**：对 BAM 文件进行排序。使用 sort 模式的 UTIL 命令的可选参数（OPTIONS）包括：
  - **-t NUMBER_THREADS**：软件用于运行并行进程的计算线程数。默认使用服务器上所有可用的线程。
  - **-r REFERENCE**：参考 FASTA 文件的位置。如果使用 CRAM 输出文件则此参数必需，否则为可选。
  - **-i INPUT**：输入文件的位置。
  - **-o OUTPUT**：输出文件的位置和文件名。
  - **--temp_dir DIRECTORY**：确定临时文件的存储位置。默认为运行命令

的文件夹（$PWD）。

- **--cram_read_options decode_md=0**：CRAM 输入选项，用于关闭输入 CRAM 中的 NM/MD 标签。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]**：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。
- **--bam_compression COMPRESSION_LEVEL[0-9]**：输出 BAM 文件的 gzip 压缩级别。默认值为 6。
- **--sam2bam**：表示输入将是未压缩的 SAM 文件，需要转换为 BAM。如果不使用此选项，输入应该已经使用 samtools 从 BWA 输出转换为 BAM 格式。
- **--block_size BLOCK_SIZE**：用于排序的块大小。
- **--trim_primer AMPLICON_TABLE,clip=CLIPPING**：确定是否将引物标记为软/硬剪切，取决于 CLIPPING 的值是 soft 还是 hard。当使用选项--trim_primer AMPLICON_TABLE,clip=soft 时，需要确保在后续的变异调用命令中使用--trim_soft_clip 选项，以便忽略软剪切的引物。引物基于 AMPLICON_TABLE 中提供的信息确定。AMPLICON_TABLE 是一个 BED 文件，包含扩增子测序中引物位置的制表符分隔文件，有八列：contig、amplicon_start、amplicon_end、name（忽略）、score（忽略）、strand（忽略）、insert_start、insert_end；该工具将修剪 amplicon_start 和 insert_start 之间，以及 amplicon_end 和

insert_end 之间的扩增子读段碱基。

```
sentieon util sort -t NUMBER_THREADS --sam2bam -i INPUT.sam -o OUTPUT.bam
```

- **vcfconvert**：压缩和解压 VCF 和 GVCF 文件。

以下命令将压缩并为输入文件建立索引

```
sentieon util vcfconvert INPUT.vcf OUTPUT.vcf.gz
```

以下命令将解压使用 gzip 生成的非索引 vcf 文件，然后压缩并建立索引。使用此命令时确保输入和输出文件不相同：

```
sentieon util vcfconvert INPUT.gz OUTPUT.vcf.gz
```

- **stream**：以流模式执行碱基质量校正。使用 stream 模式的 UTIL 命令的可选参数（OPTIONS）包括：
  - **-r REFERENCE**：参考 FASTA 文件的位置。如果使用 CRAM 输出文件则此参数必需，否则为可选。
  - **-i INPUT**：输入文件的位置。默认情况下二进制程序将使用 stdin 作为输入。
  - **-q RECAL_TABLE**：重校准表的位置。
  - **-t NUMBER_THREADS**：软件用于运行并行进程的计算线程数。默认值为 1。
  - **-o OUTPUT**：输出文件的位置和文件名。默认情况下二进制程序将输出到 stdout，以便能够流式传输结果。
  - **--output_format FORMAT**：确定输出流的格式。可能的 FORMAT 值为 BAM 或 CRAM。默认为 BAM。
  - **--output_index_file OUTPUT_INDEX**：确定将在何处创建相应的索引文件。如果省略此选项，工具将不生成索引文件。

- **--read_filter FILTER,OPTION=VALUE,OPTION=VALUE**：在应用算法之前执行读段过滤或转换。请参阅驱动程序使用说明的 第 *5.1.3 节* 获取有关可用过滤器及其功能的其他信息。
- **--cram_read_options decode_md=0**：CRAM 输入选项，用于关闭输入 CRAM 中的 NM/MD 标签。
- **--cram_write_options compressor=[gzip|bzip2|lzma|rans]** ：CRAM 输出压缩选项。如果未定义，默认为 compressor=gzip。
- **--cram_write_options version=[2.1|3.0]**：CRAM 输出版本选项。如果未定义，默认为 version=3.0。
- **--bam_compression COMPRESSION_LEVEL[0-9]**：输出 BAM 文件的 gzip 压缩级别。默认值为 6。

以下命令将应用重校准并将相应的重校准 BAM 文件输出到 stdout：

```
sentieon util stream -i INPUT.bam -q RECAL_TABLE \
--output_index_file OUTPUT.bam.bai -o -
```

除非包含 output_index_file 选项，否则上述命令不会生成索引文件。

5.6 UMI 二进制程序

UMI 是用于处理包含 UMI 序列的读段的二进制程序。

5.6.1 UMI 语法

UMI 二进制程序的一般语法是：

```
sentieon umi MODE [OPTIONS]
```

该命令支持的模式（MODE）包括：

- **extract**：预处理包含 UMI 序列读段的 FASTQ 文件。extract 模式的语法是：

```
sentieon umi extract [OPTIONS] read_structure fastq1 [fastq2] [fastq3]
```

其中：

  - read_structure 是读段的逻辑结构。它由描述#bases+type 的整数+字符对组成；类型可以是 M（分子条形码）、T（模板）和 S（跳过）。读段结构由逗号分隔的组组成，每组将从相应的输入 FASTQ 读段（第一组来自第一个 FASTQ，第二组来自第二个 FASTQ...）
  - fastq1/2/3 是 FASTQ 文件。最多支持 3 个输入 FASTQ 文件，以允许 UMI 序列已经在单独的 FASTQ 文件中的用例。

使用 extract 模式的 UMI 二进制程序的可选参数（OPTIONS）包括：

  - **-o OUTPUT**：输出文件的位置和文件名。如果省略，输出将是 stdout。
  - **-d**：如果存在，将以双工模式进行提取。
  - **--umi_tag TAG**：逻辑 UMI 标签。默认值为 XR。

5.7 PLOT 脚本

PLOT 是用于创建度量和重校准阶段结果图的脚本。图表存储在 PDF 文件中。

5.7.1 PLOT 语法

PLOT 脚本的一般语法是：

```
sentieon plot STAGE -o OUTPUPT_FILE INPUTS [OPTIONS]
```

该命令支持的模式（STAGE）包括：

- **GCBias**：从 GCBias 的度量结果生成 PDF 文件。
- **QualDistribution**：从 QualDistribution 的度量结果生成 PDF 文件。
- **InsertSizeMetricAlgo**：从 InsertSizeMetricAlgo 的度量结果生成 PDF 文件。
- **MeanQualityByCycle**：从 MeanQualityByCycle 的度量结果生成 PDF 文件。
- **QualCal**：从 BQSR QualCal 工具生成 PDF 文件。
- **VarCal**：从 VQSR VarCal 工具生成 PDF 文件。

（1） PLOT 绘制 GCBias 阶段的结果

生成 GCBias 阶段图的 INPUTS 是：

- **GC_METRIC_TXT**：即 *第 5.1.2 节* 中 GCBias 算法的输出文件。

GCBias 度量 STAGE 的绘图不接受任何 OPTIONS。

（2）绘制 MeanQualityByCycle 阶段的结果

生成 MeanQualityByCycle 阶段图的 INPUTS 是：

- **MQ_METRIC_TXT**：即 *第 5.1.2 节* 中 MeanQualityByCycle 算法的输出文件。

MeanQualityByCycle 度量 STAGE 的绘图不接受任何 OPTIONS。

（3）绘制 QualDistribution 阶段的结果

生成 QualDistribution 阶段图的 INPUTS 是：

- **QD_METRIC_TXT**：即 *第 5.1.2 节* 中 QualDistribution 算法的输出文件。

QualDistribution 度量 STAGE 的绘图不接受任何 OPTIONS。

（4）绘制 InsertSizeMetricAlgo 阶段的结果

生成 InsertSizeMetricAlgo 阶段图的 INPUTS 是：

- **IS_METRIC_TXT**：即 *第 5.1.2 节* 中 InsertSizeMetricAlgo 算法的输出文件。

InsertSizeMetricAlgo 度量 STAGE 的绘图不接受任何 OPTIONS。

（5）绘制 QualCal 阶段的结果

生成 bqsr 阶段图的 INPUTS 是：

- **RECAL_RESULT.CSV**：即 *第 5.1.2 节* 中 QualCal 算法的输出 csv 文件。

bqsr STAGE 的绘图不接受任何 OPTIONS。

（6）脚本绘制 VarCal 阶段的结果

生成 vqsr 阶段图的 INPUTS 是：

- **PLOT_FILE**：由 *第 5.1.2 节* 中 VarCal 算法创建的文件，包含创建报告所需的数据。

vqsr STAGE 的绘图接受以下 OPTIONS：

- **tranches_file=TRANCHES_FILE**：包含调用集分区到质量梯度的文件位置，由 *第 5.1.2 节* 中的 VarCal 算法生成。
- **target_titv=TITV_THRES**：物种的预期 TiTv 数；用于计算图中的真阳性和假阳性数。

- **min_fp_rate=MIN_RATE**：最小假阳性数；用于计算图中的真阳性和假阳性数。

5.8 LICSRVR 二进制程序

LICSRVR 是用于运行许可证服务器的二进制程序，用于促进动态许可证分配和记录集群内的许可证使用情况。

5.8.1 LICSRVR 语法

LICSRVR 二进制程序的一般语法是：

```
<SENTIEON_FOLDER>/bin/sentieon licsrvr [--start|--stop] [--log LOG_FILE] LICENSE_FILE
```

该命令的以下输入是可选的：

- **LOG_FILE**：包含服务器日志的输出文件的位置和文件名。
- **LICENSE_FILE**：服务器许可证文件的位置。

许可证服务器运行后，客户端应用程序可以通过将 SENTIEON_LICENSE 环境变量设置为形式为 HOST:PORT 的服务器地址来从服务器请求许可证令牌。

licsrvr 二进制程序支持以下附加模式：

- **--version**：将报告二进制程序所属的软件包版本。
- **--dump**：将报告许可证服务器的当前状态，包括可用许可证的数量。

```
<SENTIEON_FOLDER>/bin/sentieon licsrvr --dump LICENSE_FILE
```

- **--dump=update**：如果许可证信息已更新并被许可证服务器自动拉取，此模式将更新的许可证信息转储到 stdout。以下命令将报告许可证服务器拥有的更新许可证信息（如果有任何更改）：

```
<SENTIEON_FOLDER>/bin/sentieon licsrvr --dump=update LICENSE_FILE
```

5.9 LICCLNT 二进制程序

LICCLNT 是用于测试许可证服务器功能的二进制程序，帮助确定许可证服务器是否运行，以及不同算法有多少可用许可证。

5.9.1 LICCLNT 语法

LICCLNT 二进制程序有两种模式，一种用于 ping 许可证服务器，另一种用于检查特定算法的可用许可证。

你可以运行以下命令来检查许可证服务器是否运行：

```
<SENTIEON_FOLDER>/bin/sentieon licclnt ping --server HOST:PORT
```

如果服务器运行，命令将返回 0。

你可以运行以下命令来检查有多少许可证：

```
<SENTIEON_FOLDER>/bin/sentieon licclnt query --server HOST:PORT FEATURE
```

该命令将返回特定许可证功能可用的许可证数量，这可用于管理你的作业：在提交使用特定线程数的作业之前，你可以检查是否有足够的许可证用于这些线程，防止工具在等待许可证时闲置。

6 工具功能

6.1 处理多个输入文件

通常，有两种情况需要同时处理多个输入文件：单个样本在多个测序通道中测序，或者处理来自一个群体的多个样本。

6.1.1 处理来自同一样本的多个输入文件

当您有多于一组输入 fastq 文件时，应该为每组输入 fastq 文件执行单独的比对阶段，然后使用多个排序后的 BAM 文件作为下一阶段的输入。下一阶段将负责将所有 BAM 文件合并为一个。您需要确保每组输入 fastq 文件使用相同的 SM 样本名称但不同的 RG 读组名称进行比对，以便后续阶段能正确处理不同的数据。

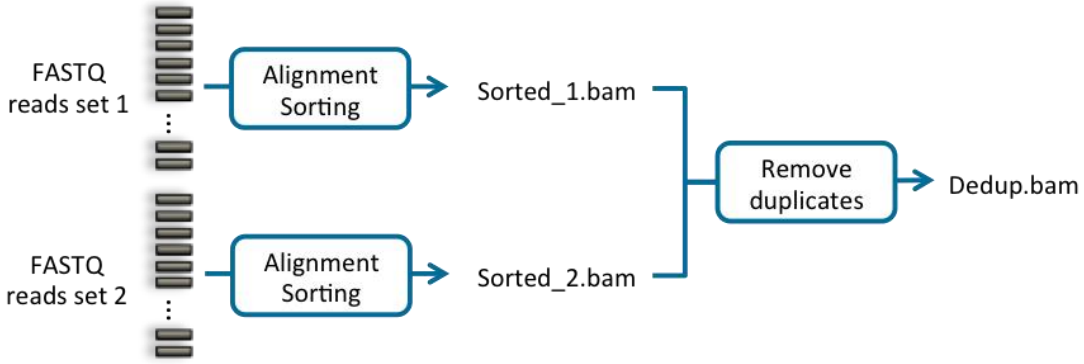

图 6-1 使用来自同一样本的多个输入文件的生物信息学流程

```
#Run alignment for 1st input file set
(sentieon bwa mem -R '@RG\tID:GROUP_NAME_1 \tSM:SAMPLE_NAME \tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE SAMPLE_1 [SAMPLE_1_2] || echo -n 'error' ) \
  | sentieon util sort -o SORTED_BAM_1 -t NUMBER_THREADS --sam2bam -i -
#Run alignment for 2nd input file set
```

```
(sentieon bwa mem -R '@RG\tID:GROUP_NAME_2 \tSM:SAMPLE_NAME \tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE SAMPLE_2 [SAMPLE_2_2] || echo -n 'error' \
  sentieon util sort -o SORTED_BAM_2 -t NUMBER_THREADS --sam2bam -i -
#Run dedup on both BAM files
sentieon driver -t NUMBER_THREADS -i SORTED_BAM_1 -i SORTED_BAM_2 \
  --algo LocusCollector --fun score_info SCORE_TXT
sentieon driver -t NUMBER_THREADS -i SORTED_BAM_1 -i SORTED_BAM_2 \
  --algo Dedup --rmdup --score_info SCORE_TXT --metrics DEDUP_METRIC_TXT DEDUPED_
BAM
```

另外，您也可以将多个 BWA 比对的结果输入到单个 util sort 命令中，以尽快将结果合并到一个排序好的 bam 文件中。您仍然需要确保每组输入的 fastq 文件在比对时使用相同的 SM 样本名称，但使用不同的 RG 读组名称，这样后续阶段才能正确处理不同的数据。

```
#Run alignment for both input file sets sequentially
(sentieon bwa mem -R '@RG\tID:GROUP_NAME_1\tSM:SAMPLE_NAME\tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE SAMPLE_1 [SAMPLE_1_2]  && sentieon bwa mem -
R \
  '@RG\tID:GROUP_NAME_2 \tSM:SAMPLE_NAME \tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE SAMPLE_2 [SAMPLE_2_2] || echo -n 'error' ) \
  | sentieon util sort -o SORTED_COMBINED_BAM -t NUMBER_THREADS --sam2bam -i -
#Run dedup on combined BAM file
sentieon driver -t NUMBER_THREADS -i SORTED_COMBINED_BAM_1 \
  --algo LocusCollector --fun score_info SCORE_TXT
sentieon driver -t NUMBER_THREADS -i SORTED_COMBINED_BAM_1 \
  --algo Dedup --rmdup --score_info SCORE_TXT --metrics DEDUP_METRIC_TXT DEDUPED_
BAM
```

6.1.2 处理来自不同样本的多个输入文件

共同处理来自一个群体的多个样本（也称为联合变异检测）可以通过两种方式完成：

- 单独处理每个样本，并使用 Haplotyper 算法的 --emit_mode gvcf 选项创建包含额外信息的 GVCF 文件，然后使用 GVCFtyper 算法处理所有 GVCF。这种方法允许在处理额外样本后轻松快速地重新处理。

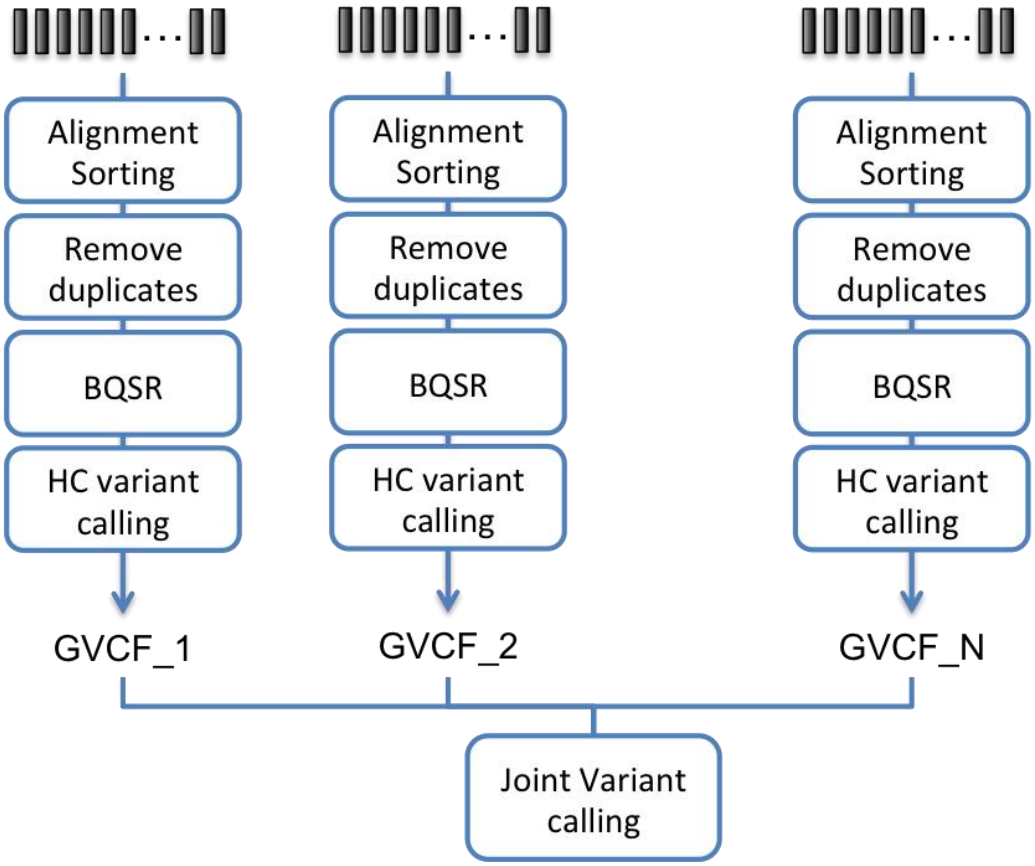

图 6-2 使用 GVCF 输出模式的 Haplotyper 和 GVCFtyper 进行联合调用的生物信息学流程

```
#Process all samples independently until GVCF
for sample in s1 s2 s3; do
  (sentieon bwa mem -R '@RG\tID:GROUP_${sample}\tSM:${sample}\tPL:PLATFORM' \
    -t NUMBER_THREADS REFERENCE ${sample}_FASTQ_1 ${sample}_FASTQ_2 || echo -n
 \
    'error') | sentieon util sort -o ${sample}_SORTED_BAM -t NUMBER_THREADS \
    --sam2bam -i -
  sentieon driver -t NUMBER_THREADS -i ${sample}_SORTED_BAM \
    --algo LocusCollector --fun score_info ${sample}_SCORE_TXT
  sentieon driver -t NUMBER_THREADS -i SORTED_BAM_1 -i SORTED_BAM_2 \
    --algo Dedup --rmdup --score_info ${sample}_SCORE_TXT ${sample}_DEDUPED_BAM
  sentieon driver -r REFERENCE -t NUMBER_THREADS -i ${sample}_DEDUPED_BAM \
    --algo QualCal -k $known_sites ${sample}_RECAL_DATA_TABLE
  sentieon driver -r REFERENCE -t NUMBER_THREADS -i ${sample}_DEDUPED_BAM \
    -q ${sample}_RECAL_DATA_TABLE --algo Haplotyper --emit_mode gvcf \
    ${sample}_VARIANT_GVCF
```

```
done
#Run joint variant calling
sentieon driver -r REFERENCE --algo GVCFtyper -v s1_VARIANT_GVCF \
  -v s2_VARIANT_GVCF -v s3_VARIANT_GVCF VARIANT_VCF
```

- 单独处理每个样本并为每个样本创建重校准的 BAM 文件或去重的 BAM 和重校准表，然后使用 Haplotyper 算法处理所有文件。

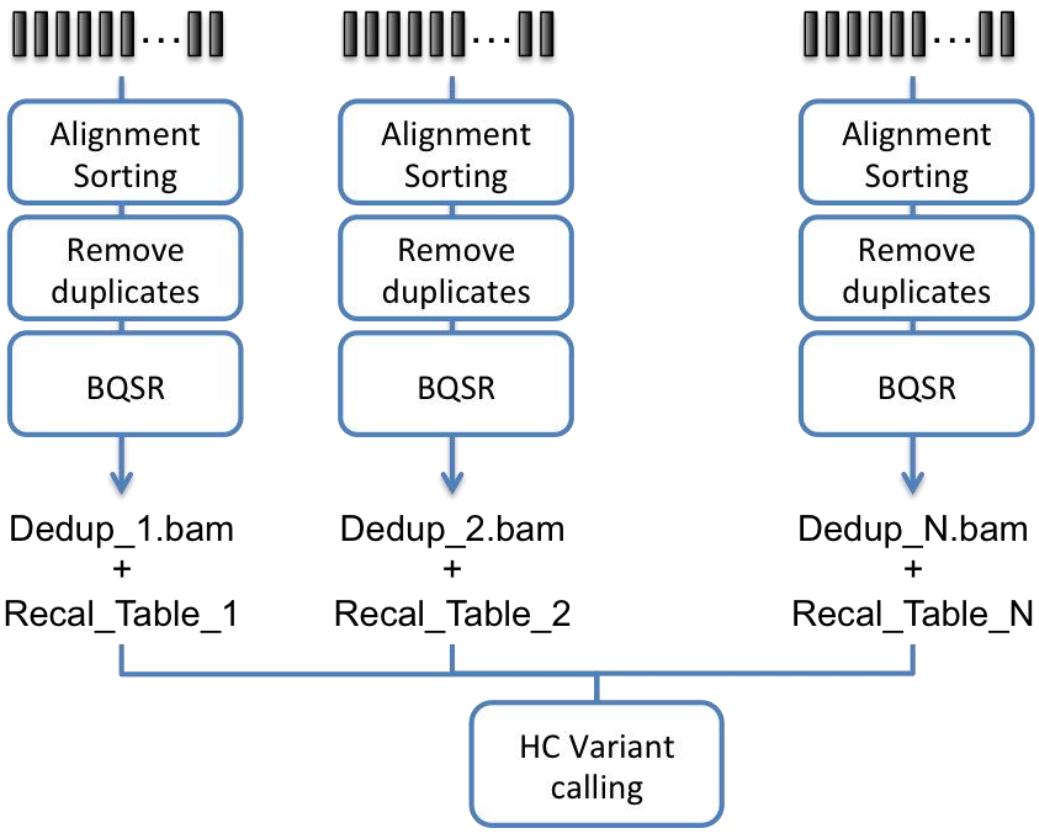

图 6-3 使用 Haplotyper 和多个 BAM 文件进行联合调用的生物信息学流程

```
#Process all samples independently until BQSR
for sample in s1 s2 s3; do
  (sentieon bwa mem -R '@RG\tID:GROUP_${sample}\tSM:${sample}\tPL:PLATFORM' \
    -t NUMBER_THREADS REFERENCE ${sample}_FASTQ_1 ${sample}_FASTQ_2 || echo -n
 \
    'error' )| sentieon util sort -o ${sample}_SORTED_BAM -t NUMBER_THREADS \
    --sam2bam -i -
  sentieon driver -t NUMBER_THREADS -i ${sample}_SORTED_BAM \
    --algo LocusCollector --fun score_info ${sample}_SCORE_TXT
```

```
  sentieon driver -t NUMBER_THREADS -i SORTED_BAM_1 -i SORTED_BAM_2 \
    --algo Dedup --rmdup --score_info ${sample}_SCORE_TXT ${sample}_DEDUPED_BAM
  sentieon driver -r REFERENCE -t NUMBER_THREADS -i ${sample}_DEDUPED_BAM \
    --algo QualCal -k $known_sites ${sample}_RECAL_DATA_TABLE
done
#Run joint variant calling
sentieon driver -t NUMBER_THREADS -i s1_DEDUPED_BAM -q s1_RECAL_DATA_TABLE \
  -i s2_DEDUPED_BAM -q s2_RECAL_DATA_TABLE -i s3_DEDUPED_BAM \
  -q s3_RECAL_DATA_TABLE --algo Haplotyper VARIANT_VCF
```

6.2 Haplotyper 和 Genotyper 中支持的注释

Haplotyper 和 Genotyper 这两种胚系变异调用方法都接受输出额外注释到结果 VCF 的选项。以下是支持的可能注释：

- **AC**: 基因型中的等位基因计数，针对每个 ALT 等位基因，按列出的顺序
- **AF**: 等位基因频率，针对每个 ALT 等位基因，按列出的顺序
- **AN**: 在调用的基因型中的总等位基因数
- **BaseQRankSum**: Alt 与 Ref 碱基质量的 Wilcoxon 秩和检验的 Z 分数
- **ClippingRankSum**: Alt 与 Ref 硬裁剪碱基数量的 Wilcoxon 秩和检验的 Z 分数
- **DP**: 跨样本的组合深度
- **ExcessHet**: 过度杂合性精确检验的 Phred 标度 p 值
- **FS**: 使用 Fisher 精确检验检测链偏好的 Phred 标度 p 值
- **InbreedingCoeff**: 与 Hardy-Weinberg 期望相比，根据每个样本的基因型似然估计的近交系数
- **MLEAC**: 等位基因计数的最大似然期望 （MLE），针对每个 ALT 等位基因，按列出的顺序
- **MLEAF**: 等位基因频率的最大似然期望 （MLE），针对每个 ALT 等位基因，按列出的顺序
- **MQ**: RMS 映射质量
- **MQ0**: MAPQ == 0 的读数数量
- **MQRankSum**: Alt 与 Ref 读数映射质量的 Wilcoxon 秩和检验的 Z 分数

- **QD**: 变异置信度/深度质量
- **RAW_MQ**: RMS 映射质量的原始数据
- **ReadPosRankSum**: Alt 与 Ref 读数位置偏好的 Wilcoxon 秩和检验的 Z 分数
- **SOR**: 检测链偏好的 2x2 列联表的对称比值比
- **SAC**: 支持每个等位基因（包括参考）的正向和反向链上的读数数量
- **AS_BaseQRankSum**: 等位基因特异性
- **AS_FS**: 等位基因特异性使用 Fisher 精确检验检测链偏好的 Phred 标度 p 值
- **AS_InbreedingCoeff**: 与 Hardy-Weinberg 期望相比，根据每个样本的基因型似然估计的等位基因特异性近交系数
- **AS_MQRankSum**: 等位基因特异性 Alt 与 Ref 读数映射质量的 Wilcoxon 秩和检验的 Z 分数
- **AS_QD**: 等位基因特异性变异置信度/深度质量
- **AS_MQ**: 等位基因特异性 RMS 映射质量
- **AS_ReadPosRankSum**: 等位基因特异性 Alt 与 Ref 读数位置偏好的 Wilcoxon 秩和检验的 Z 分数
- **AS_SOR**: 等位基因特异性检测链偏好的 2x2 列联表的对称比值比

6.3 比对后移除低映射质量的读数

以下是比对后如何移除低映射质量的读数的示例，以便它们不会占用输出 BAM 文件的空间。

```
sentieon bwa mem -R '@RG\tID:GROUP_NAME_1 \tSM:SAMPLE_NAME \tPL:PLATFORM' \
  -t NUMBER_THREADS REFERENCE SAMPLE_1 [SAMPLE_2] \
  |samtools view -h -q MAP_QUALITY_THRESHOLD - \
  | sentieon util sort -o SORTED_BAM_1 -t NUMBER_THREADS --sam2bam -i -
```

6.4 执行 Dedup 以标记主要和非主要读数

标准的双通道 dedup 将忽略非主要读数，并且永远不会改变它们的重复标志。

以下是如何执行 dedup 以同时标记主要和非主要读数的示例：

```
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo LocusCollector --fun score_info SCORE_TXT
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup --score_info SCORE_TXT --metrics DEDUP_METRIC_TXT \
  -- output_dup_read_name DUPLICATED_READNAMES_TXT
sentieon driver -t NUMBER_THREADS -i SORTED_BAM \
  --algo Dedup --rmdup --dup_read_name DUPLICATED_READNAMES_TXT DEDUPED_BAM
```

6.5 使用量化质量分数数据时的流程修改

当您的输入数据具有量化的碱基质量分数时，输入 FASTQ 中报告的碱基质量很可能比经验碱基质量高得多；在这种情况下，变异调用步骤可能需要更长时间来处理数据，因为在局部组装中本不会考虑的低质量碱基将导致生成许多不相关的低质量单倍型。这个问题应该通过运行流程的碱基质量分数重校准步骤来解决；或者，您可以修改 Haplotyper 或 TNhaplotyper 的 min_base_qual 参数的默认值以减少影响：在命令行中添加 --min_base_qual 15 选项以防止此问题。

6.6 当肿瘤和正常输入都具有相同的 RGID 时修改 BAM 文件的 RG 信息

如果您输入多个不同的 BAM 文件，其中包含具有相同 ID 但不同属性的读组，Sentieon® 工具将报错。这种情况可能发生在 TNseq® 和 TNscope® 中，当肿瘤和正常样本 BAM 文件的 RG ID 都是 "1" 时。为了解决这个问题，您可以使用该功能来修改 RG 信息。

```
#first lane tumor, RGID incorrectly set to uninformative 1
sentieon bwa mem -R '@RG\tID:1\tSM:TUMOR_SM\tPL:PLATFORM' -t NT \
  REFERENCE TUMOR_1_1.fq.gz TUMOR_1_2.fq.gz | sentieon util sort \
  -o TUMOR_1.bam -t NT --sam2bam -i -

#second lane tumor, RGID incorrectly set to uninformative 1
sentieon bwa mem -R '@RG\tID:1\tSM:TUMOR_SM\tPL:PLATFORM' -t NT \
  REFERENCE TUMOR_2_1.fq.gz TUMOR_2_2.fq.gz | sentieon util sort \
  -o TUMOR_2.bam -t NT --sam2bam -i -

#normal, RGID incorrectly set to uninformative 1
sentieon bwa mem -R '@RG\tID:1\tSM:NORMAL_SM\tPL:PLATFORM' -t NT \
  REFERENCE TUMOR_1.fq.gz TUMOR_2.fq.gz | sentieon util sort \
  -o NORMAL.bam -t NT --sam2bam -i -

#using replace_rg argument to modify the RGID and make them unique
#each replace_RG argument will only affect the following input BAM file
sentieon driver -r REFERENCE \
  --replace_rg 1='ID:T_1\tSM:TUMOR\tPL:PLATFORM' -i TUMOR_1.bam \
  --replace_rg 1='ID:T_2\tSM:TUMOR\tPL:PLATFORM' -i TUMOR_2.bam \
  --replace_rg 1='ID:N\tSM:NORMAL\tPL:PLATFORM'-i NORMAL.bam \
  --algo TNscope --tumor_sample TUMOR --normal_sample NORMAL OUT_TN_VCF
```

7 故障排除与常见问题

7.1 故障排除

7.1.1 准备参考文件以供使用

如果您的参考 FASTA 文件尚未预处理，使得软件无法使用中指定的数据，您需要按照以下步骤对其进行处理：

1. 使用 BWA 生成 BWA 索引。这将创建".fasta.amb"、".fasta.ann"、".fasta.bwt"、".fasta.pac"和".fasta.sa"文件。

```
sentieon bwa index reference.fasta
```

2. 使用 samtools 生成 FASTA 文件索引。这将创建".fasta.fai"文件。

```
samtools faidx reference.fasta
```

3. 使用 Picard 生成序列字典。这将创建".dict"文件。

```
java -jar picard.jar CreateSequenceDictionary REFERENCE=reference.fasta \
  OUTPUT=reference.dict
```

7.1.2 准备 RefSeq 文件以供使用

RefSeq 文件用于将 CoverageMetrics 算法的结果聚合到基因级别。为了使用从 UCSC 基因组浏览器下载的 RefSeq 文件，需要按染色体和位点对它们进行排序。要执行排序，您需要执行以下步骤：

1. 从文件中删除头部

```
grep -v “^#” FILE.refSeq > FILE.refSeq.headerless
grep -e “^#” FILE.refSeq > FILE.refSeq.header
```

2. 首先使用 unix sort 对位点进行排序。

```
sort -k 5 -n FILE.refSeq.headerless > FILE.refSeq.presorted
```

3. 使用 GATK sortByRef.pl（可从获取）和 FASTA 索引 fai 按染色体排序。

```
perl sortByRef.pl --k 3 FILE.refSeq.presorted FASTA.fai --tmp ~/tmp \
    > FILE_sorted_headerless.refSeq
```

4. 将头部放回文件。

```
cat FILE.refSeq.header > FILE_sorted.refSeq
cat FILE_sorted_headerless.refSeq >> FILE_sorted.refSeq
```

7.2 常见问题

7.2.1 Sentieon 安装时出现 jemalloc error

在安装运行 Sentieon 过程中可能会出现以下错误：

```
ERROR: ld.so: object '/usr/lib64/libjemalloc.so.2' from LD_PRELOAD cannot be preloaded: ignored. Failed to contact the license server at 10.10.10.1:8990
```

这个错误与 jemalloc 有关。Jemalloc 是一个内存分配器，针对多线程方案中的高内存分配性能和更少的内存碎片进行了优化。Sentieon 建议使用 jemalloc 来改善 Sentieon 应用程序中的内存管理和整体性能，尤其是 Sentieon bwa-mem。

解决方案：

1. 安装 jemalloc：

对于不同的操作系统，安装命令如下：

- RHEL/CentOS 8.x:

```
yum install epel-release
yum install jemalloc
```

默认安装在 /usr/lib64/libjemalloc.so.2

- RHEL/CentOS 7.x:

```
yum install epel-release
yum install jemalloc
```

默认安装在 /usr/lib64/libjemalloc.so.1

- Ubuntu 20.04:

```
apt update
apt install libjemalloc2
```

默认安装在 /usr/lib/x86_64-linux-gnu/libjemalloc.so.2

- Ubuntu 18.04:

```
apt update
apt install libjemalloc1
```

默认安装在 /usr/lib/x86_64-linux-gnu/libjemalloc.so.1

2. 对于没有预构建软件包的其他系统，请参考 jemalloc GitHub 页面（*https：//github.com/jemalloc/jemalloc*）以获取有关如何构建和安装 jemalloc 的更多信息。

3. 使用环境变量在运行时加载 jemalloc 库到 Sentieon 中：

例如，在 CentOS 8.x 系统上，在运行 Sentieon 工具之前，您可以使用以下命令设置环境变量：

```
export LD_PRELOAD=/usr/lib64/libjemalloc.so.2
```

7.2.2 许可证消息：No more license available for Sentieon...

当您请求运行 Sentieon®软件的线程数超过您的许可证当前允许的数量时，会产生此消息。这种情况发生是因为您同时运行的命令集体请求的线程数超过了您的许可证支持的核心数。可使用以下命令查看授权剩余的线程数：

```
sentieon licclnt query -s LICSRVR_HOST:LICSRVR_PORT klib
```

Sentieon®命令在等待空闲许可证时将处于空闲状态，但命令不会失败。

7.2.3 Driver 失败并显示错误：Readgroup XX is present in multiple BAM files with different attributes

当您输入两个不同的 BAM 文件，其中包含具有相同 ID 但不同属性的读组时，会产生此错误。例如，在 TNseq®和 TNscope®中，肿瘤和正常样本 BAM 文件的 RG ID 都是"1"。

在使用 BAM 文件之前，您需要编辑它们以使 RG ID 唯一，例如通过将 SM 名称

添加到 RG ID 中。您可以查看此问题的解决方法示例。

或者，您可以使用 samtools addreplacerg 功能来修改输入 BAM 文件的 RG ID 并使其唯一：

```
#add the new RG and modify all reads in the BAM file
RGtag=$(samtools view -H $INPUT_BAM|grep ^@RG|sed "s|ID:$ORIG_RGID|ID:$NEW_RGID|g")
samtools addreplacerg -r "$RGtag" -o $TMP_BAM $INPUT_BAM
#reheader the BAM to remove the original RG that is no longer used
samtools view -H $TMP_BAM|grep -v "^@RG.*$ORIGINAL_RGID" \
    |samtools reheader - $TMP_BAM > $OUTPUT_BAM
rm $TMP_BAM
```

7.2.4 Driver 报告警告：none of the QualCal tables is applicable to the input BAM files

此警告意味着重校准表输入文件中的信息都不能应用于输入的 BAM 文件，这可能是由于使用了与 BAM 文件不对应的重校准表。

当 QualCal 的输入 BAM 文件在 RG 字段中没有正确的字段时，可能会产生此警告。例如，如果 RG 的 PL 标签设置为 ILLUMINA 以外的其他值，就可能发生这种情况；在这种情况下，您需要修改 BAM 头以包含/修改缺失/不正确的字段，为此您可以使用 samtools reheader 功能。

7.2.5 使用从 BAM 文件创建的 FASTQ 文件时，BWA 使用异常大量的内存

当您使用通过转换已排序的 BAM 文件创建的 FASTQ 文件时，可能会发生所有未映射的读段都被分组到 FASTQ 输入的末尾。在这种情况下，BWA 可能在对

齐结束时使用异常大量的内存，因为映射质量差或无法映射的读段需要额外的内存。

为了减少异常内存使用，您应该首先重新排序 bam 文件，以确保未映射的读段不会被分组在一起。您可以使用 samtools 来做到这一点：

```
samtools sort -n -@ 32 input.bam | samtools fastq -@ 32 \
-s >(gzip -c > single.fastq.gz) -0 >(gzip -c > unpaired.fastq.gz) \
-1 >(gzip -c > output_1.fastq.gz) -2 >(gzip -c > output_2.fastq.gz) -
```

7.2.6 Driver 或 Util 失败并显示错误：can not open file （xxx) in mode（r），Too many open files

此错误的根本原因是系统中同时打开的文件限制设置得不够高。

您可以通过设置系统 ulimit -n 来解决此错误。在基于 Linux 的系统中：

检查系统中最大打开文件数的限制，运行以下命令：

```
ulimit -n
```

通过以 root 身份编辑文件/etc/security/limits.conf 来设置更高的限制，并添加以下 2 行：

```
* soft nofile 16384
* hard nofile 16384
```

如果您的系统运行的是 Ubuntu，您还需要将此行添加到您的 shell 配置文件 ~/.bashrc 中：

```
ulimit -n 16384
```

您需要注销系统并重新登录才能使更改生效。登录后，通过运行以下命令检查更改是否正确应用：

```
ulimit -n
```

命令应返回 16384。

7.2.7 Driver 失败并显示错误：Contig XXX from vcf/bam is not present in the reference， or error Contig XXX has different size in vcf/bam than in the reference

此错误的根本原因是输入的 VCF 或 BAM 文件与参考 fasta 文件不兼容。文件中的 contig 不存在于参考中，或 contig 的大小不同。这很可能是由于使用了与不同参考处理的 VCF 或 BAM 文件造成的。

7.2.8 Driver 报告警告：Contigs in the vcf file XXX do not match any contigs in the reference

此警告的根本原因是输入的 VCF 文件与参考 fasta 文件不兼容，文件中的 contig 不存在于参考中。这很可能是由于使用了来自不同参考的 VCF 文件造成的。

7.2.9 BWA 失败并显示错误：Killed

当 BWA 从操作系统接收到 SIGKILL 信号时，会产生此错误。如果系统可用内存不足，SIGKILL 可能是由内核的内存不足（OOM）管理器发送的。您可以检查系统上的内核日志以确认 SIGKILL 信号是否由 OOM 管理器发送。

要解决此错误，您可以使用 bwt_max_mem 环境变量减少 BWA 的内存使用。

7.3 KPNS - 已知问题无解决方案

7.3.1 不支持未使用 bgzip 压缩的 gzip 压缩 vcf 文件

普通的 gzip 文件不允许随机或索引访问其中包含的信息，只有使用 bgzip 压缩的文件才可索引。

因此，Sentieon®软件不支持 gzip 压缩的 VCF 文件作为输入。

要使用这些文件，您需要使用 gunzip 解压缩它们，然后不压缩使用或使用 bgzip 重新压缩。

或者，您可以使用 util vcfconvert 重新压缩和索引文件。

```
sentieon util vcfconvert INPUT.vcf.gz OUTPUT.vcf.gz
```

7.3.2 不支持 gzip 压缩的 fasta 文件

目前，软件不支持 gzip 压缩的 FASTA 文件作为输入。您需要在使用之前 gunzip 这些文件。

7.3.3 FASTQ 文件需要采用 SANGER 质量格式

如果您的 FASTQ 文件是在 1.8 之前使用 Illumina™测序技术编码的，读段质量分数将不是 SANGER 格式，这可能会产生意外结果。Sentieon®基因组学软件不会检测到您正在使用不支持的格式。

7.3.4 Driver 失败并显示错误： ImportError： No module named argparse

当在 Python 版本为 2.6.x 且不存在 argparse 模块的环境中运行 tnhapfilter 时，

会产生此错误。您需要为您的 Python 安装安装 argparse 模块；您可以通过运行 pip install argparse 或您使用的其他包管理器来完成此操作。

8 附录

8.1 缩写和缩略语

GATK：基因组分析工具包（Genome Analysis Tool Kit）

VCF：变异检测格式（Variant Call Format）

SAM：序列比对/映射格式（Sequence Alignment/Map）

BAM：二进制压缩的 SAM 格式（Binary compressed SAM）

BCF：二进制压缩的变异检测格式（Binary compressed Variant Call Format）

BED：浏览器可扩展显示格式（Browser Extensible Display）

Indels：插入-缺失（INsertion-DELetions）

8.2 免责声明和责任限制

SENTIEON 公司保留程序交付时的所有权利。未经 SENTIEON 同意，除许可证规定外，不得以任何形式复制程序或其任何部分。

不得从程序中删除本声明。

SENTIEON、SENTIEON 的任何成员、以下组织或代表他们行事的任何人：

1. 不对程序作出任何明示或暗示的保证或陈述，包括对适销性或特定用途适用性的任何保证；

2. 不对程序或其任何部分的任何使用或可能由此产生的任何损害承担任何责任。

在法律允许的最大范围内，除本文明确保证外，SENTIEON 提供的软件、文档和其他产品及服务均按"原样"提供，不附带任何种类的明示或暗示保证，包括但不限于所有权保证或对适销性、非侵权或特定用途适用性的暗示保证。SENTIEON 及其许可方不保证 SENTIEON 软件、文档或服务将满足被许可方的要求，无错误或不间断运行。

www.insvast.com

上海毅硕信息技术有限公司