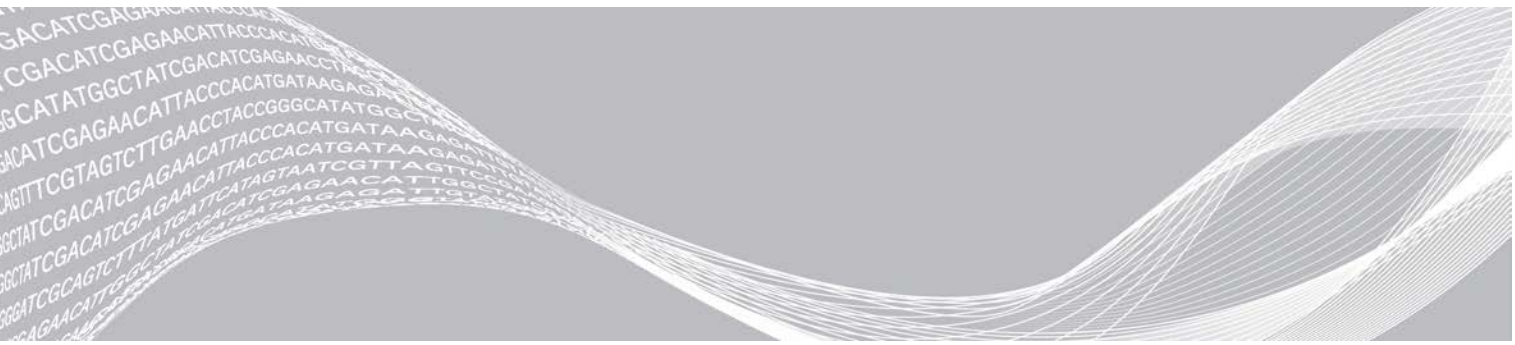
illumina®

Illumina DRAGEN Bio-IT-Plattform v3.4

Benutzerhandbuch



Dokument-Nr. 1000000112459 v00 DEU

Oktober 2019

Nur für Forschungszwecke. Nicht zur Verwendung in Diagnoseverfahren.

Versionshistorie

Dokument	Datum	Beschreibung der Änderung
Dokumentnr. 1000000112459 v00	Januar 2020	• Neue Dokumentnummer erstellt. • 3.4 zum Titel hinzugefügt.
Dokumentnr. 1000000070494 v06	Oktober 2019	• Optionsname (--qc-coverage-reports) korrigiert. • Fehlende Abbildung hinzugefügt.
Dokumentnr. 1000000070494 v05	August 2019	Pipelineinformationen in der Einführung aktualisiert. Folgende Abschnitte wurden hinzugefügt: • Downsampling-Optionen für das Calling kleiner mitochondrialer Varianten • De-novo-Variantenfilterung • Optionen für die De-novo-Variantenfilterung • Manta Structural Variant Caller • Anwendungsfälle für Coverage-Berichte und erwartete Ausgabe • Ploidie-Unterstützung • QUAL, QD und GQ • ROH-Caller • Kompatibilität mit Cufflinks Folgende Abschnitte wurden aktualisiert: • gVCF, Combine gVCF und Joint VCF • De-novo-Qualität struktureller Varianten • Qualitätssicherungsmetriken und Berichte zur Coverage/Callfähigkeit • Mapping- und Alignment-Metriken • Mapping- und Alignment-Metriken im somatischen Modus • Metriken für das Varianten-Calling • Berichte zur Coverage/Callfähigkeit • Unique Molecular Identifiers • RNA-Pipeline • RNA-Alignment • Konvertieren von Illumina-BCL-Daten Folgende Optionen wurden hinzugefügt: • --sample-sex • --vc-target-bed-padding • --umi-min-supporting-reads • --vc-clustered-events-threshold • --vc-enable-roh • --vc-roh-blacklist-bed • --vc-enable-baf Folgende Optionen wurden entfernt: • --vc-sample-name • --cnv-enable-split-intervals „--bcl-output-dir“ in „--bcl-output-directory“ geändert. „--cnv-wgs-interval-width“ in „--cnv-interval-width“ geändert. „--cnv-wgs-skip-contig-list“ in „--cnv-skip-contig-list“ geändert.
Dokumentnr. 1000000070494 v04	April 2019	Versionsverlauf für v03 hinzugefügt, Telefonnummer des technischen Supports für Südkorea hinzugefügt.

Dokument	Datum	Beschreibung der Änderung
Dokumentnr. 1000000070494 v03	April 2019	Verweis auf die obsolet gewordene Option „--cram-reference“ entfernt. Option „--annotation-sj-file“ entfernt. Hinweis auf die neue Option „--annotation-file“. Informationen zu folgenden Optionen wurden hinzugefügt: • --vc-enable-phasing • --vc-tlod-filter-threshold • --vc-enable-triallelic-filter • --cnv-merge-distance • --cnv-enable-tracks Folgende Abschnitte wurden hinzugefügt: • Downsampling-Optionen für das Calling kleiner Keimbahn-Varianten • Phasierung und phasierte Varianten • SMA-Calling • De-novo-Qualitätsbewertung • Mehrproben-CNV-Calling • Gemeinsame Segmentierung • De-novo-Calling-Phase • Mehrproben-CNV-VCF-Ausgabe • MapQ- und BQ-Coverage-Filterung • Unique Molecular Identifiers (UMIs) • Genquantifizierung Folgende Abschnitte wurden aktualisiert: • Mitochondrien-Calling • De-novo-Joint-Calling • Harte Variantenfilterung • Repeat-Genotypisierung • Calling struktureller Varianten • gVCF, Combine gVCF und Joint VCF • Qualitätsbewertung • Mapping und Alignierung
Dokumentnr. 1000000070494 v02	März 2019	Folgende Anwendungsnamen wurden aktualisiert: • DRAGEN-Genom-Pipeline heißt nun DRAGEN-DNA-Anwendungen. • DRAGEN-Transkriptom-Pipeline heißt nun DRAGEN-RNA-Anwendungen. • DRAGEN-Epigenom-Pipeline heißt nun DRAGEN-Methylierungsanwendungen. Repeat-Genotypisierung heißt nun Repeat-Expansion-Bestimmung.
Dokumentnr. 1000000070494 v01	Januar 2019	Für Softwareversion 3.2.5 aktualisiert. Abschnitt „Kopienzahlvarianten-Calling“ hinzugefügt. Neue Optionen zu Anhang A hinzugefügt.
Dokumentnr. 1000000070494 v00	Dezember 2018	Erste Version.

Inhaltsverzeichnis

Kapitel 1 Illumina DRAGEN Bio-IT-Plattform

Die Illumina DRAGEN™ Bio-IT-Plattform basiert auf dem umfassend rekonfigurierbaren DRAGEN Bio-IT-Prozessor, der auf einer Field Programmable Gate Array(FPGA)-Karte in einem vorkonfigurierten Server zur Verfügung steht, der sich nahtlos in Bioinformatik-Workflows integrieren lässt. Die Plattform kann hochgradig optimierte Algorithmen für viele verschiedene NGS-Sekundäranalyse-Pipelines ausführen, darunter:

- Gesamtgenom
- Exom
- RNA-Sequenzierung
- Methylom
- Krebs

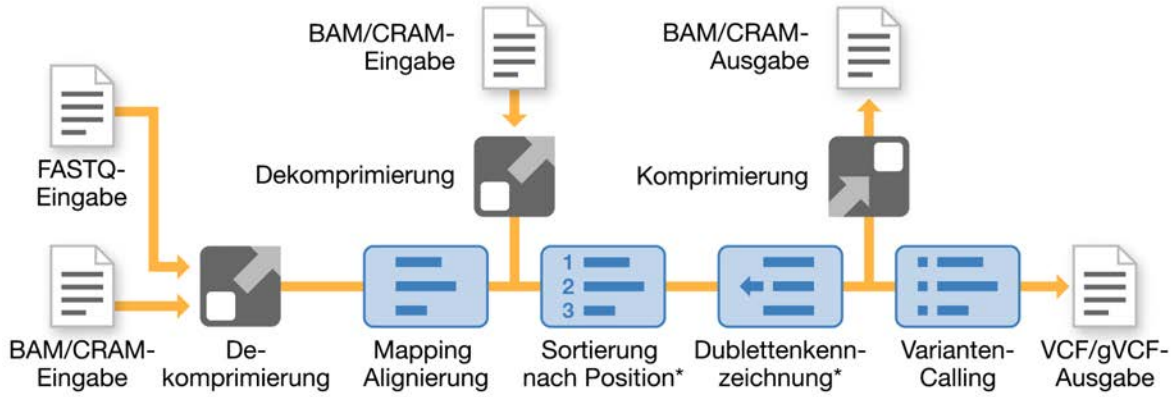
Die komplette Bedienung erfolgt über die DRAGEN-Software, die auf dem Hostserver ausgeführt wird und die gesamte Kommunikation mit dem DRAGEN-Board übernimmt.

Das vorliegende Benutzerhandbuch erläutert die technischen Aspekte und enthält ausführliche Informationen zu allen Befehlszeilenoptionen für DRAGEN.

Sollten Sie zum ersten Mal mit DRAGEN arbeiten, empfiehlt Illumina, dass Sie sich zunächst mit der *Kurzanleitung zur Illumina DRAGEN Bio-IT-Plattform* (1000000076675) vertraut machen, die auf der Illumina-Supportseite zum Download bereitsteht. Dieses Dokument enthält eine kurze Einführung zu DRAGEN, in der das Ausführen eines Servertests, die Generierung eines Referenzgenoms und die Ausführung von Beispielbefehlen beschrieben werden.

DRAGEN-DNA-Pipeline

Abbildung 1 DRAGEN-DNA-Pipeline



* Optional

Die DRAGEN-DNA-Pipeline beschleunigt die Sekundäranalyse von NGS-Daten erheblich. Beispielsweise dauert die Verarbeitung eines kompletten Humangenoms bei 30-facher Coverage statt ca. 10 Stunden (mit dem derzeitigen Branchenstandard BWA-MEM+GATK-HC-Software) nur noch 20 Minuten. Der Zeitfaktor ist linear abhängig von der Coverage-Tiefe.

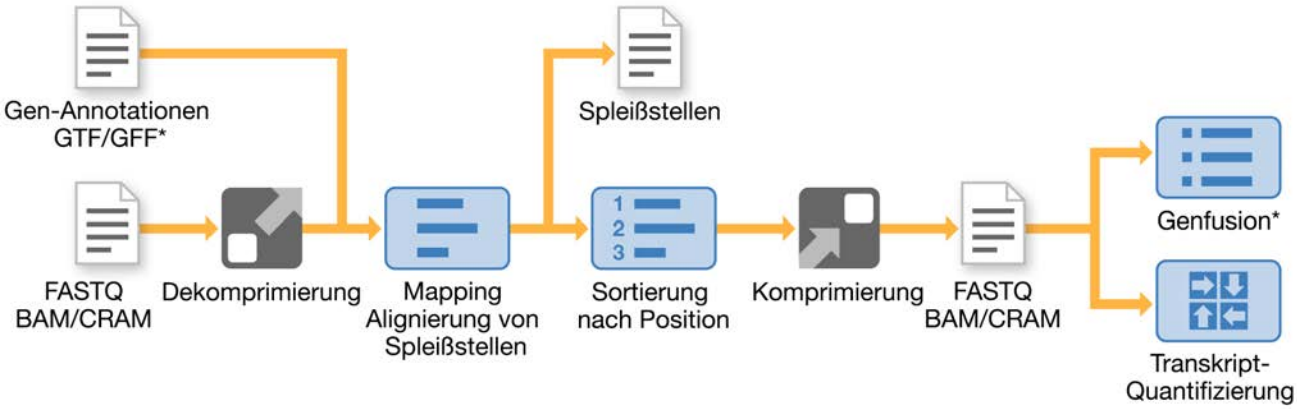
Diese Pipelines nutzen die enorme Leistung der DRAGEN Bio-It-Plattform und enthalten hochgradig optimierte Algorithmen für Mapping, Alignment, Sortierung, Dublettenkennzeichnung und Haplotyp-Varianten-Calling. Außerdem profitieren Sie von Merkmalen der Plattform wie Hardwarebeschleunigung und optimierter BCL-Konvertierung sowie von einem kompletten Satz an plattformeigenen Tools.

Im Gegensatz zu anderen Sekundäranalyseverfahren erfolgt bei DRAGEN-DNA-Anwendungen keine Beeinträchtigung der Genauigkeit zugunsten von höherer Geschwindigkeit. Die Genauigkeit von SNPs und INDELs ist im direkten Vergleich höher als bei BWA-MEM+GATK-HC.

Zusätzlich zum Haplotyp-Varianten-Calling ermöglicht die Pipeline das Calling von Kopienzahl- und strukturellen Varianten sowie die Bestimmung von Repeat-Expansionen.

DRAGEN-RNA-Pipeline

DRAGEN umfasst einen RNA-Seq-Aligner (spleiß-sensibel) sowie RNA-spezifische Analysekomponenten für die Genexpressionsquantifizierung und die Genfusionserkennung.



Die RNA-Pipeline und die DNA-Pipeline von DRAGEN nutzen viele Komponenten gemeinsam. Das Mapping von kurzen Seed-Sequenzen aus RNA-Seq-Reads ist vergleichbar mit dem Mapping von DNA-Reads. Zusätzlich werden Spleißstellen (Verbindungsstellen nicht benachbarter Exons in RNA-Transkripten) in der Nähe der gemappten Seeds erkannt und in die vollständigen Read-Alignments aufgenommen.

DRAGEN verwendet hardwarebeschleunigte Algorithmen, um RNA-Seq-basierte Reads schneller und genauer als gängige Software-Tools zu mappen und zu alignieren. DRAGEN kann beispielsweise 100 Millionen RNA-Seq-basierte Paired-End-Reads in ca. drei Minuten alignieren. Dank simulierter Benchmark-RNA-Seq-Datensätze ist die Spleißstellensensitivität und -spezifität einzigartig.

DRAGEN-Methylierungspipeline

Die DRAGEN-Methylierungspipeline bietet durch die Generierung einer BAM-Datei mit für die Methylierungsanalyse erforderlichen Tags Unterstützung bei der automatisierten Verarbeitung von Bisulfit-Sequenzierungsdaten.

Systemaktualisierungen

DRAGEN ist eine flexible und erweiterbare Plattform mit umfangreichen Konfigurationsmöglichkeiten. Mit Ihrem DRAGEN-Abonnement können Sie Aktualisierungen für die Prozessoren und Software von DRAGEN herunterladen. Diese Aktualisierungen verbessern die Geschwindigkeit, die Leistung, den Durchsatz und die Genauigkeit.

Weitere Ressourcen und Support

Weitere Informationen, Ressourcen, Systemupdates und Support finden Sie auf der DRAGEN-Supportseite auf der Illumina-Website.

Erste Schritte

Mit spezifischen Tests von DRAGEN können Sie überprüfen, ob das DRAGEN-System richtig installiert und konfiguriert wurde. Stellen Sie vor dem Ausführen der Tests sicher, dass der DRAGEN-Server über eine ausreichende Stromversorgung und Kühlung verfügt und an ein Netzwerk angeschlossen ist, in dem sich die Daten mit angemessener Geschwindigkeit vom Gerät übertragen lassen.

Systemprüfung durchführen

Nach dem Einschalten können Sie die korrekte Funktion Ihres DRAGEN-Servers überprüfen, indem Sie /opt/edico/self_test/self_test.sh ausführen. Die Systemprüfung umfasst folgende Schritte:

- Automatische Indizierung von Chromosom M aus dem Referenzgenom hg19
- Laden von Referenzgenom und -index
- Mapping und Alignment eines Read-Satzes
- Speichern der alignierten Reads in einer BAM-Datei
- Überprüfung der Alignments auf Übereinstimmung mit den erwarteten Ergebnissen

Die FASTQ-Testeingabedaten für dieses Skript sind im Lieferumfang des Servers enthalten. Der Speicherort lautet /opt/edico/self_test. Die Systemprüfung nimmt etwa 25 bis 30 Minuten in Anspruch.

Das folgende Beispiel zeigt den Ausgabetext nach erfolgreicher Ausführung des Skripts.

```
[root@edico2 ~]# /opt/edico/self_test/self_test.sh
   --------------------------------
   test hash creating
   test hash created
   --------------------------------
   reference loading /opt/edico/self_test/ref_data/chrM/hg19_chrM
   reference loaded
   --------------------------------

   real0m0.640s
   user0m0.047s
   sys0m0.604s
   not properly paired and unmapped input records percentages: PASS
   --------------------------------
   md5sum check dbam sorted: PASS
   --------------------------------
   SELF TEST COMPLETED
   SELF TEST RESULT : PASS
```

Sollte die ausgegebene BAM-Datei nicht mit den erwarteten Ergebnissen übereinstimmen, lautet die letzte Zeile des Ausgabetexts:

```
SELF TEST RESULT : FAIL
```

Sollten Sie nach dem Einschalten des DRAGEN-Servers und der Ausführung des Testskripts das Ergebnis FAIL erhalten, wenden Sie sich an den technischen Support von Illumina.

Ausführen eines eigenen Tests

Wenn Sie der Meinung sind, dass das DRAGEN-System wie vorgesehen funktioniert, können Sie wie folgt einen Test mit eigenen Daten durchführen:

- Referenztabelle für das Referenzgenom laden
- Speicherort für Ein- und Ausgabedateien festlegen
- Eingabedaten verarbeiten

Laden des Referenzgenoms

Vor der Verwendung eines Referenzgenoms mit DRAGEN muss dieses vom FASTA-Format in ein spezielles Binärformat für die DRAGEN-Hardware konvertiert werden. Weitere Informationen finden Sie unter *Vorbereiten eines Referenzgenoms* auf Seite 129.

Die in der Befehlszeile angegebene Hashtabelle wird automatisch auf das Board geladen, wenn Sie zum ersten Mal Daten mit einer Pipeline verarbeiten. Die Hashtabelle für das Referenzgenom kann mit dem folgenden Befehl manuell geladen werden:

```
dragen -r <Verzeichnis mit der Referenz-Hashtabelle>
```

Stellen Sie sicher, dass sich das Verzeichnis mit der Referenz-Hashtabelle auf dem schnellen EA-Dateilaufwerk befindet.

Der Standardspeicherort für die Hashtabelle für hg19 ist:

```
/staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149
```

Der Referenzgenom hg19 wird mit folgendem Befehl vom Standardspeicherort geladen:

```
dragen -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149
```

Mit diesem Befehl wird die Binärdatei mit dem Referenzgenom in den Arbeitsspeicher auf dem DRAGEN-Board geladen und dort für die Verarbeitung einer beliebigen Anzahl von Eingabedatensätzen verwendet. Das Referenzgenom muss nur bei einem Neustart des Systems und zum Wechsel des Referenzgenoms neu geladen werden. Das Laden des Referenzgenoms dauert bis zu einer Minute.

DRAGEN prüft, ob das angegebene Referenzgenom bereits auf dem Board vorhanden ist. Ist dies der Fall, wird der Upload des Referenzgenoms automatisch übersprungen. Das erneute Laden des Referenzgenoms kann mit der Befehlszeilenoption *force-load-reference (-l)* erzwungen werden.

Beim Ausführen des Befehls zum Laden des Referenzgenoms werden die Software- und die Hardwareversion über die Standardausgabe ausgegeben. Beispiel:

```
DRAGEN Host Software Version 01.001.035.01.00.30.6682 and
Bio-IT Processor Version 0x1001036
```

Nachdem das Referenzgenom geladen wurde, wird über die Standardausgabe folgende Meldung ausgegeben:

```
DRAGEN finished normally
```

Festlegen des Speicherorts für Ein- und Ausgabedateien

Die DRAGEN Bio-IT-Plattform ist extrem schnell, was eine besondere Beachtung des Speicherorts von Ein- und Ausgabedateien erfordert. Wenn sich die Ein- oder Ausgabedateien auf einem langsamen Dateisystem befinden, schränkt dessen Durchsatz die Leistung des Systems ein.

Das DRAGEN-System ist mit mindestens einem schnellen Dateisystem vorkonfiguriert, das aus schnellen SSDs besteht, die zur Steigerung der Leistung mit RAID-0 zusammengefasst werden. Das Dateisystem ist unter /staging gemounted. Hierbei handelt es sich um einen großen und schnellen Speicherbereich, der jedoch keine Redundanz bietet. Der Ausfall eines der Datenträger im Dateisystem führt zum Verlust sämtlicher gespeicherter Daten.

Vor dem Start einer Analyse wird Folgendes empfohlen:

- Kopieren Sie die Eingabedaten nach /staging.
- Legen Sie die Ausgabe auf /staging fest.
- Speichern Sie eine Kopie der Staging-Eingabedaten an einem anderen Speicherort.

Verarbeiten von Eingabedaten

Verwenden Sie zur Analyse der FASTQ-Daten den Befehl *dragen*. Mit folgendem Befehl können Sie beispielsweise eine Single-End-FASTQ-Datei analysieren:

```
dragen \
   -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149 \
   -1 /staging/test/data/SRA056922.fastq \
   --output-directory /staging/test/output \
   --output-file-prefix SRA056922_dragen \
   --RGID DRAGEN_RGID \
   --RGSM DRAGEM_RGSM
```

Weitere Informationen zu den Befehlszeilenoptionen finden Sie unter *DRAGEN-Hostsoftware* auf Seite 6

Kapitel 2 DRAGEN-Hostsoftware

Mit dem DRAGEN-Hostsoftwareprogramm *dragen* können Sie Referenzgenome erstellen und laden und anschließend die Sequenzierungsdaten durch Dekomprimieren der Daten, Mapping, Alignieren, Sortieren, Dublettenkennzeichnung mit optionalem Entfernen und Varianten-Calling analysieren.

Rufen Sie die Software mithilfe des Befehls *dragen* auf. Die Befehlszeilenoptionen werden in den folgenden Abschnitten beschrieben.

Befehlszeilenoptionen können außerdem in einer Konfigurationsdatei festgelegt werden. Weitere Informationen zu Konfigurationsdateien finden Sie unter *Konfigurationsdateien* auf Seite 15. Wenn eine Option in der Konfigurationsdatei festgelegt ist und außerdem in der Befehlszeile angegeben wird, überschreibt die Befehlszeilenoption die Konfigurationsdatei.

Befehlszeilenoptionen

Im Folgenden finden Sie eine Zusammenfassung zur Verwendung des Befehls *dragen*:

- Referenz/Hashtabelle erstellen

```
dragen --build-hash-table true --ht-reference <REFERENZ-FASTA> \
   --output-directory <REFERENZVERZEICHNIS> [Optionen]
```

- Mapping/Alignment und Varianten-Caller ausführen *(*.fastq zu *.vcf)*

```
dragen -r <REFERENZVERZEICHNIS> --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <DATEIPRÄFIX> [Optionen] -1 <FASTQ1> \
   [-2 <FASTQ2>] --RGID <RG0> --RGSM <SM0> --enable-variant-caller true
```

- Mapping/Alignment ausführen *(*.fastq zu *.bam)*

```
dragen -r <REFERENZVERZEICHNIS> --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <DATEIPRÄFIX> [Optionen] \
   -1 <FASTQ1> [-2 <FASTQ2>] \
   --RGID <RG0> --RGSM
```

- Varianten-Caller ausführen *(*.bam zu *.vcf)*

```
dragen -r <REFERENZVERZEICHNIS> --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <DATEIPRÄFIX> [Optionen] -b <BAM> \
   --enable-variant-caller true
```

- BCL-Konverter ausführen *(BCL zu *.fastq)*

```
dragen --bcl-conversion-only true --bcl-input-directory <BCL-VERZEICHNIS>
   \
   --output-directory <AUSGABEVERZEICHNIS>
```

- RNA-Mapping/Alignment ausführen *(*.fastq zu *.bam)*

```
dragen -r <REFERENZVERZEICHNIS> --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <DATEIPRÄFIX> [Optionen] -1 <FASTQ1> \
   [-2 <FASTQ2>] --enable-rna true
```

Eine vollständige Liste der Befehlszeilenoptionen finden Sie unter *Befehlszeilenoptionen* auf Seite 148.

Referenzgenomoptionen

Bevor Sie das DRAGEN-System für das Alignment von Reads verwenden können, müssen Sie ein Referenzgenom und die zugehörigen Hashtabellen auf die PCIe-Karte laden. Informationen zur vorbereitenden Konvertierung der FASTA-Dateien des Referenzgenoms in eine native DRAGEN-Binärreferenz

sowie zum Erstellen von Hashtabellen finden Sie unter *Vorbereiten eines Referenzgenoms* auf Seite 129. Zusätzlich müssen Sie mit der Option *-r* [oder *--ref-dir*] das Verzeichnis mit der vorbereiteten Binärreferenz sowie den Hashtabellen angeben. Dieses Argument ist immer erforderlich.

Sie können das Referenzgenom und die Hashtabellen getrennt vor der Read-Verarbeitung wie folgt in den Arbeitsspeicher der DRAGEN-Karte laden.

```
dragen -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149
```

Mit der Option *-l (--force-load-reference)* können Sie das Laden des Referenzgenoms wie folgt erzwingen, selbst wenn dieses bereits geladen wurde.

```
dragen -l -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149
```

Wie lange das Laden des Referenzgenoms dauert, hängt von der Größe der jeweiligen Referenz ab. Mit normalen empfohlenen Einstellungen dauert das Laden in der Regel zwischen 30 und 60 Sekunden.

Betriebsmodi

DRAGEN verfügt über zwei primäre Betriebsmodi:

- Mapper/Aligner
- Varianten-Caller

Das DRAGEN-System kann beide Modi unabhängig voneinander ausführen oder auch als Komplettlösung verwenden. Außerdem ermöglicht das DRAGEN-System die Aktivierung/Deaktivierung von Dekomprimierung, Sortierung, Dublettenkennzeichnung und Komprimierung in der DRAGEN-Pipeline.

- **Modus mit vollständiger Pipeline**

  Wenn Sie den Modus mit vollständiger Pipeline ausführen möchten, legen Sie *--enable-variant-caller* auf *true* fest und stellen Sie eine Eingabe in Form nicht gemappter Reads im *.fastq-, *.bam- oder *.cram-Format bereit.

  DRAGEN führt die Dekomprimierung, das Mapping, das Alignment, die Sortierung sowie die optionale Dublettenkennzeichnung durch und übergibt die Daten direkt an den Varianten-Caller, der eine VCF-Datei erstellt. In diesem Modus verwendet DRAGEN für die gesamte Pipeline parallele Phasen, was die Gesamtlaufzeit deutlich verkürzt.

- **Mapping-Alignment-Modus**

  Der Mapping-Alignment-Modus ist standardmäßig aktiviert. Die Eingabe erfolgt in Form von nicht gemappten Reads im *.fastq-, *.bam- oder *.cram-Format. DRAGEN generiert eine alignierte und sortierte BAM- oder CRAM-Datei. Legen Sie *--enable-duplicate-marking* auf *true* fest, wenn Sie doppelte Reads gleichzeitig kennzeichnen möchten.

- **Varianten-Caller-Modus**

  Legen Sie die Option *--enable-variant-caller* auf „true“ fest, um den Varianten-Caller-Modus auszuführen.

  Die Eingabe erfolgt in Form einer gemappten und alignierten BAM-Datei. DRAGEN generiert eine VCF-Datei. Ist die BAM-Datei bereits sortiert, kann die Sortierung durch Festlegen von *--enable-sort* auf *false* übersprungen werden. Für BAM-Dateien kann in der DRAGEN-Pipeline vor dem Varianten-Calling keine Dublettenkennzeichnung erfolgen, wenn diese nicht bereits zuvor gekennzeichnet wurden. Verwenden Sie den Komplettmodus, wenn Sie die Dublettenkennzeichnung nutzen möchten.

- **RNA-Seq-Daten**

  Legen Sie für die Verarbeitung von RNA-Seq-basierten Daten *--enable-rna* auf *true* fest.

  DRAGEN verwendet während der Mapper-/Aligner-Phase den RNA-Spliced-Aligner.
  Der DRAGEN Bio-IT-Prozessor wechselt dynamisch zwischen den erforderlichen Betriebsmodi.

- Bisulfit-MethylSeq-Daten

  Legen Sie für die Verarbeitung von Bisulfit-MethylSeq-Daten die Option *--enable-methylation-calling* auf „true“ fest. DRAGEN automatisiert die Verarbeitung von Daten für Lister- und Cokus-Protokolle (direktional und nicht direktional). Es wird eine einzelne BAM mit Bismark-kompatiblen Tags generiert.

  Wahlweise lässt sich DRAGEN auch in einem Modus ausführen, bei dem für jede Kombination aus C->T- und G->A-konvertierten Reads und Referenzen eine separate BAM-Datei erstellt wird. Zur Aktivierung dieses Verarbeitungsmodus müssen Sie bei aktivierter Option *--ht-methylated* eine Gruppe von Referenz-Hashtabellen erstellen und *dragen* mit der entsprechenden *--methylation-protocol*-Einstellung ausführen.

Dieser Abschnitt enthält im weiteren Verlauf ausführliche Informationen für eine genauere Steuerung der DRAGEN-Pipeline.

Ausgabeoptionen

Die folgenden Befehlszeilenoptionen für die Ausgabe sind obligatorisch:

- *--output-directory <Ausgabeverzeichnis>* gibt das Ausgabeverzeichnis für generierte Dateien an.
- *--output-file-prefix <Ausgabepräfix>* gibt das Präfix der Ausgabedatei an. DRAGEN hängt an dieses Präfix bei jeder generierten Datei die richtige Dateierweiterung an.
- *-r [--ref-dir]* gibt die Referenz-Hashtabelle an.

Der Kürze halber sind diese obligatorischen Optionen in den folgenden Beispielen nicht enthalten.

Für Mapping und Alignment wird die Ausgabe vor dem Speichern auf der Festplatte standardmäßig sortiert und in das BAM-Format komprimiert. In der Mapping-Alignment-Phase kann der Benutzer das Ausgabeformat mit der Option *--output-format <SAM|BAM|CRAM>* steuern. Wenn die Ausgabedatei vorhanden ist, gibt die Software eine Warnung aus und beendet den Vorgang. Wenn die Ausgabedatei bereits vorhanden ist, können Sie mit der Option *-f [--force]* das Überschreiben erzwingen.

Mit folgenden Befehlen erfolgt beispielsweise die Ausgabe in eine komprimierte BAM-Datei und das Überschreiben wird erzwungen:

```
dragen ... -f
dragen ... -f --output-format bam
```

Legen Sie zum Erstellen einer BAM-Indexdatei im BAI-Format (Dateierweiterung .bai) *--enable-bam-indexing* auf „true“ fest.

Das folgende Beispiel führt zur Ausgabe einer SAM-Datei und erzwingt das Überschreiben:

```
dragen ... -f --output-format sam
```

Das folgende Beispiel führt zur Ausgabe einer CRAM-Datei und erzwingt das Überschreiben:

```
dragen ... -f --output-format cram
```

Wie im BAM-Standard beschrieben, kann DRAGEN Mismatch Difference(MD)-Tags erstellen. Da es bei der Erstellung dieser Zeichenfolge jedoch zu geringen Leistungseinbußen kommt, ist diese Funktion standardmäßig deaktiviert. Legen Sie *--generate-md-tags* auf „true“ fest, wenn Sie MD-Tags generieren möchten.

Legen Sie *--generate-zs-tags* auf „true“ fest, wenn Sie ZS:Z-Alignmentstatus-Tags generieren möchten. Diese Tags werden nur im primären Alignment und nur dann generiert, wenn ein Read über suboptimale Alignments verfügt, die ihn für eine sekundäre Ausgabe qualifizieren (auch dann, wenn keine Ausgabe vorliegt, weil *--Aligner.sec-aligns* auf 0 festgelegt wurde). Gültige Tag-Werte sind:

- ZS:Z:R: Es wurden mehrere Alignments mit ähnlichem Score gefunden.

- ZS:Z:NM: Es wurden keine Alignments gefunden.
- ZS:Z:QL: Es wurde ein Alignment gefunden, das jedoch unterhalb des Qualitätsschwellenwerts lag.

Legen Sie *--generate-sa-tags* auf „true“ (den Standardwert) fest, wenn Sie SA:Z-Tags generieren möchten. Diese Tags stellen Alignment-Informationen (Position, CIGAR, Ausrichtung) von Gruppen zusätzlicher Alignments bereit, die beim Calling struktureller Varianten von Nutzen sind.

Eingabeoptionen

Das DRAGEN-System kann Reads im FASTQ- oder im BAM-/CRAM-Format verarbeiten. FASTQ-Eingabedateien mit der Endung .gz werden von DRAGEN mithilfe hardwarebeschleunigter Dekomprimierung automatisch dekomprimiert.

FASTQ-Eingabedateien

FASTQ-Eingabedateien können Single-End- und Paired-End-Daten enthalten, wie in den folgenden Beispielen dargestellt.

- Single-End-Daten in einer FASTQ-Datei *(Option -1)*

```
dragen -r <REFERENZVERZEICHNIS> -1 <fastq> --output-directory
   <AUSGABEVERZEICHNIS> \
   --output-file-prefix <AUSGABEPRÄFIX> --RGID <RGID> --RGSM <RGSM>
```

- Paired-End-Daten in zwei zusammengehörigen FASTQ-Dateien *(Optionen -1 und -2)*

```
dragen -r <REFERENZVERZEICHNIS> -1 <fastq1> -2 <fastq2> \
   --output-directory <AUSGABEVERZEICHNIS> --output-file-prefix
   <AUSGABEPRÄFIX> \
   --RGID <RGID> --RGSM <RGSM>
```

- Paired-End-Daten in einer überlappenden FASTQ-Datei *(Option --interleaved (-i))*

```
dragen -r <REFERENZVERZEICHNIS> -1 <ÜBERLAPPENDE_FASTQ> -i \
   --RGID <RGID> --RGSM <RGSM>
```

FASTQ-Proben können in mehrere Dateien segmentiert werden, um die Dateigröße zu begrenzen oder die für die Erstellung erforderliche Zeit zu verkürzen. Der bcl2fastq- und der DRAGEN BCL-Befehl verwenden dieselbe Konvention für Dateinamen:

```
<Proben-ID>_S<#>_<Lane>_<Read>_<Segment-Nr.>.fastq.gz
```

Beispiele:

```
RDRS182520_S1_L001_R1_001.fastq.gz
RDRS182520_S1_L001_R1_002.fastq.gz
...
RDRS182520_S1_L001_R1_008.fastq.gz
```

Die Dateinamenskonvention ist bei HiSeqX- und NextSeq-Geräten unterschiedlich.

Diese Dateien müssen nicht verkettet werden, damit sie von DRAGEN gemeinsam verarbeitet werden können. Geben Sie für das Mapping/Alignment von Proben die erste Datei in der Reihe an (*-1 <Dateiname>_001.fastq*). DRAGEN liest alle Segmentdateien in der Probe der Reihenfolge nach ein. Dies gilt sowohl für FASTQ-Dateifolgen, die mit den Optionen -1 und -2 für die Paired-End-Eingabe angegeben wurden, als auch für komprimierte fastq.gz-Dateien. Dieses Verhalten lässt sich deaktivieren, indem die Befehlszeilenoption *--enable-auto-multifile* auf „false“ festgelegt wird.

DRAGEN kann wahlweise auch mehrere Dateien nach dem im Dateinamen angegebenen Probennamen einlesen, wodurch sich über mehrere BCL-Lanes oder -Fließzellen verteilte Proben zusammenfassen lassen. Legen Sie die Option *--combine-samples-by-name* auf „true“ fest, um diese Funktion zu aktivieren.

Wenn die in der Befehlszeile angegebenen FASTQ-Dateien die oben dargestellte Dateinamenskonvention von Casava 1.8 verwenden und weitere Dateien mit diesem Probennamen im selben Verzeichnis vorhanden sind, werden diese Dateien und sämtliche zugehörigen Segmente automatisch verarbeitet. Beachten Sie, dass der Probenname, die Read-Nummer und die Dateinamenserweiterung übereinstimmen müssen. Der Index-Barcode und die Lane-Nummer können abweichen.

Die Eingabedateien müssen sich in einem schnellen Dateisystem befinden, um eine Beeinträchtigung der Systemleistung zu verhindern.

fastq-list-Eingabedatei

Zur Bereitstellung mehrerer FASTQ-Eingabedateien wird die Option *--fastq-list <Name der CSV-Datei>* empfohlen. Mithilfe dieser Option können Sie den Namen einer CSV-Datei, die eine Liste der FASTQ-Dateien enthält, angeben, anstatt die Option *--combine-samples-by-name* zu verwenden. Beispiel:

```
dragen -r <Referenzverzeichnis> --fastq-list <CSV-DATEI> \
   --fastq-list-sample-id <Proben-ID> \
   --output-directory <AUSGABEVERZEICHNIS> --output-file-prefix
   <AUSGABEPRÄFIX>
```

Bei Verwendung einer CSV-Datei können FASTQ-Eingabedateien beliebig benannt werden und mehrere Unterverzeichnisse nutzen. Für jede Read-Gruppe können BAM-Tags explizit angegeben werden. DRAGEN generiert bei der BCL-Konvertierung in das FASTQ-Format automatisch eine CSV-Datei im korrekten Format. Diese CSV-Datei mit der Bezeichnung fastq_list.csv enthält für jede während des Laufs generierte FASTQ-Datei bzw. für jedes während des Laufs generiertes Paired-End-Dateipaar einen Eintrag.

FASTQ-CSV-Dateiformat

Die erste Zeile der CSV-Datei enthält die Titel der einzelnen Spalten. Anschließend folgt mindestens eine Datenzeile. Sämtliche Zeilen der CSV-Datei müssen dieselbe Anzahl kommagetrennter Werte enthalten. Es dürfen weder Leerzeichen noch nicht erforderliche Zeichen enthalten sein.

Bei den Spaltentiteln wird zwischen Groß- und Kleinschreibung unterschieden. Die folgenden Spaltentitel sind erforderlich:

- RGID: Read-Gruppe
- RGSM: Proben-ID
- RGLB: Bibliothek
- Lane: Fließzellen-Lane
- Read1File: vollständiger Pfad zu einer gültigen FASTQ-Eingabedatei
- Read2File: vollständiger Pfad zu einer gültigen FASTQ-Eingabedatei (erforderlich für die Paired-End-Eingabe, andernfalls leer lassen)

In der CSV-Liste kann auf jede FASTQ-Datei jeweils nur einmal verwiesen werden. Die Werte in der Spalte Read2File müssen entweder alle einen Pfad zu einer gültigen Datei enthalten oder alle leer sein.

Wenn eine BAM-Datei mit der fastq-list-Eingabe generiert wird, wird eine Read-Gruppe pro eindeutigem RGID-Wert generiert. Die BAM-Kopfzeile enthält RG-Tags für Folgendes:

- ID (aus RGID)
- SM (aus RGSM)

- LB (aus RGLB)

Für die einzelnen Read-Gruppen können zusätzliche Tags angegeben werden, indem ein Spaltentitel aus vier Großbuchstaben angegeben wird, der mit RG beginnt. Fügen Sie für ein PU-Tag (Plattform Unit, Plattformeinheit) beispielsweise die Spalte RGPU hinzu und geben Sie in dieser Spalte die Werte für die einzelnen Read-Gruppen an. Spaltentitel müssen eindeutig sein.

Eine fastq-list-Datei kann Dateien für mehr als eine Probe enthalten. Wenn eine fastq-list-Datei nur einen eindeutigen RGSM-Eintrag enthält, müssen keine weiteren Optionen angegeben werden und DRAGEN verarbeitet alle in der fastq-list-Datei aufgeführten Dateien. Wenn mehr als ein eindeutiger RGSM-Eintrag in einer fastq-list-Datei enthalten ist, muss zusätzlich zu *--fastq-list <Dateiname>* eine der beiden folgenden Optionen angegeben werden.

- Verwenden Sie *--fastq-list-sample-id <Proben-ID>* für die Verarbeitung einer bestimmten Probe aus der CSV-Datei. Es werden nur die Einträge in der fastq-list-Datei verarbeitet, deren RGSM-Wert der angegebenen Proben-ID entspricht.
- Legen Sie *--fastq-list-all-samples* auf „true“ fest, um sämtliche Proben unabhängig vom RGSM-Wert gemeinsam im selben Lauf zu verarbeiten.

HINWEIS

Bei einem einzelnen Lauf werden nur eine BAM- und eine VCF-Ausgabedatei erstellt, da davon ausgegangen wird, dass alle Eingabe-Read-Gruppen zur selben Probe gehören. Wenn mehrere Proben aus einem BCL-Konvertierungslauf verarbeitet werden sollen, muss die DRAGEN-Sekundäranalyse mehrfach mit unterschiedlichen Werten für die Option *--fastq-list-sample-id* ausgeführt werden.

Es gibt keine Option, mit der Gruppen oder Untergruppen von RGSM-Werten für eine komplexere Filterung angegeben werden können. Jedoch lässt sich derselbe Effekt durch eine Modifikation der fastq-list-Datei erreichen.

Im Folgenden finden Sie ein Beispiel für eine CSV-Datei mit einer FASTQ-Liste, die die erforderlichen Spalten enthält:

```
RGID,RGSM,RGLB,Lane,Read1File,Read2File
   CACACTGA.1,RDSR181520,UnknownLibrary,1,/staging/RDSR181520_S1_L001_R1_
   001.fastq, /staging/RDSR181520_S1_L001_R2_001.fastq
   AGAACGGA.1,RDSR181521,UnknownLibrary,1,/staging/RDSR181521_S2_L001_R1_
   001.fastq, /staging/RDSR181521_S2_L001_R2_001.fastq
   TAAGTGCC.1,RDSR181522,UnknownLibrary,1,/staging/RDSR181522_S3_L001_R1_
   001.fastq, /staging/RDSR181522_S3_L001_R2_001.fastq
   AGACTGAG.1,RDSR181523,UnknownLibrary,1,/staging/RDSR181523_S4_L001_R1_
   001.fastq, /staging/RDSR181523_S4_L001_R2_001.fastq
```

Wenn Sie die Option *--tumor-fastq-list* für die somatische Eingabe nutzen, verwenden Sie die Option *--tumor-fastq-list-sample-id <Proben-ID>* zur Angabe der Proben-ID für die entsprechende FASTQ-Liste. Beispiel:

```
dragen -r <Referenzverzeichnis> --tumor-fastq-list <CSV-Datei> \
   --tumor-fastq-list-sample-id <Proben-ID> \
   --output-directory <Ausgabeverzeichnis> \
   --output-file-prefix <Ausgabepräfix> --fastq-list <CSV-Datei_2> \
   --fastq-list-sample-id <Proben-ID_2>
```

BAM-Eingabedateien

Legen Sie *--enable-map-align* auf „true“ fest, wenn Sie BAM-Dateien als Eingabe für den Mapper/Aligner verwenden möchten. Behalten Sie die Standardeinstellung „false“ bei, wenn Sie die BAM-Datei als Eingabe für den Varianten-Caller verwenden möchten.

Wenn Sie eine BAM-Datei als Eingabe festlegen, ignoriert DRAGEN alle Alignment-Informationen in der Eingabedatei und gibt für alle Reads neue Alignments aus. Wenn in der Eingabedatei Paired-End-Reads enthalten sind, muss festgelegt werden, dass die Eingabedaten sortiert werden, sodass Paare zusammen verarbeitet werden können. Für andere Pipelines ist eine Neusortierung des Eingabedatensatzes nach Read-Name erforderlich. DRAGEN beschleunigt diesen Vorgang erheblich, da die Eingabe-Reads in Paaren angeordnet und nach Identifizierung von Paaren an den Mapper/Aligner gesendet werden. Sie können diese Funktion mit der Option *--pair-by-name* aktivieren oder deaktivieren. (Die Standardeinstellung ist „true“.)

Verwenden Sie für Single-End-Eingaben in einer BAM-Datei die Optionen *(-b)* und `--pair-by-name=false` wie folgt:

```
dragen -r <Referenzverzeichnis> -b <BAM> --output-directory
   <Ausgabeverzeichnis> \
   --output-file-prefix <Ausgabepräfix> --pair-by-name false
```

Verwenden Sie für Paired-End-Eingaben in einer BAM-Datei die Optionen *(-b)* und `--pair-by-name=true` wie folgt:

```
dragen -r <Referenzverzeichnis> -b <BAM> --output-directory
   <Ausgabeverzeichnis> \
   --output-file-prefix <Ausgabepräfix> --pair-by-name true
```

CRAM-Eingabe

Sie können CRAM-Dateien als Eingabe für den Mapper/Aligner und den Varianten-Caller von DRAGEN verwenden. Die bei Verwendung der CRAM-Eingabe verfügbaren DRAGEN-Funktionen sind mit den Funktionen bei Verwendung der BAM-Eingabe identisch.

Die Option *--cram-reference* wird nicht mehr benötigt. Der CRAM-Komprimierer und -Dekomprimierer verwendet die DRAGEN-Referenz.

Folgende Optionen werden verwendet, um eine CRAM-Eingabe für den Mapper/Aligner oder den Varianten-Caller bereitzustellen:

- *--cram-input*: Name und Pfad der CRAM-Datei
- *--cram-input*: Ein Anwendungsbeispiel ist die Paired-End-Eingabe in einer einzelnen CRAM-Datei. Legen Sie außerdem die Option *--pair-by-name* auf „true“ fest.

  ```
  dragen -r <Referenzverzeichnis> --cram-input <CRAM> --output-directory
     <Ausgabeverzeichnis> \
     --output-file-prefix <Ausgabepräfix> --pair-by-name true
  ```

Handhabung von N-Basen

Eine der Methoden, mit der DRAGEN die Handhabung von Sequenzen optimiert, kann zum Überschreiben des Basenqualitäts-Scores führen, der N-Basen-Calls zugewiesen wurde.

Mit den Optionen *--fastq-n-quality* und *--fastq-offset* werden die Basenqualitäts-Scores mit einer festen Basenqualität überschrieben. Die Standardwerte für diese Optionen lauten 2 bzw. 33. Zusammen entsprechen sie der Mindestqualität von Illumina von 35 (ASCII-Zeichen „#“).

Read-Bezeichnungen für Paired-End-Reads

Gemäß einer gängigen Konvention können Read-Bezeichnungen Suffixe (wie „/1“ oder „/2“) enthalten, die darauf verweisen, welches Ende eines Paares der Read repräsentiert. Bei BAM-Eingaben mit der Option *--pair-by-name* ignoriert DRAGEN diese Suffixe, um übereinstimmende Paarbezeichnungen zu finden. Standardmäßig verwendet DRAGEN den Schrägstrich als Trennzeichen für diese Suffixe und ignoriert „/1“ und „/2“ beim Vergleich von Bezeichnungen. Standardmäßig entfernt DRAGEN diese Suffixe von den ursprünglichen Read-Bezeichnungen.

DRAGEN bietet folgende Optionen zur Steuerung der Verwendung von Suffixen:

- Verwenden Sie die Option *--pair-suffix-delimiter*, wenn Sie das Trennzeichen für Suffixe ändern möchten. Gültige Werte für diese Option sind Schrägstrich (/), Punkt (.) und Doppelpunkt (:).
- Legen Sie *--strip-input-qname-suffixes* auf „false“ fest, wenn Sie die gesamte Bezeichnung einschließlich der Suffixe beibehalten möchten.
- Legen Sie *--append-read-index-to-name* auf „true“ fest, um einen neuen Satz von Suffixen an alle Read-Bezeichnungen anzuhängen, wobei das Trennzeichen durch die Option *--pair-suffix-delimiter* bestimmt wird. Standardmäßig wird der Schrägstrich als Trennzeichen verwendet, sodass „/1“ und „/2“ zu den Bezeichnungen hinzugefügt werden.

Eingabedateien für die Gen-Annotation

Beim Verarbeiten von RNA-Seq-Daten können Sie mithilfe der Option *--annotation-file* eine Annotationsdatei bereitstellen. Mit dieser Datei steigern Sie die Genauigkeit beim Mapping und Alignment (siehe *Eingabedateien* auf Seite 113). Die Datei muss den GTF-/GFF-Formatspezifikationen entsprechen und annotierte Transkripte auflisten, die mit dem für das Mapping verwendeten Referenzgenom übereinstimmen. Das ähnliche Format GFF3 wird derzeit nicht unterstützt.

DRAGEN kann die Datei SJ.out.tab (siehe *SJ.out.tab* auf Seite 116) als Annotationsdatei verwenden und den Aligner damit im Two-Pass-Modus unterstützen.

Probengeschlecht

Mit der Option *--sample-sex* wird in der Befehlszeile das Probengeschlecht angegeben. Die Information wird an alle Caller (Caller für kleine Varianten, CNV, SV und Repeat Genotyper) übergeben. Der CNV-Caller verfügt über eine Geschlechtsbestimmungsfunktion. Wird *--sample-sex* angegeben, hat diese Angabe jedoch Vorrang vor dem bestimmten Geschlecht. Beispiele:

```
--sample-sex MALE
--sample-sex FEMALE
```

Beibehalten oder Entfernen von BQSR-Tags

Das Picard Base Quality Score Recalibration(BQSR)-Tool gibt BAM-Ausgabedateien mit den Tags BI und BD aus. BQSR berechnet diese Tags in Relation zur exakten Sequenz für einen Read. Wenn eine BAM-Datei mit BI- und BD-Tags als Eingabe für Mapper/Aligner mit aktiviertem Hard Clipping verwendet wird, können die BI- und/oder BD-Tags ungültig werden.

Es wird empfohlen, diese Tags bei der Verwendung von BAM-Dateien als Eingabe zu entfernen. Legen Sie zum Entfernen der BI- und BD-Tags die Option *--preserve-bqsr-tags* auf *false* fest. Wenn Sie die Tags beibehalten, gibt DRAGEN eine Warnung bezüglich der Deaktivierung von Hard Clipping aus.

Optionen für Read-Gruppen

DRAGEN geht davon aus, dass alle in einem bestimmten FASTQ enthaltenen Reads zur selben Read-Gruppe gehören. Das DRAGEN-System erzeugt in der Kopfzeile der BAM-Ausgabedatei einen einzelnen @RG-Read-Gruppendeskriptor, der die folgenden BAM-Standardattribute angeben kann:

Attribut	Argument	Beschreibung
ID	--RGID	Read-Gruppenbezeichner. Wenn Read-Gruppenparameter enthalten sind, ist RGID erforderlich. Hierbei handelt es sich um den in jedem BAM-Ausgabedatensatz erfassten Wert.
LB	--RGLB	Bibliothek.
PL	--RGPL	Zum Erstellen der Reads verwendete Plattform/Technologie. Der BAM-Standard lässt die Werte CAPILLARY, LS454, ILLUMINA, SOLID, HELICOS, IONTORRENT und PACBIO zu.
PU	--RGPU	Plattformeinheit, z. B. flowcell-barcode.lane.
SM	--RGSM	Probe.
CN	--RGCN	Name des Sequenzierungszentrums, das den Read erstellt hat.
DS	--RGDS	Beschreibung.
DT	--RGDT	Erstellungsdatum des Laufs.
PI	--RGPI	Prognostizierte mittlere Insert-Größe.

Wenn eines dieser Argumente vorhanden ist, fügt die DRAGEN-Software an alle Ausgabedatensätze ein RG-Tag an, um anzuzeigen, dass die Datensätze zu einer Read-Gruppe gehören. Das folgende Beispiel zeigt eine Befehlszeile, in der Read-Gruppenparameter enthalten sind:

```
dragen --RGID 1 --RGCN Broad --RGLB Solexa-135852 \
   --RGPL Illumina --RGPU 1 --RGSM NA12878 \
   -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149 \
   -1 SRA056922.fastq --output-directory /staging/tmp/ \
   --output-file-prefix rg_example
```

Wenn die Option *--fastq-list* für die Eingabe mehrerer Read-Gruppen verwendet wird, werden für jede Read-Gruppe BAM-Tags (und weitere Tags) angegeben, indem zur fastq_list.csv-Datei Spalten hinzugefügt werden. Jede Spaltenüberschrift besteht aus vier Großbuchstaben und beginnt mit „RG“. In jeder Spalte werden die Werte der einzelnen Read-Gruppen für diese Spalte in einem Tag mit derselben Bezeichnung an die BAM-Ausgabedatei weitergegeben.

Lizenzoptionen

Mithilfe der Option *--lic-no-print* können Sie die Lizenzstatusmeldung am Ende eines Laufs unterdrücken. Im Folgenden ist ein Beispiel für eine Lizenzstatusmeldung abgebildet:

```
LICENSE_MSG| ==================================================
   LICENSE_MSG| License report
   LICENSE_MSG|   Genome status [ACxxxxxxxxxx] : used 1263.9 Gbases
   since 2018-Feb-15 (1263886160894 bases, unlimited)
   LICENSE_MSG|   Genome  bases [ACxxxxxxxxxx] : 202000000
   LICENSE_MSG|   Genome  bases [total]        : 202000000
```

Automatisch erstellte MD5SUM für BAM- und CRAM-Ausgabedateien

Für BAM- und CRAM-Ausgabedateien wird automatisch eine MD5SUM-Datei erstellt. Die MD5SUM-Datei ist so wie die Ausgabedatei benannt, verfügt jedoch zusätzlich über eine MD5SUM-Erweiterung (z. B. whole_genome_run_123.bam.md5sum). Die MD5SUM-Datei ist eine einzeilige Textdatei, die die md5sum der Ausgabedatei enthält. Sie entspricht exakt der Ausgabe des md5sum-Befehls von Linux.

Die MD5SUM-Berechnung wird gleichzeitig mit dem Schreiben der Ausgabedatei durchgeführt. Es gibt daher keine messbare Leistungsauswirkung (im Vergleich zum md5sum-Befehl von Linux, der bei einer 30x BAM-Datei einige Minuten dauern kann).

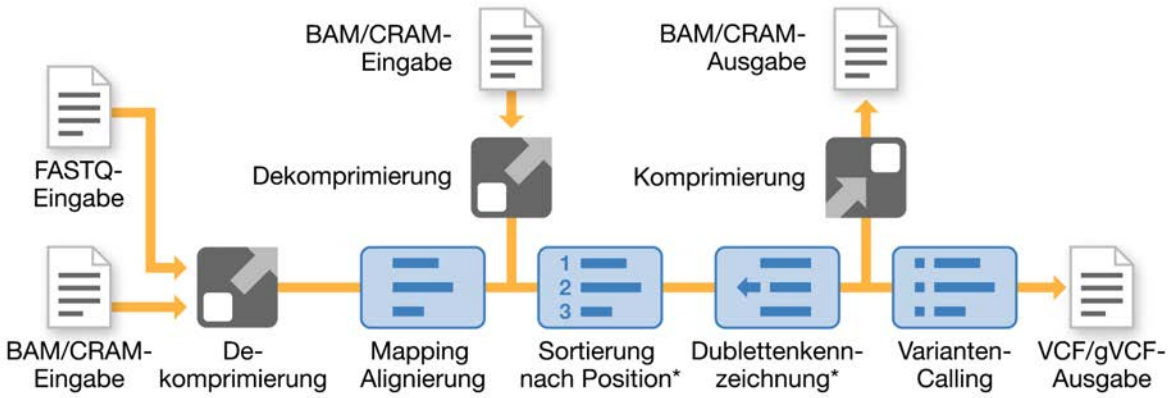
Konfigurationsdateien

Befehlszeilenoptionen können in einer Konfigurationsdatei gespeichert werden. Die Standardkonfigurationsdatei wird unter /opt/edico/config/dragen-user-defaults.cfg gespeichert. Sie können mit der Option *--config-file (-c)* eine andere Datei angeben und die vorhandene überschreiben. Die für einen Lauf verwendete Konfigurationsdatei enthält die Standardeinstellungen für diesen Lauf, die sich alle mit Befehlszeilenoptionen überschreiben lassen.

Empfohlen wird die Verwendung der Datei dragen-user-defaults.cfg als Vorlage zum Erstellen von Standardeinstellungen für unterschiedliche Anwendungsszenarien. Kopieren Sie dragen-user-defaults.cfg, benennen Sie die Kopie um und bearbeiten Sie die neue Datei passend zur spezifischen Anwendung. Die Best Practice besteht darin, selten geänderte Optionen in die Konfigurationsdatei aufzunehmen und die laufspezifischen Optionen über die Befehlszeile anzugeben.

Kapitel 3 DRAGEN-DNA-Pipeline

Abbildung 2 DNA-Pipeline für DRAGEN



* Optional

DNA-Mapping

Option für die Seed-Dichte

Die Option *seed-density* steuert, nach wie vielen (normalerweise überlappenden) primären Seeds aus jedem Read der Mapper in der Hashtabelle nach exakten Übereinstimmungen sucht. Der maximale Wert für die Dichte von 1.0 generiert einen Seed, der bei jeder Read-Position startet, z. B. (L-K+1) K-Basen-Seeds von einem L-Basen-Read.

Die Seed-Dichte muss zwischen 0.0 und 1.0 liegen. Intern wird ein verfügbares Seed-Muster ausgewählt, das der angeforderten Dichte genau oder ungefähr entspricht. Das am dünnsten besetzte Muster ist ein Seed je 32 Positionen bzw. eine Dichte von 0.03125.

- **Hinweise hinsichtlich der Genauigkeit:** In der Regel verbessert ein dichteres Seed-Suchmuster die Mapping-Genauigkeit. Für mäßig lange Reads (z. B. über 50 bp) und niedrige Sequenzierer-Fehlerraten wird über der Standarddichte für Seed-Muster von 50 % keine höhere Genauigkeit erzielt.
- **Hinweise hinsichtlich der Geschwindigkeit:** Dichtere Seed-Suchmuster verlangsamen in der Regel das Mapping, wobei dünn besetzte Seed-Muster den Vorgang beschleunigen. Wenn jedoch das Seed-Mapping schneller als das Alignment ausgeführt werden kann, wird der Mapper durch ein dünner besetztes Seed-Muster nicht wesentlich schneller.

Beziehung zum Referenz-Seed-Intervall

Bezogen auf die Funktion hat die Dichte des Seed-Suchmusters einen Effekt, der vergleichbar ist mit dem der Länge des Referenz-Seed-Intervalls (Build-Hashtabellenoption *--ht-ref-seed-interval*). Das Ausfüllen von 100 % der Referenz-Seed-Positionen und Durchsuchen von 50 % der Read-Seed-Positionen hat denselben Effekt wie das Ausfüllen von 50 % Referenz-Seed-Positionen und Durchsuchen von 100 % der Read-Seed-Positionen. Die erwartete Dichte der Seed-Treffer beträgt in beiden Fällen 50 %.

Allgemeiner ausgedrückt ist die erwartete Dichte der Seed-Treffer das Produkt aus der Referenz-Seed-Dichte (der Umkehr des Referenz-Seed-Intervalls) und der Dichte der Seed-Suche. Beispielsweise beträgt bei 50 % ausgefüllten Referenz-Seeds und einer Suche an 33,3 % (1/3) der Read-Seed-Positionen die erwartete Seed-Trefferdichte 16,7 % (1/6).

DRAGEN passt sein präzises Seed-Suchmuster automatisch an und gewährleistet so, dass die aus der Referenz ausgefüllten Seed-Positionen nicht systematisch übergangen werden. Beispielsweise durchsucht der Mapper keine Seeds, die nur mit ungeraden Positionen in der Referenz übereinstimmen, wenn in der Hashtabelle ausschließlich gerade Positionen ausgefüllt sind. Dies gilt auch bei einem Seed-Intervall von 2 und einer Seed-Dichte von 0,5.

Ausrichtungsoption für das Mapping

Die Option *--Mapper.map-orientations* wird beim Mapping von Reads für die Bisulfit-Methylierungsanalyse verwendet. Sie wird automatisch auf Grundlage des für *--methylation-protocol* festgelegten Werts festgelegt.

Mit der Option *--Mapper.map-orientations* kann die Ausrichtung des Read-Mappings so festgelegt werden, dass diese im Referenzgenom nur vorwärts oder nur in Richtung des Gegenstrangs (umgekehrtes Komplement) erfolgt. Die gültigen Werte für *--map-orientations* lauten wie folgt.

- 0: Beliebige Ausrichtung (Standard)
- 1: Mapping nur vorwärts
- 2: Mapping nur in Richtung des Gegenstrangs

Wenn die Mapping-Ausrichtungen beschränkt sind und Paired-End-Reads verwendet werden, kann die erwartete Paar-Ausrichtung nur FR (und nicht FF oder RF) sein.

Seed-Editing-Optionen

DRAGEN mappt Reads vorrangig durch die Suche nach exakten Übereinstimmungen kurzer Seeds mit der Referenz. Durch die Suche nach editierten Single-SNP-Seeds können jedoch auch Seeds gemappt werden, die um ein Nukleotid von der Referenz abweichen. Bei längeren Reads (mehr als 100 bp) ist Seed-Editing in der Regel nicht erforderlich, da bei längeren Reads die Wahrscheinlichkeit hoch ist, dass mindestens eine exakte Seed-Übereinstimmung enthalten ist. Dies gilt besonders bei der Verwendung von Paired-Ends, da eine Seed-Übereinstimmung von einem der Mates zur Alignierung des Paares ausreichend ist. Seed-Editing kann jedoch beispielsweise helfen, die Mapping-Genauigkeit bei kurzen Single-End-Reads zu erhöhen. Dadurch steigt der Zeitaufwand für das Mapping etwas an. Das Seed-Editing wird mit folgenden Optionen gesteuert:

Tabelle 1 Seed-Editing-Optionen

Name der Befehlszeilenoption	Name der Konfigurationsdateioption
--Mappper.seed-density	seed-density
--Mapper.edit-mode	edit-mode
--Mapper.edit-seed-num	edit-seed-num
--Mapper.edit-read-len	edit-read-len
--Mapper.edit-chain-limit	edit-chain-limit

edit-mode und edit-chain-limit

Mithilfe der Optionen edit-mode und edit-chain-limit lässt sich steuern, wann Seed-Editing zum Einsatz kommt. Folgende vier Werte sind für edit-mode verfügbar:

Modus	Beschreibung
0	Kein Editing (Standardeinstellung)
1	Kettenlängentest

Modus	Beschreibung
2	Gepaarter Kettenlängentest
3	Vollständiges Seed-Editing

Für den Edit-Modus 0 müssen alle Seeds exakt übereinstimmen. Modus 3 ist am aufwendigsten, da alle Seeds editiert werden, die nicht exakt mit der Referenz übereinstimmen. Die Modi 1 und 2 nutzen Heuristik, um nur nach editierten Seeds von Reads zu suchen, die am wahrscheinlichsten für präzises Mapping eingesetzt werden können.

In den Edit-Modi 1 und 2 ist die wichtigste Heuristik ein Seed-Kettenlängentest. Exakte Seeds werden in einem ersten Durchlauf über einen bestimmten Read zur Referenz gemappt und die übereinstimmenden Seeds werden in Ketten von Seeds mit ähnlichem Alignment gruppiert. Wenn die längste Seed-Kette (im Read) einen Schwellenwert für edit-chain-limit überschreitet, ist für den Read kein Seed-Editing erforderlich, da bereits eine vielversprechende Mapping-Position vorhanden ist.

Edit-Modus 1 löst Seed-Editing unter Verwendung des Seed-Kettenlängentests aus. Wenn keine Seed-Kette den Wert für *edit-chain-limit* überschreitet (auch wenn es keine exakte Seed-Übereinstimmung gibt), wird ein zweiter Seed-Mapping-Durchlauf mit editierten Seeds durchgeführt. Edit-Modus 2 optimiert die Heuristik für Paired-End-Reads weiter.Wenn bei einem der Mates eine exakte Seed-Kette vorhanden ist, deren Länge den Wert für *edit-chain-limit* überschreitet, wird das Seed-Editing für dieses Paar deaktiviert, da es wahrscheinlich ist, dass mit einem Rescue-Scan das Alignment des Mates auf Grundlage von Seed-Übereinstimmungen aus einem Read rekonstruiert werden kann. Edit-Modus 2 entspricht Modus 1 für Single-End-Reads.

edit-seed-num und edit-read-len

Wenn die Heuristik in den Edit-Modi 1 und 2 Seed-Editing auslöst, steuern diese Optionen, wie viele Seed-Positionen beim zweiten Durchlauf über den Read editiert werden. Obwohl exaktes Seed-Mapping mit einem stark überlappenden Seed-Muster, beispielsweise mit Seeds, die bei 50 % oder 100 % der Read-Positionen beginnen, möglich ist, kann der Seed-Editing-Wert zum größten Teil durch die Editierung von Seed-Mustern mit wesentlich geringerer Dichte und selbst aus überhaupt nicht überlappenden Mustern ermittelt werden. Wenn eine Anwendung beim Mapping einen gewissen Zeitaufwand für das Seed-Editing zulässt, gilt allgemein, dass bei gleichem Zeitaufwand eine größere Mapping-Genauigkeit erzielt werden kann, wenn Seeds in Mustern mit geringer Dichte für eine große Anzahl von Reads editiert werden, als beim Editing von Seeds in dicht besetzten Mustern mit einer geringen Anzahl von Reads.

Bei jedem Auslösen von Seed-Editing rufen diese beiden Optionen *edit-seed-num*-Seed-Editing-Positionen ab, die gleichmäßig über die ersten *edit-read-len*-Basen des Reads verteilt sind. Beispielsweise können bei 21-Basen-Seeds mit *edit-seed-num*=6 und *edit-read-len*=100 editierte Seeds bei einem Versatz von {0, 16, 32, 48, 64, 80} vom 5'-Ende beginnen, mit einer Überlappung von 5 Basen bei aufeinanderfolgenden Seeds. Da Sequenzierungstechnologien im Bereich des (5')-Anfangs des jeweiligen Reads oft eine bessere Basenqualität erzielen, kann das Seed-Editing so auf eine Stelle mit besonders hoher Erfolgswahrscheinlichkeit ausgerichtet werden. Wenn ein bestimmter Read kürzer als *edit-read-len* ist, werden weniger Seeds editiert.

Seed-Editing ist aufwendiger, wenn das Referenz-Seed-Intervall (Option zum Generieren von Hashtabellen --*ht-ref-seed-interval*) größer als 1 ist. In den Edit-Modi 1 und 2 werden automatisch zusätzliche Seed-Editing-Positionen generiert, um ein Verfehlen der ausgefüllten Referenz-Seed-Positionen zu vermeiden. Im Edit-Modus 3 kann der Zeitaufwand erheblich steigen, da Abfrage-Seeds, die mit nicht ausgefüllten Referenzpositionen übereinstimmen, in der Regel fehlschlagen und ein Editing auslösen.

DNA-Alignierung

Einstellungen für das Alignment-Scoring nach Smith-Waterman

In der ersten Mapping-Phase werden Seeds aus dem Read generiert und exakte Übereinstimmungen im Referenzgenom gesucht. Diese Ergebnisse werden dann durch Ausführen eines kompletten Smith-Waterman-Alignments an den Positionen mit der höchsten Dichte an Seed-Übereinstimmungen präzisiert. Dieser gut dokumentierte Algorithmus gleicht jede Read-Position mit allen Kandidatenpositionen der Referenz ab. Dieser Abgleich entspricht einer Matrix aus möglichen Alignments zwischen Read und Referenz. Der Smith-Waterman-Algorithmus generiert für jede dieser potenziellen Alignment-Positionen Scores, anhand derer beurteilt wird, ob das beste Alignment mit einer Nukleotid-Übereinstimmung oder -Nichtübereinstimmung (diagonale Bewegung), einer Deletion (horizontale Bewegung) oder einer Insertion (vertikale Bewegung) durch diese Matrixzelle gewandert ist. Bei einer Übereinstimmung zwischen Read und Referenz wird zum Score hinzuaddiert, bei einer Nichtübereinstimmung oder einem Indel wird vom Score subtrahiert. Als Alignment wird der höchste Gesamtscore gewählt, der beim Durchwandern der Matrix erzielt wird.

Die spezifischen für Scores gewählten Werte in diesem Algorithmus weisen darauf hin, wie bei einem Alignment mit mehreren möglichen Interpretationen ein Gleichgewicht zwischen dem möglichen Vorhandensein eines Indels im Gegensatz zu einem oder mehreren SNPs und der Präferenz für ein Alignment ohne Clipping erzielt werden kann. Die DRAGEN-Standardwerte für den Score sind für das Alignieren von Reads moderater Länge zu einem Referenz-Humangesamtgenom für Varianten-Calling-Anwendungen angemessen. Jede beliebige Menge an Smith-Waterman-Score-Parametern stellt jedoch ein ungenaues Modell der Genommutations- und Sequenzierungsfehler dar und unterschiedlich festgelegte Alignment-Score-Werte sind für einige Anwendungen möglicherweise passender.

Folgende Alignment-Optionen regeln das Smith-Waterman-Alignment:

Name der Befehlszeilenoption	Name der Konfigurationsdateioption
--Aligner.global	global
--Aligner.match-score	match-score
--Aligner.match-n-score	match-n-score
--Aligner.mismatch-pen	mismatch-pen
--Aligner.gap-open-pen	gap-open-pen
--Aligner.gap-ext-pen	gap-ext-pen
--Aligner.unclip-score	unclip-score
--Aligner.no-unclip-score	no-unclip-score
--Aligner.aln-min-score	aln-min-score

- *global*

  Die Option *global* (Wert kann 0 oder 1 sein) gibt an, ob das Alignment komplett im Read abgeschlossen werden muss. Ist die Option auf 1 festgelegt, erfolgen Alignments stets komplett wie beim globalen Alignment mit dem Needleman-Wunsch-Algorithmus (auch bei einem unvollständigen Alignment der Referenz). Alignment-Scores können einen positiven oder negativen Wert einnehmen. Ist die Option auf 0 festgelegt, kann ein Clipping der Alignments an einem oder an beiden Read-Ende(n) erfolgen, wie beim lokalen Alignment mit dem Smith-Waterman-Algorithmus. Alignment-Scores können keine negativen Werte einnehmen.

Allgemein wird *global*=0 für längere Reads bevorzugt, sodass signifikante Read-Segmente nach einem Bruch (großes Indel, strukturelle Variante, chimärischer Read usw.) ohne starke Reduzierung des Alignment-Scores geclippt werden können. Die Festlegung *global*=1 hat bei längeren Reads möglicherweise nicht den gewünschten Effekt, da Insertionen bei oder in der Nähe von Read-Enden eine Pseudoclipping-Funktion übernehmen. Bei *global*=0 werden mehrere (chimärische) Alignments gemeldet, wenn verschiedene Read-Teile mit weit auseinanderliegenden Referenzpositionen übereinstimmen.

Manchmal wird *global*=1 für kürzere Reads bevorzugt, die nur unwahrscheinlich mit strukturellen Brüchen überlappen, chimärische Reads nicht unterstützen können und ohne komplettes Alignment vermutlich zu einem falschen Mapping führen.

Verwenden Sie die Option *unclip-score* oder erhöhen Sie den Wert, um eine weiche Präferenz für Alignments ohne Clipping festzulegen, anstatt *global*=1 festzulegen.

- *match-score*

  Die Option *match-score* ist der Score für ein Read-Nukleotid, das mit einem Referenz-Nukleotid (A, C, G oder T) übereinstimmt. Der Wert ist eine Ganzzahl ohne Vorzeichen zwischen 0 und 15. *match_score*=0 kann nur verwendet werden, wenn *global*=1. Ein höherer Übereinstimmungsscore führt zu längeren Alignments und weniger langen Insertionen.

- *match-2-score*

  Die Option match-2-score ist der Score für ein Read-Nukleotid, das mit einem 2-Basen-IUPAC-IUB-Code in der Referenz übereinstimmt (K, M, R, S, W oder Y). Der Wert ist eine Ganzzahl mit Vorzeichen zwischen -16 und 15.

- *match-3-score*

  Die Option *match-3-score* ist der Score für ein Read-Nukleotid, das mit einem 3-Basen-IUPAC-IUB-Code in der Referenz übereinstimmt (B, D, H oder V). Der Wert ist eine Ganzzahl mit Vorzeichen zwischen -16 und 15.

- *match-n-score*

  Die Option *match-n-score* ist der Score für ein Read-Nukleotid, das mit einem N-Code in der Referenz übereinstimmt. Der Wert ist eine Ganzzahl mit Vorzeichen zwischen -16 und 15.

- *mismatch-pen*

  Die Option *mismatch-pen* ist der Abzug (negativer Score) für ein Read-Nukleotid, das mit einem beliebigen Referenz-Nukleotid oder IUPAC-IUB-Code nicht übereinstimmt („N“ ausgenommen, da keine Nichtübereinstimmung möglich). Der Wert ist eine Ganzzahl ohne Vorzeichen zwischen 0 und 63. Ein höherer Abzug aufgrund von Nichtübereinstimmungen führt zu Alignments mit mehr Insertionen, Deletionen und Clippings, um SNPs zu vermeiden.

- *gap-open-pen*

  Die Option *gap-open-pen* ist der Abzug (negativer Score) für die Öffnung einer Lücke (z. B. eine Insertion oder Deletion). Dieser Wert gilt nur für eine 0-Basen-Lücke. Er wird stets der Lückenlänge multipliziert mit gap-ext-pen hinzugefügt. Der Wert ist eine Ganzzahl ohne Vorzeichen zwischen 0 und 127. Ein höherer Abzug aufgrund von geöffneten Lücken führt zu weniger Insertionen und Deletionen einer beliebigen Länge in Alignment-CIGARs, wobei Clipping oder Alignment über SNPs verwendet wird.

- *gap-ext-pen*

  Die Option *gap-ext-pen* ist der Abzug (negativer Score) für die Erweiterung einer Lücke (d. h. eine Insertion oder Deletion) um eine Base. Der Wert ist eine Ganzzahl ohne Vorzeichen zwischen 0 und 15. Ein höherer Abzug aufgrund von erweiterten Lücken führt zu weniger langen Insertionen und Deletionen in Alignment-CIGARs, wobei kurze Indels, Clipping oder Alignment über SNPs verwendet werden.

- *unclip-score*

  Die Option *unclip-score* ist der Score-Zusatz für ein Alignment, das den Anfang oder das Ende eines Reads erreicht. Bei einem kompletten Alignment (End-to-End) wird dieser Zusatz verdoppelt. Der Wert ist eine Ganzzahl ohne Vorzeichen zwischen 0 und 127. Bei einem höheren Zusatz ohne Clipping erreicht das Alignment häufiger den Anfang und/oder das Ende eines Reads, wobei hierfür auch weniger SNPs oder Indels ausreichen.

  Ein Wert ungleich null für unclip-score ist bei *global=0* hilfreich, um eine weiche Präferenz für Alignments ohne Clipping zu erzielen. Zusätze ohne Clipping wirken sich bei *global=1* kaum auf Alignments aus, da komplette Alignments unabhängig davon erzwungen werden (bei 2 × unclip-score wird dem Alignment-Score nichts hinzugefügt, außer *no-unclip-score = 1*). Es wird empfohlen, bei *global=1* den Standardwert für unclip-score zu verwenden, da einige interne Heuristiken Annahmen über das Clipping von lokalen Alignments machen.

  Insbesondere bei längeren Reads kann das Festlegen von unclip-score auf einen viel höheren Wert als gap-open-pen den unerwünschten Effekt haben, dass Insertionen an einem Read-Ende oder in dessen Nähe für Pseudoclipping verwendet werden (wie bei global=1).

- *no-unclip-score*

  Die Option *no-unclip-score* kann den Wert 0 oder 1 einnehmen. Der Standardwert ist 1. Wenn *no-unclip-score* auf 1 festgelegt ist, werden alle Zusätze ohne Clipping (unclip-score), die für ein Alignment verwendet werden, vor der weiteren Verarbeitung vom Alignment-Score entfernt, z. B. beim Vergleich mit *aln-min-score*, beim Vergleich mit anderen Alignment-Scores und bei Berichten in AS- oder XS-Tags. Der Zusatz ohne Clipping wirkt sich jedoch auf das Alignment mit dem besten Score des Smith-Waterman-Alignments für ein gegebenes Referenzsegment aus – mit einer Verzerrung hin zu Alignments ohne Clipping.

  Wenn unclip-score > 0 zu einer Erweiterung eines lokalen Smith-Waterman-Alignments an einem oder beiden Read-Enden führt, bleibt der Alignment-Score bei *no-unclip-score*=0 gleich bzw. erhöht sich und bleibt bei *no-unclip-score*=1 gleich bzw. verringert sich.

  Die Standardeinstellung *no-unclip-score*=1 wird für *global*=1 empfohlen, da alle Alignments komplett sind und kein Bedarf an einem Zusatz für die Alignments besteht.

  Bei einer Änderung in *no-unclip-score* sollten Sie überlegen, ob *aln-min-score* angepasst werden muss. Wenn *no-unclip-score*=0, werden Zusätze ohne Clipping in die Alignment-Scores im Vergleich zur Ebene *aln-min-score* aufgenommen, sodass sich die Untergruppe von Alignments, die durch *aln-min-score* herausgefiltert werden, mit *no-unclip-score* signifikant ändern kann.

- *aln-min-score*

  Die Option *aln-min-score* gibt den akzeptierten minimalen Alignment-Score an. Alle Alignment-Ergebnisse unter diesem Score werden verworfen. Durch Erhöhen oder Verringern von *aln-min-score* kann der Prozentsatz an gemappten Reads reduziert oder erhöht werden. Der Wert ist eine Ganzzahl mit Vorzeichen (mit *global*=0 sind negative Alignment-Scores möglich).

  aln-min-score wirkt sich auch auf die MAPQ-Prognosen aus. Der primäre Faktor bei der MAPQ-Berechnung ist die Differenz zwischen dem besten und dem zweitbesten Alignment-Score und *aln-min-score* dient als suboptimaler Alignment-Score, wenn kein höherer Wert außer dem besten Score gefunden wurde. Daher kann eine Erhöhung von *aln-min-score* den gemeldeten MAPQ für einige Alignment mit niedrigerem Score reduzieren.

Paired-End-Optionen

DRAGEN kann Paired-End-Daten verarbeiten, die entweder über ein FASTQ-Dateipaar oder eine überlappende FASTQ-Datei eingegeben werden. Die Hardware mappt die beiden Enden separat und bestimmt dann einen Satz an Alignments, der mit hoher Wahrscheinlichkeit ein Paar in der erwarteten Ausrichtung bildet und ungefähr über die erwartete Insertgröße verfügt. Die Alignments für die beiden Enden werden in Bezug auf die Qualität der Paarung ausgewertet, wobei der Abzug bei Insertgrößen größer ist, die stark von der erwarteten Größe abweichen. Die folgenden Optionen steuern die Verarbeitung von Paired-End-Daten:

- *pe-orientation*

  Die Option *pe-orientation* gibt die erwartete Paired-End-Ausrichtung an. Nur Paare mit dieser Ausrichtung können als korrekte Paare markiert werden Die gültigen Werte sind:

  - 0: FR (Standard)
  - 1: RF
  - 2: FF

- *unpaired-pen*

  Für Paired-End-Reads werden die besten Mapping-Positionen gemeinsam für jedes Paar bestimmt. Herangezogen wird der größte gefundene Paar-Score, wobei die verschiedenen Kombinationen von Alignments für jeden Mate berücksichtigt werden. Ein Paar-Score ist die Summe der beiden Alignment-Scores minus einen Abzug für die Paarung. Es wird geschätzt, wie unwahrscheinlich es ist, dass die Insertlängen weiter von der mittleren Insertgröße abweichen als dieses alignierte Paar.

  Die Option *unpaired-pen* gibt an, wie viele Alignment-Paar-Scores einen Abzug erhalten, wenn die beiden Alignments nicht über eine gepaarte Position oder Ausrichtung verfügen. Diese Option gibt gleichzeitig den maximalen Paar-Abzug für korrekt gepaarte Alignments mit extremen Insertlängen an.

  Die Option *unpaired-pen* ist entsprechend der möglichen Auswirkung auf den MAPQ-Wert in der Phred-Skala angegeben. Intern wird sie basierend auf Smith-Waterman-Scoring-Parametern im Alignment-Score-Raum skaliert.

- *pe-max-penalty*

  Die Option *pe-max-penalty* schränkt ein, wie stark der geschätzte MAPQ-Wert unter Berücksichtigung des alignierten Mates in der Nähe für einen Read erhöht werden kann. Ein gepaartes Alignment erhält niemals einen MAPQ-Wert, der höher als der bei einem Single-End-Mapping zugewiesene MAPQ-Wert plus diesem Wert liegt. Standardmäßig gilt: *pe-max-penalty* = *mapq-max* = 255, wodurch diese Einschränkung gewissermaßen deaktiviert wird.

  Der wichtigste Unterschied zwischen *unpaired-pen* und *pe-max-penalty* ist, dass *unpaired-pen* die berechneten Paar-Scores beeinflusst und somit auch die Auswahl der Alignments. *pe-max-penalty* beeinflusst nur den gemeldeten MAPQ-Wert für gepaarte Alignments.

Bestimmung der mittleren Insert-Größe

Bei der Verwendung von Paired-End-Daten muss DRAGEN eine Auswahl unter den Alignments mit der höchsten Qualität treffen, um die Paarung von Enden zu ermitteln. Diese Auswahl trifft DRAGEN anhand eines gaußschen statistischen Modells, mit dem die Wahrscheinlichkeit bestimmt wird, dass zwei Alignments zu einem Paar gehören. Das Modell basiert auf der Annahme, dass eine spezifische Bibliotheksvorbereitung Fragmente mit ungefähr gleicher Länge erzeugt, woraus folgt, dass die Insert-Länge der entstandenen Paare um eine bestimmte mittlere Insert-Länge schwankt.

Wenn die statistischen Werte der Bibliotheksvorbereitung für eine Eingabedatei bekannt sind (und die Datei aus einer einzelnen Read-Gruppe besteht), können Sie die Eigenschaften der Insert-Längenverteilung angeben: Mittelwert, Standardabweichung und drei Quartile. Die Eigenschaften lassen sich mit den Optionen *Aligner.pe-stat-mean-insert*, *Alinger.pe-stat-stddev-insert*, *Aligner.pe-stat-quartiles-insert* und *Aligner.pe-stat-mean-read-len* angeben. In der Regel ist es jedoch vorteilhaft, DRAGEN die Eigenschaften automatisch bestimmen zu lassen.

Legen Sie *--enable-sampling* auf „true“ fest, um die automatische Bestimmung der Insert-Längenverteilung zu aktivieren. Bei der Ausführung testet die Software eine Probe mit 100.000 Paaren mithilfe des Aligners, berechnet die Verteilung und verwendet die erhaltenen statistischen Werte zur Bestimmung aller Paare in den Eingabedaten.

Die DRAGEN-Hostsoftware erstellt einen Bericht mit den statistischen Werten im stdout-Protokoll. Dieser sieht aus wie folgt:

```
Final paired-end statistics detected for read group 0, based on 79935
   high quality pairs for FR orientation
   Quartiles (25 50 75) = 398 410 421
   Mean = 410.151
   Standard deviation = 14.6773
   Boundaries for mean and standard deviation: low = 352, high = 467
   Boundaries for proper pairs: low = 329, high = 490
NOTE: DRAGEN's insert estimates include corrections for clipping (so they are
no identical to TLEN)
```

Die Insert-Längenverteilung für die einzelnen Proben wird in der Datei fragment_length_hist.csv gespeichert. Jede Probe beginnt mit den folgenden Zeilen:

```
#Sample: sample name
FragmentLength,Count
```

Auf diese Zeilen folgt das Histogramm.

Ist die Anzahl der Probenpaare besonders gering, sind nicht genügend Daten für eine zuverlässige Bestimmung vorhanden. In diesem Fall verwendet DRAGEN statistische Standardwerte für eine besonders breite Insert-Verteilung. Dabei werden Alignmentpaare oft als zusammengehörig gewertet, selbst wenn diese Zehntausende Basen voneinander entfernt liegen. In diesem Fall gibt DRAGEN die folgende Meldung aus:

```
WARNING: Less than 28 high quality pairs found - standard deviation is
   calculated from the small samples formula
```

Die Formel für kleine Proben berechnet die Standardabweichung wie folgt:

```
if samples < 3 then
     standard deviation = 10000
else if samples < 28 then
    standard deviation = 25 * (standard deviation + 1) / (samples - 2)
end if
if standard deviation < 12 then
     standard deviation = 12
end if
```

Das Standardmodell lautet „standard deviation = 10000“. Wenn die ersten 100.000 Reads nicht zugeordnet sind oder es sich bei allen Paaren um nicht zusammengehörige Paare handelt, werden die Standardabweichung auf 10000 und der Mittelwert sowie die Quartilen auf 0 festgelegt. Beachten Sie, dass der Mindestwert für die Standardabweichung unabhängig von der Anzahl der Proben 12 ist.

Bei RNA-Seq-Daten ist die Insert-Größenverteilung nicht normal, da Paare mit Introns vorhanden sind. Die DRAGEN-Software bestimmt die Verteilung mithilfe eines Kernel-Dichteschätzwerts, damit für die Proben viele Nachkommastellen verwendet werden können. Bei dieser Art der Bestimmung sind Mittelwert und Standardabweichung für RNA-Seq-Daten und zusammengehörige Paare präziser.

Rescue-Scans

Wenn bei Paired-End-Reads ein Seed-Treffer nur für ein Mate gefunden wird, suchen Rescue-Scans nach fehlenden Mate-Alignments innerhalb eines Rescue-Radius der mittleren Insertlänge. Normalerweise legt die DRAGEN-Hostsoftware den Rescue-Radius auf 2,5 Standardabweichungen von der empirischen Insertverteilung fest. Wenn die Insert-Standardabweichung im Vergleich zur Read-Länge jedoch groß ist, wird der Rescue-Radius eingeschränkt, um Mapping-Verzögerungen möglichst gering zu halten. In diesem Fall wird folgende Warnmeldung angezeigt:

```
Rescue radius = 220
     Effective rescue sigmas = 0.5
          WARNING: Default rescue sigmas value of 2.5 was overridden by host software!
          The user may wish to set rescue sigmas value explicitly with --Aligner.rescue-sigmas
```

Der Benutzer kann diese Warnung entweder ignorieren oder einen vorläufigen Rescue-Radius angeben, um die Mapping-Geschwindigkeit nicht zu beeinträchtigen. Es wird empfohlen, den Rescue-Radius auf 2,5 Standardabweichungen festzulegen, um die Mapping-Sensitivität beizubehalten. Legen Sie *max-rescues* auf 0 fest, wenn Sie Rescue-Scans deaktivieren möchten.

Ausgabeoptionen

DRAGEN kann bis zu vier unabhängige Alignments für jeden Read verfolgen. Diese Alignments können chimärisch sein, das heißt, es werden unterschiedliche Regionen des Reads gemappt, oder sie können suboptimale Mappings des Reads zu verschiedenen Bereichen des Alignments sein.

Da DRAGEN die vier besten Alignments für einen Read verfolgt, gibt die Software chimärischen (ergänzenden) Alignments Vorrang vor suboptimalen (sekundären). Sie verfolgt also so viele chimärische Alignments wie möglich bis zu einem Limit von vier. Gibt es weniger als vier chimärische Alignments, werden die verbleibenden Plätze für die Verfolgung von suboptimalen Alignments genutzt.

Mit den folgenden Konfigurationsoptionen können Sie steuern, wie viele von jedem Alignment-Typ in der DRAGEN-Ausgabe berücksichtigt werden sollen.

- *mapq-max*

  Die Option *mapq-max* gibt eine Obergrenze für den geschätzten MAPQ-Wert an, der für jedes Alignment gemeldet werden kann, von 0 bis 255. Ist der berechnete MAPQ höher, wird stattdessen dieser Wert gemeldet. Der Standardwert ist 60.

- *supp-aligns*, *sec-aligns*

  Die Optionen *supp-aligns* und *sec-aligns* begrenzen die jeweils maximale Anzahl an ergänzenden (d. h. chimärisch und SAM FLAG 0x800) Alignments und sekundären (d. h. suboptimal und SAM FLAG 0x100) Alignments, die für die einzelnen Reads gemeldet werden können.

Insgesamt wird ein Maximum von 31 Alignments für jeden Read gemeldet, einschließlich primärer, ergänzender und sekundärer Alignments. Aus diesem Grund liegt der Bereich von *supp-aligns* und *sec-aligns* jeweils bei 0 bis 30. Ergänzende Alignments werden mit höherer Priorität als sekundäre verfolgt und ausgegeben.

Hohe Einstellungen für diese zwei Optionen beeinträchtigen die Geschwindigkeit, daher sollte die Anzahl nur wenn notwendig erhöht werden.

- *sec-phred-delta*

  Die Option *sec-phred-delta* steuert, welche sekundären Alignments basierend auf dem Alignment-Score in Relation zum primären gemeldeten Alignment ausgegeben werden. Nur sekundäre Alignments, die sich wahrscheinlich innerhalb dieses Phred-Werts des primären Alignments befinden, werden gemeldet.

- *sec-aligns-hard*

  Die Option *sec-aligns-hard* unterdrückt die Ausgabe aller sekundären Alignments, wenn mehr sekundäre Alignments vorhanden sind, als ausgegeben werden können. Legen Sie *sec-aligns-hard* auf 1 fest, um zu erzwingen, dass der Read nicht gemappt wird, wenn nicht alle sekundären Alignments ausgegeben werden können.

- *supp-as-sec*

  Wenn die Option *supp-as-sec* auf 1 festgelegt ist, werden ergänzende (chimärische) Alignments mit SAM FLAG 0x100 statt mit 0x800 gemeldet. Der Standardwert ist 0. Die Option *supp-as-sec* ermöglicht die Kompatibilität mit Tools, die FLAG 0x800 nicht unterstützen.

- *hard-clips*

  Die Option *hard-clips* wird als ein Feld von 3 Bits mit einem Wertebereich von 0 bis 7 verwendet.
  Die Bits geben Alignments wie folgt an:
  - Bit 0: primäre Alignments
  - Bit 1: ergänzende Alignments
  - Bit 2: sekundäre Alignments

  Jedes Bit legt fest, ob lokale Alignments dieses Typs mit Hard Clipping (1) oder Soft Clipping (0) gemeldet werden. Der Standardwert ist 6. Primäre Alignments verwenden dementsprechend Soft Clipping und ergänzende sowie sekundäre Alignments verwenden hartes Clipping.

ALT-sensibles Mapping

Das humane Referenzgenom GRCh38 enthält im Vergleich zu früheren Referenzversionen wesentlich mehr alternative Haplotypen (ALT-Contigs). Durch das Einfügen von ALT-Contigs in die Mapping-Referenz werden im Allgemeinen das Mapping und die Spezifität des Varianten-Callings verbessert, da fehlerhafte Alignments für Reads vermieden werden, die mit einem ALT-Contig übereinstimmen, jedoch im Vergleich mit der primären Assembly einen schlechten Score erzielen. Durch das Mapping mit den ALT-Contigs von GRCh38 ohne Sonderverarbeitung kann allerdings die Sensitivität des Varianten-Callings in zugehörigen Regionen deutlich verringert werden. Denn viele Reads alignieren gleich gut an ein ALT-Contig und an die zugehörige Position in der primären Assembly. ALT-sensibles Mapping mit DRAGEN vermeidet dieses Problem und ermöglicht in Bezug auf ALT-Contigs Verbesserungen bei Sensitivität und Spezifität.

Für ALT-sensibles Mapping sind Hashtabellen erforderlich, die mit spezifizierten ALT-Liftover-Alignments erstellt werden (siehe *ALT-sensible Hashtabellen* auf Seite 132). Wenn eine mit Liftover-Alignments erstellte Hashtabelle bereitgestellt wird, wird DRAGEN automatisch mit ALT-sensiblem Mapping ausgeführt. Legen Sie die Option *--alt-aware* auf „false“ fest, um ALT-sensibles Mapping mit einer Liftover-Referenz zu deaktivieren.

DRAGEN erfordert ALT-sensible Hashtabellen für jede hg19- oder GRCh38-Referenz mit erkannten ALT-Contigs. Wenn Sie die Option *--ht-alt-aware-validate* auf „false“ festlegen, wird diese Anforderung in DRAGEN außer Kraft gesetzt.

Bei aktiviertem ALT-sensiblem Mapping berücksichtigen Mapper und Aligner die Liftover-Beziehung zwischen ALT-Contig-Positionen und zugehörigen Positionen der primären Assembly. Mit Seed-Übereinstimmungen innerhalb von ALT-Contigs werden zugehörige Alignments der primären Assembly abgerufen, auch wenn die Alignments schlechte Scores aufweisen. Es werden Liftover-Gruppen gebildet. Dabei beinhaltet jede Gruppe ein potenzielles Alignment der primären Assembly und null oder mehr potenzielle ALT-Alignments, deren Konvertierung auf die gleiche Position erfolgt. Der Score für jede Liftover-Gruppe wird anhand der Alignments mit der höchsten Übereinstimmung unter Berücksichtigung gepaarter Alignments ermittelt. Der Repräsentant der primären Assembly aus der Liftover-Gruppe mit dem höchsten Score wird als primäres Ausgabe-Alignment verwendet. Die MAPQ wird anhand der Differenz zwischen höchstem Score und dem Score der zweitbesten Liftover-Gruppe berechnet. Die Ausgabe primärer Alignments innerhalb der primären Assembly gewährleistet die normal alignierte Coverage und fördert das Varianten-Calling. Wenn die Option *--Aligner.en-alt-hap-aln* auf 1 festgelegt und *--Aligner.supp-aligns* größer als 0 ist, können auch entsprechende alternative Haplotyp-Alignments, markiert als ergänzende Alignments, ausgegeben werden.

Im Folgenden finden Sie einen Vergleich der Möglichkeiten für die Handhabung von alternativen Haplotypen.

- Mapping ohne ALT-Contigs in der Referenz:
  - Es ergeben sich falsch-positive Varianten-Calls, wenn Reads, die mit einem alternativen Haplotypen übereinstimmen, an anderer Stelle fehlerhaft alignieren.
  - Geringe Sensitivität beim Mapping und Varianten-Calling für Reads, die bei Übereinstimmung mit einem ALT-Contig erheblich von der primären Assembly abweichen.
- Mapping mit ALT-Contigs, jedoch ohne ALT-Sensibilität:
  - Falsch-positive Varianten-Calls von fehlerhaft alignierten Reads, die mit ALT-Contigs übereinstimmen, werden vermieden.
  - Geringe oder keine alignierte Coverage in Regionen der primären Assembly, die durch alternative Haplotypen abgedeckt werden, aufgrund einiger Reads mit Mappings zu ALT-Contigs.
  - Geringe MAPQ oder ein MAPQ-Wert von 0 in Regionen, die durch alternative Haplotypen abgedeckt werden, die der primären Assembly ähneln oder mit dieser identisch sind.
  - Die Sensitivität des Varianten-Callings verringert sich in Regionen, die durch alternative Haplotypen abgedeckt werden, erheblich.
- Mapping mit ALT-Contigs und ALT-Sensibilität:
  - Falsch-positive Varianten-Calls von fehlerhaft alignierten Reads, die mit ALT-Contigs übereinstimmen, werden vermieden.
  - Normal alignierte Coverage in Regionen, die durch alternative Haplotypen abgedeckt werden, da primäre Alignments zur primären Assembly erfolgen.
  - Es werden normale MAPQs zugewiesen, da potenzielle Alignments innerhalb einer Liftover-Gruppe als nicht kompetitiv betrachtet werden.
  - Hohe Sensitivität beim Mapping und Varianten-Calling für Reads, die bei Übereinstimmung mit einem ALT-Contig erheblich von der primären Assembly abweichen.

Sortierung

Das Mapping-Alignment-System generiert eine BAM-Datei, die standardmäßig nach Referenzsequenz und -position sortiert ist. Durch das Erstellen dieser BAM-Datei müssen *samtools sort* oder vergleichbare Befehle zur Nachbearbeitung in der Regel nicht mehr ausgeführt werden. Das Erstellen der BAM-Datei kann mit der

Option *--enable-sort* wie folgt aktiviert bzw. deaktiviert werden:

- Legen Sie zur Aktivierung die Option auf „true“ fest.
- Legen Sie zur Deaktivierung die Option auf „false“ fest.

Auf dem Referenz-Hardwaresystem verlängert sich die Laufzeit für ein Gesamtgenom mit 30-facher Coverage um ca. 6–7 Minuten.

Dublettenkennzeichnung

Das Kennzeichnen oder Entfernen doppelt alignierter Reads ist in der Gesamtgenom-Sequenzierung ein gängiges Verfahren. Bleibt dieser Schritt aus, kommt es möglicherweise zu einer Verzerrung des Varianten-Callings und zu falschen Ergebnissen.

Das DRAGEN-System kann doppelte Reads kennzeichnen oder entfernen und generiert eine BAM-Datei, in der Dubletten im Feld FLAG gekennzeichnet oder komplett entfernt sind.

Wenn Sie beim Testen die Dublettenkennzeichnung aktivieren, verlängert sich die minimale Laufzeit ungefähr um die Dauer, die zur Generierung der sortierten BAM-Datei erforderlich ist. Bei einem 30-fachen Humangesamtgenom werden etwa 1–2 Minuten zusätzlich benötigt. Dies stellt im Vergleich zu den langen Laufzeiten von Open-Source-Tools eine erhebliche Verbesserung dar.

Der Algorithmus für die Dublettenkennzeichnung

Der DRAGEN-Algorithmus für die Dublettenkennzeichnung orientiert sich an der MarkDuplicates-Funktion des Picard-Toolkits. Alle alignierten Reads werden in Untergruppen zusammengefasst, in denen alle Elemente der jeweiligen Untergruppe potenzielle Dubletten sind.

Bei zwei Paaren handelt es sich um Dubletten, wenn die folgenden Bedingungen erfüllt sind:

- Identische Alignment-Koordinaten (Position für Soft oder Hard Clipping vom CIGAR angepasst) an beiden Enden
- Identische Ausrichtungen (Richtung der beiden Enden, beginnend mit der Koordinate am linken Ende)

Zusätzlich kann ein nicht gepaarter Read als Dublette gekennzeichnet werden, wenn dessen Koordinate und Ausrichtung mit einem beliebigen Ende eines anderen Reads, gepaart oder nicht gepaart, übereinstimmen.

Nicht gemappte oder sekundäre Alignments werden unter keinen Umständen als Dubletten gekennzeichnet.

Wenn DRAGEN eine Gruppe Dubletten erkannt hat, wählt es aus der Gruppe einen primären Read aus und kennzeichnet die anderen mit der Markierung für BAM PCR oder optische Dubletten (0x400 oder Dezimal 1024). Für diesen Vergleich werden Dubletten anhand der durchschnittlichen Phred-Sequenzqualität bewertet. Paaren wird die Summe der Scores beider Enden zugewiesen, nicht gepaarte Reads erhalten den Score des einen gemappten Endes. Mit diesem Score sollen, sofern alle anderen Attribute gleich sind, die Reads mit den hochwertigsten Base-Calls beibehalten werden.

Sind die Qualitäts-Scores zweier Reads (oder Paare) gleich, wählt DRAGEN das Paar mit dem höheren Alignment-Score aus. Weisen mehrere Paare auch bei diesem Attribut einen übereinstimmenden Wert auf, wählt DRAGEN das Paar nach dem Zufallsprinzip aus.

Der Score für einen nicht gepaarten Read (R) ist der durchschnittliche Phred-Qualitäts-Score pro Base und wird wie folgt berechnet:

$$Score\,(R) = \frac{\sum_i (R.QUAL[i] \ wobei \ R.QUAL[i] \geq dedup\_min\_qual)}{sequence\_length(R)}$$

Falls es sich bei R um einen BAM-Datensatz handelt, ist QUAL der Array der Phred-Qualitäts-Scores. Bei *dedup-min-qual* handelt es sich um eine DRAGEN-Konfigurationsoption mit dem Standardwert 15. Bei einem Paar ergibt sich der Score aus der Summe der Scores für die beiden Enden.

Dieser Score wird als Ein-Byte-Zahl gespeichert, dabei werden die Werte auf das nächste Viertel abgerundet. Diese Rundung kann zu Dublettenkennzeichnungen führen, die von Picard abweichen. Da die Reads jedoch fast die gleiche Qualität aufgewiesen haben, lässt sich die Auswirkung auf die Ergebnisse des Varianten-Callings vernachlässigen.

Einschränkungen bei der Dublettenkennzeichnung

Die Dublettenkennzeichnung in DRAGEN unterliegt folgenden Einschränkungen:

- Wenn zwei doppelte Reads oder Paare vorliegen, deren Phred-Qualitäts-Scores einander stark ähneln, trifft DRAGEN möglicherweise eine von Picard abweichende Auswahl. Der Einfluss dieser Abweichungen auf die Ergebnisse beim Varianten-Calling ist jedoch vernachlässigbar gering.
- Das ausführbare Programm in DRAGEN akzeptiert nur einfache Bibliotheks-IDs als Befehlszeilenargumente (PGLB). Aus diesem Grund müssen die FASTQ-Dateien vor der Eingabe ins System entsprechend ihrer Bibliotheks-IDs getrennt werden. Zur Unterscheidung nicht doppelt vorliegender Reads ist die Bibliotheks-ID nicht als Kriterium geeignet.

Einstellungen für die Dublettenkennzeichnung

DRAGEN bietet folgende Konfigurationsoptionen für die Dublettenkennzeichnung:

- *--enable-duplicate-marking*

  Legen Sie diese Option auf „true“ fest, um die Dublettenkennzeichnung zu aktivieren. Wenn die Option *--enable-duplicate-marking* aktiviert ist, werden die Ausgabedateien unabhängig vom Wert der Option *enable-sort* sortiert.

- *--remove-duplicates*

  Legen Sie diese Option auf „true“ fest, um die Ausgabe doppelter Datensätze zu unterdrücken. Wenn Sie die Option auf „false“ festlegen, legen Sie im Feld FLAG für doppelte BAM-Datensätze die Option 0x400 fest. Bei Aktivierung der Option *--remove-duplicates* wird die Option *enable-duplicate-marking* automatisch mit aktiviert.

- *--dedup-min-qual*

  Legt den Phred-Qualitäts-Score fest, unterhalb dessen eine Base von der Berechnung des Qualitäts-Scores zur Auswahl von doppelt vorliegenden Reads ausgenommen ist.

Calling kleiner Varianten

Beim DRAGEN-Caller für kleine Varianten handelt es sich um einen Hochgeschwindigkeits-Haplotyp-Caller, für den eine gemischte Hardware-Software-Implementierung erforderlich ist. Bei diesem Ansatz werden eine lokal begrenzte De-novo-Assemblierung in Regionen von Interesse zur Generierung von potenziellen Haplotypen sowie Read-Wahrscheinlichkeitsberechnungen mithilfe eines Hidden Markov Models (HMM) ausgeführt.

Das Varianten-Calling ist standardmäßig deaktiviert. Legen Sie die Option *--enable-variant-caller* auf „true“ fest, wenn Sie das Varianten-Calling aktivieren möchten.

Der Varianten-Caller-Algorithmus

Der DRAGEN-Haplotyp-Caller führt folgende Schritte aus:

- Identifizieren der aktiven Region: Bereiche mit mehreren nicht mit der Referenz übereinstimmenden Reads werden identifiziert und diese Bereiche umgebende Fenster (aktive Regionen) werden für die Verarbeitung ausgewählt.
- Assemblieren lokal begrenzter Haplotypen: In jeder aktiven Region werden alle überlappenden Reads in einem De-Bruijn-Graphen (DBG) zusammengefügt. Der De-Bruijn-Graph ist ein direktionales Diagramm, das auf überlappenden k-mere (Untersequenzen mit der Länge k) in den einzelnen oder in mehreren Reads basiert. Sind alle Reads identisch, ist der DBG linear. Bei Unterschieden bildet der DBG Blasen aus mehreren auseinander- und zusammenlaufenden Pfaden. Wenn die lokale Sequenz zu repetitiv und k zu klein ist, können sich Zyklen bilden, die das Diagramm ungültig machen. Werte von k=10 und 25 werden standardmäßig getestet. Wenn diese Werte zu ungültigen Diagrammen führen, werden die zusätzlichen Werte von k=35, 45, 55, 65 getestet, bis ein zyklusfreies Diagramm vorliegt. Von diesem zyklusfreien DBG wird jeder mögliche Pfad extrahiert, um eine vollständige Liste an potenziellen Haplotypen, d. h. Hypothesen über die echte DNA-Sequenz an mindestens einem Strang, zu generieren.
- Alignieren der Haplotypen: Jeder extrahierte Haplotyp wird nach Smith-Waterman wieder auf das Referenzgenom aligniert, um zu ermitteln, welche Variationen aus der Referenz beinhaltet sind.
- Berechnen der Read-Wahrscheinlichkeit: Jeder Read wird gegen jeden Haplotyp getestet, um die Wahrscheinlichkeit einer Read-Beobachtung abzuschätzen, wobei angenommen wird, dass der Haplotyp aus der echten ursprünglichen DNA-Probe stammt. Diese Berechnung wird mithilfe eines Hidden Markov Model(HMM)-Paars durchgeführt, das sich aus den verschiedenen Möglichkeiten einer Änderung des Haplotyps durch PCR- oder Sequenzierungsfehler im beobachteten Read ergibt. Die HMM-Auswertung berechnet mithilfe einer dynamischen Programmiermethode die Gesamtwahrscheinlichkeit einer Serie an Markov-Zustandsübergängen für den beobachteten Read.
- Genotypisierung: Es werden die möglichen diploiden Kombinationen von Varianten-Ereignissen aus den potenziellen Haplotypen gebildet und für jede davon wird eine bedingte Wahrscheinlichkeit für die Beobachtung des gesamten Read-Pile-ups berechnet. Dabei werden die einzelnen Wahrscheinlichkeiten für die Beobachtung jedes Reads verwendet, die für jeden Haplotyp mithilfe der Auswertung des HMM-Paars bestimmt wurden. Diese werden in die Bayes-Formel eingefügt, um die Wahrscheinlichkeit für die Echtheit aller Genotypen vor dem Hintergrund des gesamten beobachteten Read-Pile-ups zu berechnen. Die Genotypen mit der höchsten Wahrscheinlichkeit werden in den Bericht aufgenommen.

Varianten-Caller-Optionen

Mithilfe der folgenden Optionen kann die Varianten-Caller-Phase der DRAGEN-Hostsoftware gesteuert werden.

- *--enable-variant-caller*

  Legen Sie *--enable-variant-caller* auf *true* fest, um die Varianten-Caller-Phase für die DRAGEN-Pipeline zu aktivieren.

- *--vc-target-bed*

  Diese optionale Befehlszeileneingabe beschränkt die Verarbeitung des Callers für kleine Varianten sowie die zur Target-BED-Datei gehörigen Metriken zu Coverage und Callfähigkeit auf die in einer BED-Datei angegebenen Regionen.

DRAGEN akzeptiert Target-BED-Eingabedateien, in denen die vom Caller für kleine Varianten zu verarbeitenden Regionen und die Metriken zu Coverage/Callfähigkeit angegeben sind. Die Textdatei *--vc-target-bed* umfasst mindestens drei tabulatorgetrennte Spalten, wobei in den ersten drei Spalten Chromosom, Startposition bzw. Endposition aufgeführt sind. Die Positionen haben die Basis null.

Beispiel:

```
# header information
   chr11 0 246920
   chr11 255660 255661
```

- *--vc-target-bed-padding*

  Die Option *--vc-target-bed-padding* ist nicht zwingend erforderlich. Sie kann verwendet werden, um alle Target-BED-Regionen mit dem festgelegten Wert aufzufüllen. Beispiel: Bei einer BED-Region von 1:1000–2000 führt die Verwendung eines Padding-Werts von 100 zum gleichen Ergebnis wie bei einer BED-Region von 1:900–2100 die Verwendung eines Padding-Werts von 0. Alle der Option *--vc-target-bed-padding* hinzugefügten Padding-Werte werden vom Caller kleiner Varianten und den Target-BED-Berichten zu Coverage/Callfähigkeit verwendet. Der Padding-Standardwert ist 0.

- *--vc-sample-name*

  Die Option *--vc-sample-name* ist veraltet. Im kompletten Keimbahn-Modus (mit FASTQ-Eingabe) verwendet der Varianten-Caller den RGSM-Wert als Probenname. Im kompletten somatischen Modus kann mithilfe von *--RGSM-tumor* der Probenname der Tumorprobe festgelegt werden. Im eigenständigen Modus (mit BAM-Eingabe) verwendet der Varianten-Caller den RGSM-Wert aus der BAM-Kopfzeile als Probenname. Im somatischen Modus ist der RGSM-Wert einer Tumor-BAM-Datei mit dem Tumorprobennamen identisch.

- *--vc-target-coverage*

  Die Option *--vc-target-coverage* gibt die Ziel-Coverage für das Downsampling an. Standardwerte: 500 im Keimbahn-Modus und 1000 im somatischen Modus.

- *--vc-enable-gatk-acceleration*

  Ist *--vc-enable-gatk-acceleration* auf „true“ festgelegt, wird der Varianten-Caller im GATK-Modus ausgeführt (in Übereinstimmung mit GATK 3.7 im Keimbahn-Modus und GATK 4.0 im somatischen Modus).

- *--vc-remove-all-soft-clips*

  Ist *--vc-remove-all-soft-clips* auf „true“ festgelegt, werden die Varianten vom Varianten-Caller nicht anhand von Reads mit Soft Clipping bestimmt.

- *--vc-decoy-contigs*

  Mithilfe der Option *--vc-decoy-contigs* kann eine kommagetrennte Liste mit Contigs festgelegt werden, die während des Varianten-Callings übersprungen werden sollen. Diese Option kann in der Konfigurationsdatei festgelegt werden.

- *--vc-enable-decoy-contigs*

  Ist *--vc-enable-decoy-contigs* auf „true“ festgelegt, werden Varianten-Calls bei Decoy-Contigs aktiviert. Die Standardeinstellung ist „false“.

- *--vc-enable-phasing*

  Mithilfe der Option *–vc-enable-phasing* kann die Phasierung von Varianten, sofern möglich, aktiviert werden. Die Standardeinstellung ist „true“.

Downsampling-Optionen für das Calling kleiner Keimbahn-Varianten

Für das Downsampling von Reads in der Pipeline für das Calling kleiner Keimbahn-Varianten stehen folgende Optionen zur Verfügung:

- *--vc-target-coverage* gibt die maximale Anzahl der Reads an, deren Startposition mit einer beliebigen Position überlappt.
- *--vc-max-reads-per-active-region* gibt die maximale Anzahl der Reads an, die eine bestimmte aktive Region abdecken.
- *--vc-max-reads-per-raw-region* gibt die maximale Anzahl der Reads an, die eine bestimmte Rohregion abdecken.
- *--vc-min-reads-per-start-pos* gibt die maximale Anzahl der Reads an, deren Startposition mit einer beliebigen Position überlappt.

Die Optionen für Target-Coverage und max./min. Reads in der Roh-/aktiven Region beziehen sich nicht direkt aufeinander und können unabhängig voneinander ausgelöst werden.

Die Target-Coverage-Option wird zuerst ausgeführt und soll die Anzahl der Reads mit der gleichen Startposition an einer beliebigen Position beschränken. Diese Option beschränkt nicht die Gesamt-Coverage einer bestimmten Position.

Im folgenden Beispiel wird gezeigt, dass der DP aus einem Varianten-Datensatz weit über dem Standardwert für *--vc-target-coverage* (von z. B. 500) liegen kann:

Gehen wir in unserem Beispiel von einem Standardwert für *--vc-target-coverage* von 500 aus. Wenn 400 Reads an Position 1 starten, weitere 400 an Position 2 und wiederum 400 an Position 3, wird die Target-Coverage-Option nicht ausgelöst (da 400 < 500). Wenn an Position 4 eine Variante vorliegt, kann die gemeldete Tiefe bei bis zu 1.200 liegen. Das Beispiel zeigt, dass der DP aus einem Varianten-Datensatz weit über dem Wert für *--vc-target-coverage* liegen kann.

Nach dem Target-Coverage-Schritt liegt die maximale Anzahl der Reads mit der gleichen Position bei 500 (wenn *--vc-target-coverage* auf 500 festgelegt ist).

Beim folgenden Downsampling-Schritt werden die Beschränkungen *--vc-max-reads-per-raw-region* und *--vc-max-reads-per-active-region* angewendet. In diesem Schritt kann die maximale Anzahl an Reads mit der gleichen Position auf unter 500 (Maximalwert aus dem ersten Schritt) reduziert werden. Diese Optionen werden verwendet, um die Gesamtanzahl der Reads in einer kompletten Region mithilfe einer Downsampling-Methode zu beschränken.

Beim Downsampling werden alle Startpositionen ab der Startgrenze der Region gescannt und es wird jeweils ein Read von dieser Position verworfen. Danach wird der Vorgang an der nächsten Position wiederholt, bis die Gesamtanzahl der Reads unterhalb des Schwellenwerts liegt. Es müssen ggf. mehrere Durchgänge über die komplette Region erfolgen, damit die Gesamtanzahl der Reads in der gesamten Region unter den Schwellenwert sinkt. Nach dem Erreichen des Schwellenwerts wird das Downsampling gestoppt, unabhängig davon, welche Position als letzte der Region geprüft wurde.

Wenn die Anzahl der Reads an einer beliebigen Position mit der gleichen Startposition dem Wert für *--vc-min-reads-per-start-pos* entspricht oder darunter liegt, wird diese Position übersprungen. Dadurch wird sichergestellt, dass an jeder Startposition stets eine Mindestanzahl an Reads (festgelegt über *--vc-min-reads-per-start-pos*) gegeben ist.

Beim Downsampling verläuft die Auswahl der zu haltenden und der zu verwendenden Reads weitgehend zufällig. Der Zufallszahlengenerator hat jedoch einen Standardwert als Startwert, um sicherzustellen, dass in jedem Lauf der gleiche Satz an Werten generiert wird. Dadurch werden exakt reproduzierbare Ergebnisse gewährleistet. Es gibt also bei Verwendung der gleichen Eingabedaten keine Variation zwischen Läufen.

Downsampling-Optionen für das Calling kleiner mitochondrialer Varianten

Für das Downsampling von Reads in der Calling-Pipeline kleiner Varianten für das mitochondriale Contig stehen folgende Optionen zur Verfügung:

- *--vc-target-coverage-mito* gibt die maximale Anzahl der Reads an, deren Startposition mit einer beliebigen Position überlappt.
- *--vc-max-reads-per-active-region-mito* gibt die maximale Anzahl der Reads an, die eine bestimmte aktive Region abdecken.
- *--vc-max-reads-per-raw-region-mito* gibt die maximale Anzahl der Reads an, die eine bestimmte Rohregion abdecken.

Der Standardwert für alle drei Optionen ist 40000. Deren Funktion entspricht der unter *Downsampling-Optionen für das Calling kleiner Keimbahn-Varianten* auf Seite 31 beschriebenen. Sie können die Downsampling-Optionen für das mitochondriale Contig unabhängig von den anderen Contigs festlegen, da das mitochondriale Contig eine höhere Tiefe aufweist als alle anderen Contigs in einem WGS-Datensatz.

Phasierung und phasierte Varianten

DRAGEN unterstützt die Ausgabe von Datensätzen für phasierte Varianten in der Keimbahn-VCF- und -gVCF-Datei. Bei der gemeinsamen Phasierung von zwei oder mehr Varianten werden die Phasierungsinformationen in einer Annotation auf Probenebene verschlüsselt (FORMAT/PS). Diese gibt an, in welchem Satz sich die phasierte Variante befindet. Beim angegebenen Wert handelt es sich um eine Ganzzahl, die Auskunft über die Position der ersten phasierten Variante im Satz gibt. Alle Datensätze eines Contigs mit übereinstimmenden PS-Werten gehören zum gleichen Satz.

```
##FORMAT=<ID=PS,Number=1,Type=Integer,Description="Physical phasing ID
   information, where each unique ID within a given sample (but not
   across samples) connects records within a phasing group">
```

Im folgenden Beispiel ist eine DRAGEN-Einzelproben-gVCF mit einer gemeinsamen Phasierung von zwei SNPs dargestellt.

```
chr1 1947645 .       C T,<NICHT_REFERENZ> 48.44 PASS
DP=35;MQ=250.00;MQRankSum=4.983;ReadPosRankSum=3.217;FractionInformativeReads=1.
000;R2_5P_bias=0.000 GT:AD:AF:DP:F1R2:F2R1:GQ:PL:SPL:ICNT:GP:PRI:SB:MB:PS
0|1:20,15,0:0.429:35:9,7,0:11,8,0:47:83,0,50,572,758,622:255,0,255:19,0:4.844e+0
1,8.387e-
05,5.300e+01,4.500e+02,4.500e+02,4.500e+02:0.00,34.77,37.77,34.77,69.54,37.77:11
,9,10,5:12,8,8,7:1947645


chr1 1947648 .       G A,<NICHT_REFERENZ> 50.00 PASS
DP=36;MQ=250.00;MQRankSum=5.078;ReadPosRankSum=2.563;FractionInformativeReads=1.
000;R2_5P_bias=0.000 GT:AD:AF:DP:F1R2:F2R1:GQ:PL:SPL:ICNT:GP:PRI:SB:MB:PS
1|0:16,20,0:0.556:36:8,9,0:8,11,0:48:85,0,49,734,613,698:255,0,255:16,0:5.000e+0
1,7.067e-
05,5.204e+01,4.500e+02,4.500e+02,4.500e+02:0.00,34.77,37.77,34.77,69.54,37.77:10
,6,11,9:8,8,12,8:1947645
```

Während der Genotypisierung werden alle Haplotypen und alle Varianten einer aktiven Region berücksichtigt. Für jedes Variantenpaar gilt: Wenn beide Varianten auf allen identischen Haplotypen auftreten oder wenn eine der Varianten eine homozygote Variante ist, dann erfolgt eine gemeinsame Phasierung. Wenn die

Varianten nur auf unterschiedlichen Haplotypen auftreten, dann erfolgt eine entgegengesetzte Phasierung. Wenn heterozygote Varianten auf einigen, jedoch nicht auf allen identischen Haplotypen auftreten, wird die Phasierung abgebrochen und für die aktive Region werden keine Phasierungsinformationen ausgegeben.

Ploidie-Unterstützung

Der Caller für kleine Varianten unterstützt derzeit nur Ploidie 1 bzw. 2 auf allen Contigs innerhalb der Referenz, ausgenommen das Mitochondrien-Contig, da hierbei eine fortlaufende Allelhäufigkeit zur Anwendung kommt (siehe *Mitochondrien-Calling* auf Seite 33). Die Auswahl von Ploidie 1 bzw. 2 für alle anderen Contigs wird wie folgt festgelegt:

- Wenn *--sample-sex* nicht in der Befehlszeile angegeben wird, werden alle Contigs mit Ploidie 2 verarbeitet.
- Wenn *--sample-sex* in der Befehlszeile angegeben wird, werden alle Contigs wie folgt verarbeitet:
  - Bei weiblichen Proben werden alle Contigs mit Ploidie 2 verarbeitet und Varianten-Calls auf ChrY werden mit dem Filter PloidyConflict gekennzeichnet.
  - Bei männlichen Proben werden alle Contigs außer den Geschlechtschromosomen mit Ploidie 2 verarbeitet. ChrX wird mit Ploidie 1 verarbeitet, ausgenommen die PAR-Regionen, bei denen die Verarbeitung mit Ploidie 2 erfolgt. ChrY wird durchgängig mit Ploidie 1 verarbeitet.

Geschlechtschromosomen werden nach Namenskonvention erkannt, entweder X/Y oder ChrX/ChrY. Andere Namenskonventionen sind nicht zulässig.

Mitochondrien-Calling

In DRAGEN 3.2 wurden gegenüber früheren Versionen umfassende Veränderungen hinsichtlich des Calling-Prozesses für kleine Varianten des mitochondrialen Chromosoms eingeführt.

In früheren Versionen wurde chrM entweder als diploid (wenn in der Befehlszeile kein Geschlecht angegeben wurde) oder haploid (wenn in der Befehlszeile ein Geschlecht angegeben wurde) behandelt. Aufgrund der Beschaffenheit des M-Chromosoms ist weder ein haploides noch ein diploides Modell geeignet, denn eine bestimmte Zelle verfügt über zahlreiche Kopien des haploiden mitochondrialen Chromosoms und diese Kopien der Mitochondrien weisen nicht die exakt gleiche DNA-Sequenz auf. In der Regel sind in jeder Säugetierzelle rund 100 Mitochondrien vorhanden. Jedes Mitochondrium umfasst 2 bis 10 Kopien mitochondrialer DNA (mtDNA). Wenn beispielsweise 20 Prozent der chrM-Kopien über eine Variante verfügen, beträgt die Allelfrequenz (AF) 20 Prozent. Dies wird auch als fortlaufende Allelfrequenz bezeichnet. Es wird eine AF der Varianten von chrM zwischen 0 und 100 Prozent erwartet.

Ab DRAGEN 3.2 wird chrM mit einer Pipeline für fortlaufende AF verarbeitet, die der Pipeline für das Calling somatischer Varianten ähnelt. In diesem Fall wird ein einzelnes ALT-Allel berücksichtigt. Die AF wird geschätzt, es wird ein Wert zwischen 0 und 100 Prozent erwartet.

Im Vergleich zu früheren Versionen ist die Verarbeitung mitochondrialer Chromosomen präziser, denn vor DRAGEN 3.2 wurden Calls mit geringer AF nicht ausgegeben (da eine AF in einem haploiden Modell um 100 Prozent erwartet wurde). In der aktuellen Version können Sie alle ALT-Allel-Varianten auf dem mitochondrialen Chromosom über den gesamten AF-Bereich berücksichtigen (von geringer bis hoher AF).

QUAL wird nicht in den Datensätzen zu chrM-Varianten ausgegeben. Stattdessen erhält der Zuverlässigkeits-Score den Wert INFO/LOD.

```
##INFO=<ID=LOD,Number=1,Type=Float,Description="Variant LOD score">
```

Es wird die Zuverlässigkeit angegeben, mit der an einem bestimmten Locus eine Variante vorhanden ist.

GQ wird nicht in den Datensätzen zu chrM-Varianten ausgegeben, da DRAGEN nicht auf multiple diploide Genotypkandidaten testet. Stattdessen wird ein ALT-Allel als potenzielle Variante gewertet, und wenn INFO/LOD > vc-lod-call-threshold (Standardwert = 4) gilt, wird das Feld FORMAT/GT auf „0/1“ hartcodiert und im Feld FORMAT/AF wird eine Prognose für die Allelfrequenz der Variante mit einem Wert innerhalb von [0,1] angegeben.

Auf mitochondriale Varianten-Calls können folgende Filter angewendet werden.

- *--vc-lod-call-threshold*

  LOD-Schwellenwert für die Ausgabe von Calls in der VCF-Datei. Der Standardwert ist 4.

- *--vc-lod-filter-threshold*

  LOD-Schwellenwert für die Kennzeichnung ausgegebener VCF-Calls als gefiltert. Der Standardwert ist 6.3.

- Wenn INFO/LOD < vc-lod-call-threshold, wird die Variante nicht in die VCF aufgenommen.
- Wenn INFO/LOD > vc-lod-call-threshold und INFO/LOD < vc-lod-filter-threshold, wird die Variante in die VCF aufgenommen, jedoch gilt auch FILTER=lod_fstar.
- Wenn INFO/LOD > vc-lod-call-threshold und INFO/LOD > vc-lod-filter-threshold, wird die Variante in die VCF aufgenommen und es gilt FILTER=PASS.

Im Folgenden finden Sie VCF-Beispieldatensätze für chrM mit einem Call mit sehr hoher AF und einem Call mit sehr geringer AF. In beiden Fällen gilt FORMAT/LOD > emit_threshold. Weiterhin gilt FORMAT/LOD > lod_fstar threshold, sodass die FILTER-Annotation „PASS“ lautet.

```
chrM 2259 . C T . PASS
   DP=9791;MQ=60.00;LOD=38838.40;FractionInformativeReads=0.994
   GT:AD:AF:F1R2:F2R1:DP:SB:MB
   0/1:5,9729:0.999:1,5007:4,4722:9734:3,2,4885,4844:1,4,4807,4922

chrM 16192 . C T . PASS
   DP=9644;MQ=60.00;LOD=26.12;FractionInformativeReads=0.992
   GT:AD:AF:F1R2:F2R1:DP:SB:MB
   0/1:9537,26:0.003:5530,16:4007,10:9563:4484,5053,13,13:4961,4576,19,
```

gVCF- und Joint VCF-Modus

Im gVCF-Modus (für die Keimbahn-Pipeline verfügbar) werden neben den Variantendatensätzen die NICHT_REFERENZ-Regionen ausgegeben. Im Folgenden finden Sie ein Beispiel für eine NICHT_REFERENZ- und Variantenregion-Ausgabe im gVCF-Modus chrM:

```
chrM 751 . A <NICHT_REFERENZ> . PASS END=1437 GT:AD:DP:GQ:MIN_
   DP:PL:SPL:ICNT 0/0:6920,9:6929:99:4077:0,120,1800:0,255,255:40,4

chrM 1438 . A G,<NICHT_REFERENZ> . PASS
   DP=8500;MQ=57.39;LOD=30441.87;FractionInformativeReads=0.871
   GT:AD:AF:F1R2:F2R1:DP:SB:MB
   0/1:0,7400,0:1.000:0,3765,0:0,3635,0:7400:0,0,3994,3406:0,0,3633,3767

chrM 1439 . A <NICHT_REFERENZ> . PASS END=2258 GT:AD:DP:GQ:MIN_
   DP:PL:SPL:ICNT 0/0:6120,10:6130:99:4190:0,120,1800:0,255,255:40,14
```

FORMAT/GT

In NICHT_REFERENZ-Regionen ist FORMAT/GT auf 0/0 hartcodiert und bei Varianten-Loci ist FORMAT/GT auf 0/1 hartcodiert.

FORMAT/GT für chrM wird nicht über FORMAT/AF bestimmt, sondern darüber, ob eine Variante auf einer Position ausgegeben wird oder nicht. Die Entscheidung, ob die Variante ausgegeben wird oder nicht, wird durch den Vergleich des INFO/LOD-Scores mit einem Schwellenwert bestimmt. Alle Varianten mit FORMAT/LOD > emit_threshold werden ausgegeben. Varianten mit FORMAT/LOD < emit_threshold werden nicht in der gVCF ausgegeben und in einer NICHT_REFERENZ-Region mit FORMAT/GT=0/0 zusammengefasst.

Für eine bestimmte Position können folgende zwei Szenarien eintreten.

- Der Genotyper erkennt eine Variante an einer gegebenen Position, wobei FORMAT/LOD > emit_threshold ist. In diesem Fall wird FORMAT/GT auf 0/1 hartcodiert und DRAGEN gibt die für FORMAT/AD, FORMAT/DP und FORMAT/AF an dieser Position berechneten Werte aus.
- An der gegebenen Position wird keine Variante erkannt oder für die erkannte Variante gilt FORMAT/LOD < emit_threshold. In diesem Fall wird FORMAT/GT auf 0/0 hartcodiert und die Position wird mit den zusammenhängenden Positionen zusammengefasst, sofern sich diese im selben Szenario befinden. Alle Positionen mit FORMAT/GT = 0/0 werden gemeinsam zusammengefasst. FORMAT/DP für die Folge wird als mittlerer DP-Wert sämtlicher Positionen in der Folge berechnet. Die FORMAT/AD-Werte für die Folge sind AD-Werte, die an der Position ausgewählt wurden, für die gilt: FORMAT/DP = mittlerer DP. Im Joint-VCF-Modus wird FORMAT/AF anhand von FORMAT/AD berechnet.

Im Folgenden finden Sie Beispiele für einen Variantendatensatz für chrM in einer Trio-Joint-VCF.

Die erste Probe verfügt über eine Variante, wobei FORMAT/FT = lod_fstar, da FORMAT/LOD < lod_fstar_threshold.

Die zweite Probe verfügt nicht über eine Variante, wobei FORMAT/AF=0.

Die dritte Probe verfügt nicht über eine Variante, obwohl gilt: FORMAT/AF>0. Das bedeutet, dass sich die Position für die Probe in einer NICHT_REFERENZ-Region befindet, in der keine Variante mit ausreichender Zuverlässigkeit erkannt wurde.

```
chrM 2622 . G A . . DP=11199;MQ=59.73 GT:AD:AF:DP:FT:LOD:F1R2:F2R1
   0/1:3375,8:0.002:3383:lod_fstar:4.4:1689,2:1686,6
   0/0:5629,1:0:5118:PASS:.:.:. 0/0:3505,8:0.003:2645:PASS:.:.:.
```

QUAL-, QD- und GQ-Formel

In der Einzelproben-VCF und -gVCF wird QUAL gemäß der VCF-Spezifikation (https://samtools.github.io/hts-specs/VCFv4.3.pdf) berechnet.

- QUAL ist die Phred-skalierte Wahrscheinlichkeit, dass eine Stelle nicht über eine Variante verfügt. Der Wert berechnet sich wie folgt:

  ```
  QUAL = -10*log10 (posterior genotype probability of a homozygous-
     reference genotype (GT=0/0))
  ```

  Daraus folgt: QUAL = GP (GT=0/0), wobei GP für die A-posteriori-Wahrscheinlichkeit des Genotyps gemäß Phred-Skala steht.

  QUAL = 20 bezeichnet eine Wahrscheinlichkeit von 99 %, dass an der Stelle eine Variante vorhanden ist. Die GP-Werte werden in der VCF-Datei ebenfalls in der Phred-Skala angegeben.

- GQ ist die Phred-skalierte Call-Fehlerwahrscheinlichkeit.

  GQ=-10*log10(p), wobei p die Wahrscheinlichkeit für einen falschen Call ist.

  GQ=-10*log10(sum(10.^(-GP(i)/10))), wobei die Summe über das unterlegene GT gebildet wird.

  GQ = 3 bedeutet folglich eine Call-Fehlerwahrscheinlichkeit von 50 Prozent, bei GQ = 20 beträgt diese 1 Prozent.

- QD ist der mit der Read-Tiefe DP normalisierte QUAL-Wert.

Metrik	QUAL	GQ	QD
Beschreibung	Wahrscheinlichkeit, dass die Stelle keine Variante aufweist	Call-Fehlerwahrscheinlichkeit	QUAL anhand der Tiefe normalisiert
Formel	QUAL = GP(GT=0/0)	GQ=-10*log10(p)	QUAL/DP
Skala	Phred (ohne Vorzeichen)	Phred (ohne Vorzeichen)	Phred (ohne Vorzeichen)
Beispiel	QUAL=20: Wahrscheinlichkeit von 1 %, dass an der Stelle keine Variante vorhanden ist QUAL=50: Wahrscheinlichkeit von 1 in 1e5, dass an der Stelle keine Variante vorhanden ist	GQ=3, Call-Fehlerwahrscheinlichkeit von 50 % GQ=20, Call-Fehlerwahrscheinlichkeit von 1 %	

Unterschiedlicher Wertebereich zwischen DRAGEN ab Version 2.6 und älteren Versionen von DRAGEN/GATK

Der Bereich der Werte für QUAL (und damit auch GQ und QD) unterscheidet sich bei älteren Versionen von DRAGEN (und/oder GATK) sowie DRAGEN 2.6 und neueren Versionen. Bei älteren DRAGEN-Versionen orientierten sich die QUAL-Berechnungen am GATK-System. Neue DRAGEN-Versionen (ab 2.6) enthalten verbesserte Algorithmen zur Erkennung kleiner Varianten, die realistischere Werte für QUAL (und damit GQ und QD) ergeben (d. h. kleinere Werte als GATK).

Die QUAL-Werte haben sich geändert, da die Wahrscheinlichkeitsberechnungen von GATK und älteren DRAGEN-Versionen davon ausgehen, dass keine readübergreifende Korrelation zwischen Fehlern besteht. Diese Annahme gilt sowohl für Mapping- als auch für Base-Call-Fehler. Da jedoch tatsächlich Korrelationen zwischen Fehlern bestehen, sind die QUAL-Scores viel zu hoch und die Erkennungsleistung suboptimal. Die Algorithmen in DRAGEN 2.6 und neueren Versionen berücksichtigen die Hypothese korrelierter Fehler aus dem Varianten-Caller, woraus sich eine verbesserte Erkennung und realistischere QUAL-Scores ergeben. Die realistischeren QUAL-Scores in DRAGEN-Versionen ab 2.6 sind kleiner als die QUAL-Scores in GATK.

Histogramm von QUAL, QD und GQ

Beim Erstellen des Histogramms der QUAL-Werte (sowie QD- und GQ-Werte) für die Calls einer VCF-Datei von DRAGEN 2.6 oder höher kann bei einem bestimmten Wert (z. B. 50) eine Spitze auftreten. Unterhalb und oberhalb dieses Werts wird ein geringerer Anteil an QUAL-Werten festgestellt. Die Ursache hierfür ist der FRD-Algorithmus (Foreign Read Detection), der einen Grenzwert für den QUAL-Wert (und indirekt für den GQ- und den QD-Wert) von heterozygoten Varianten festlegt. Siehe *Foreign Read Detection* auf Seite 37. Der genaue Wert ist abhängig vom Mapping zwischen dem MAPQ-Höchstwert des Mappers und einer Phred-Zuverlässigkeit hinsichtlich des korrekten Mappings eines Reads. Dieser Grenzwert gilt nicht für homozygote Varianten, da die FRD-Hypothese ausschließlich auf heterozygote Varianten angewendet wird (was wiederum daran liegt, dass die Option *--vc-frd-beta-max* zum Festlegen der maximalen Allelfrequenz für die fremde Read-Hypothese auf 0.5 festgelegt wurde).

Allgemein kann festgehalten werden, dass für den QUAL-Wert heterozygoter Varianten ein Grenzwert vorhanden ist, für den QUAL-Wert homozygoter Varianten jedoch nicht. Die QUAL-Skala bleibt jedoch für alle Varianten (heterozygot oder homozygot) gleich, der einzige Unterschied besteht im Grenzwert.

Modellierung von korrelierten Fehlern über Reads hinweg

Der Varianten-Caller in DRAGEN 2.6 und neueren Versionen verfügt über zwei Algorithmen, die korrelierte Fehler über Reads hinweg in einem bestimmten Pile-up modellieren.

- Die Erkennung fremder Reads (Foreign Read Detection, FRD) erkennt fehlerhaft gemappte Reads. FRD berücksichtigt bei der Wahrscheinlichkeitsberechnung die Möglichkeit eines fehlerhaften Mappings einer Untergruppe der Reads. Statt von individuell je Read auftretenden Mapping-Fehlern auszugehen, wird die Wahrscheinlichkeit geschätzt, dass eine Serie von Reads fehlerhaft gemappt wurde. Evidenzen wie MAPQ und AF-Skew werden einbezogen.
- Der Base Quality Dropoff(BQD)-Algorithmus erkennt korrelierte Base-Call-Fehler: DRAGEN verfügt über einen Mechanismus, der systematische und korrelierte Base-Call-Fehler erkennt, die durch das Sequenzierungsgerät verursacht werden. Der Mechanismus prognostiziert anhand spezifischer Eigenschaften dieser Fehler (Strangverzerrung, Position des Fehlers im Read, Basenqualität) die Wahrscheinlichkeit, dass die Allele das Ergebnis eines systematischen Fehlerereignisses statt einer echten Variante sind.

Die Modellierung korrelierter Fehler führt zu Konfidenz-Score-Werten (QUAL, GQ, QD), die sich in einem realistischen Bereich befinden. Sie sind viel kleiner als die überhöhten Konfidenz-Score-Werte, die von GATK ausgegeben werden.

Foreign Read Detection

Herkömmliche Varianten-Caller behandeln Mapping-Fehler für jeden Read als unabhängige Fehler und ignorieren dabei die Tatsache, dass diese Art von Fehlern üblicherweise als Häufung auftritt. Dies kann zu sehr hohen Zuverlässigkeitsscores trotz geringer MAPQ-Werte und/oder AF-Skew führen. Zur Minimierung dieses Problems können herkömmliche Varianten-Caller einen MAPQ-Schwellenwert in die Berechnung einbeziehen. So werden jedoch wertvolle Evidenzen verworfen und falsch positive Ergebnisse nur unzureichend unterdrückt.

Ab DRAGEN 2.6 steht Foreign Read Detection (FRD) zur Verfügung. Hierdurch wird der vorhandene Genotypisierungsalgorithmus durch die zusätzliche Hypothese erweitert, dass es sich bei verschiedenen Reads im Pile-up um fremde Reads handelt (d. h., dass sich ihre tatsächliche Position an anderer Stelle im Referenzgenom befindet). Der Algorithmus nutzt mehrere Eigenschaften aus (verzerrte Allelfrequenz und geringe MAPQ) und bindet diese Evidenz mathematisch präzise in die Wahrscheinlichkeitsberechnung ein.

Die Sensitivität wird durch gerettete falsch negative Ergebnisse, korrigierte Genotypen und durch die Möglichkeit des Absenkens des MAPQ-Schwellenwerts für eingehende Reads im Varianten-Caller verbessert. Die Spezifität wird durch das Entfernen falsch positiver Ergebnisse und korrigierte Genotypen verbessert.

Bei FRD handelt es sich verglichen mit der Filterung nach den Varianten-Calls um ein leistungsstärkeres Tool. Anstatt nur auffällige Ergebnisse (z. B. aufgrund von Alleltiefe oder Read-Fehlern) nach dem Varianten-Calling zu erkennen, berücksichtigt der Erkennungsalgorithmus durch eine strenge Maximum-Likelihood-Erkennung unmittelbar das Vorhandensein fremder Reads.

Base Quality Dropoff

Herkömmliche Varianten-Caller werden mit der Annahme entwickelt, dass keine Beziehung zwischen Sequenzierungsfehlern einzelner Reads besteht. Diese Annahme vorausgesetzt, ist es extrem unwahrscheinlich, dass an einem bestimmten Locus mehrere identische Fehler auftreten.

Nach der Analyse von NGS-Datensätzen fiel jedoch auf, dass Häufungen von Fehlern wesentlich häufiger auftreten, als bei der Unabhängigkeitsvermutung anzunehmen war. Diese Häufungen können eine hohe Zahl falsch positiver Ergebnisse zur Folge haben.

Glücklicherweise unterscheiden sich diese Fehler in deutlichen Merkmalen von echten Varianten. Beim Base Quality Dropoff(BQD)-Algorithmus handelt es sich um einen Mechanismus, der anhand spezifischer Eigenschaften dieser Fehler (Strangverzerrung, Position des Fehlers im Read, geringe mittlere Basenqualität am Locus von Interesse innerhalb der betreffenden Untergruppe von Reads) eine Wahrscheinlichkeitsberechnung im Genotyper vornimmt.

ROH-Caller

Homozygotie-Regionen (Regions of Homozygosity, ROH) werden als Teil des Callers für kleine Varianten erkannt. Der Caller erkennt die Homozygotie-Läufe von Gesamtgenom-Calls in autosomalen menschlichen Chromosomen und gibt diese aus. Geschlechtschromosomen werden ignoriert. Die ROH-Ausgabe ermöglicht nachgeschalteten Tools das Screening auf eine Blutsverwandtschaft zwischen den Elternteilen des Probanden und eine entsprechende Prognose.

Der ROH-Algorithmus wird bei Calls kleiner Varianten angewendet. Varianten mit Bereichen mit mehreren Allelen, Indels, komplexen Varianten, gefilterten nicht erfolgreichen Calls und homozygoten Referenzbereichen werden ausgeschlossen. Die Varianten-Calls werden dann mithilfe einer Blacklist-BED-Datei weiter gefiltert und nach dem Blacklist-Filter wird schließlich ein Tiefenfilter angewendet. Der Standardwert für den Anteil der gefilterten Calls ist 0.2, wobei die Calls aus den höchsten 10 % und den niedrigsten 10 % der DP-Werte gefiltert werden. Mit den verbliebenen Calls wird dann nach den Regionen gesucht.

Eine Region ist definiert als aufeinanderfolgende Varianten-Calls auf dem Chromosom ohne große Lücke zwischen diesen Varianten. Anders formuliert werden Regionen nach Chromosom oder nach großen Lücken ohne SNV-Calls aufgeschlüsselt. Die Lückengröße ist auf 3 Mbasen festgelegt.

ROH-Optionen

- *--vc-enable-roh*

  Aktivieren bzw. deaktivieren Sie den ROH-Caller, indem Sie diese Option auf „true“ oder „false“ festlegen. Diese Option ist nur für menschliche Autosome standardmäßig aktiviert.

- *--vc-roh-blacklist-bed*

  Der ROH-Caller (sofern bereitgestellt) ignoriert Varianten, die in einer beliebigen Region in der Blacklist-BED-Datei enthalten sind. DRAGEN verteilt Blacklist-Dateien für alle gängigen Humangenome und die Software wählt automatisch eine dem verwendeten Genom entsprechende Blacklist aus, sofern diese Option nicht explizit zur Dateiauswahl verwendet wird.

ROH-Ausgabe

Der ROH-Caller generiert die ROH-Ausgabedatei <Ausgabedateipräfix>.roh.bed, in der jede Zeile für eine Homozygotie-Region steht. Die BED-Datei enthält die folgenden Spalten:

```
Chromosome Start End Score #Homozygous #Heterozygous
```

Wobei gilt:

- „Score“ ist abhängig von der Anzahl der homozygoten und heterozygoten Varianten, wobei gilt, dass jede homozygote Variante den Score um einen vordefinierten Wert erhöht und jede heterozygote Variante den Score um (1 - vordefinierter Wert) verringert. Der vordefinierte Wert muss im Bereich (0, 1) liegen.
- Start- und Endpositionen sind 0-basierte, halboffene Intervalle.

- „#Homozygous“ gibt die Anzahl der homozygoten Varianten in der Region an.
- „#Heterozygous“ gibt die Anzahl der heterozygoten Varianten in der Region an.

Der Caller generiert auch die Metrikdatei <Ausgabedateipräfix>.roh_metrics.csv, in der die Anzahl großer ROH und der Prozentsatz von SNPs in großen ROH (> 3 MB) aufgeführt sind.

Ausgabe der B-Allelfrequenz

Die Ausgabe der B-Allelfrequenz (BAF) ist bei Keimbahn- und somatischen VCF- und gVCF-Läufen standardmäßig aktiviert.

Der BAF-Wert entspricht entweder AF oder (1 – AF), wobei Folgendes gilt:

- AF = (alt_count / (ref_count + alt_count))
- BAF = 1 – AF, nur wenn ref base < alt base, die Basenreihenfolge ist A < T < G < C < N

Für jeden VCF-Eintrag für kleine Varianten mit genau einem SNP-Alternativallel enthält die Ausgabe einen entsprechenden Eintrag in der BAF-Ausgabedatei.

- <NICHT_REFERENZ>-Zeilen sind ausgenommen
  - ForceGT-Varianten (im Feld INFO mit „FGT“ gekennzeichnet) sind in der Ausgabe nicht enthalten, falls die Variante im Feld INFO nicht auch mit „NML“ gekennzeichnet ist.
  - Varianten, bei denen sowohl ref_count als auch alt_count null sind, werden nicht ausgegeben.

BAF-Optionen

--vc-enable-baf

Aktiviert bzw. deaktiviert die Ausgabe der B-Allelfrequenz. Standardmäßig aktiviert.

BAF-Ausgabe

Die generierten BAF sind BigWig-komprimierte Dateien mit Namen <Ausgabedateipräfix>.baf.bw und <Ausgabedateipräfix>.hard-filtered.baf.bw. Die hart gefilterte Datei enthält nur Einträge für Varianten, die die in der VCF definierten Filter passiert haben (d. h. PASS-Einträge).

Jeder Eintrag enthält die folgenden Informationen:

```
Chromosome Start End BAF
```

Wobei Folgendes gilt:

- „Chromosome“ ist eine dem Referenz-Contig entsprechende Zeichenfolge.
- Die Werte für „Start“ und „End“ sind halboffene Intervalle mit der Basis null.
- BAF ist ein Gleitkommawert.

Somatischer Modus

Die somatische DRAGEN-Pipeline ermöglicht die extrem schnelle Analyse von NGS-Daten zur Bestimmung von Mutationen in Zusammenhang mit Krebs. Bei DRAGEN kann das Calling von SNVs und Indels sowohl anhand von zusammengehörigen Tumor-Normal-Paaren als auch anhand von reinen Tumorproben erfolgen.

Bei der Tumor-Normal-Pipeline werden beide Proben gemeinsam analysiert, sodass Keimbahnvarianten ausgeschlossen werden und eine für Tumormutationen spezifische Ausgabe erfolgt. Bei der reinen Tumor-Pipeline wird eine VCF-Datei generiert, die zur Bestimmung von Tumormutationen weiterführend analysiert werden kann. Beide Pipelines setzen keine Ploidie voraus, was die Bestimmung von niedrigen Allelhäufigkeiten ermöglicht.

Die Ausgabe erfolgt nach mehreren Filterungsschritten in Form einer VCF-Datei. Während der Filterungsschritte erfasste Varianten sind in der Ausgabe-VCF enthalten und werden entsprechend mit der Annotation FILTER versehen.

Optionen für den somatischen Modus

Für den somatischen Modus stehen folgende Befehlszeilenoptionen zur Verfügung:

- *--tumor-fastq1* und *--tumor-fastq2*

  Mit den Optionen *--tumor-fastq1* und *--tumor-fastq2* kann ein Paar von FASTQ-Dateien in den Mapper/Aligner und den somatischen Varianten-Caller geladen werden. Diese Optionen können zusammen mit anderen FASTQ-Optionen im Tumor-Normal-Modus verwendet werden. Beispiel:

  ```
  dragen -f -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149 \
     --tumor-fastq1 <TUMOR_FASTQ1> \
     --tumor-fastq2 <TUMOR_FASTQ2> \
     --RGID-tumor <RG0-Tumor> --RGSM-tumor <SM0-Tumor> \
     -1 <NORMAL_FASTQ1> \
     -2 <NORMAL_FASTQ2> \
     --RGID <RG0> –RGSM <SM0> \
     --enable-variant-caller true \
     --output-directory /staging/examples/ \
     --output-file-prefix SRA056922_30x_e10_50M
  ```

- *--tumor-fastq-list*

  Mit der Option *--tumor-fastq-list* kann eine Liste von FASTQ-Dateien in den Mapper/Aligner und den somatischen Varianten-Caller geladen werden. Diese Option kann zusammen mit anderen FASTQ-Optionen im Tumor-Normal-Modus verwendet werden. Beispiel:

  ```
  dragen -f \
     -r /staging/human/reference/hg19/hg19.fa.k_21.f_16.m_149 \
     --tumor-fastq-list <TUMOR_FASTQ_LIST> \
     --fastq-list <NORMAL_FASTQ_LIST> \
     --enable-variant-caller true \
     --output-directory /staging/examples/ \
     --output-file-prefix SRA056922_30x_e10_50M
  ```

- *--tumor-bam-input* und *--tumor-cram-input*

  Mit der Option *--tumor-bam-input* oder *--tumor-cram-input* kann eine zugeordnete BAM- oder CRAM-Datei in den somatischen Varianten-Caller geladen werden. Diese Optionen können zusammen mit anderen BAM-/CRAM-Optionen im Tumor-Normal-Modus verwendet werden.

- *--vc-min-tumor-read-qual*

  Mit der Option *--vc-min-tumor-read-qual* wird die geringste zulässige Read-Qualität (MAPQ) für das Varianten-Calling festgelegt. Der Standardwert ist 20.

Filterung nach somatischem Calling

Für die Filterung nach somatischem Calling stehen folgende Optionen zur Verfügung:

- *--vc-tlod-call-threshold*

  TLOD-Schwellenwert für die Ausgabe von Calls in der VCF-Datei. Der Standardwert ist 3.

- *--vc-tlod-filter-threshold*

  TLOD-Schwellenwert für die Kennzeichnung ausgegebener VCF-Calls als gefiltert. Der Standardwert ist 6.5.
- *--vc-enable-clustered-events-filter*

  Aktiviert den Clusterereignisfilter. Der Standardwert ist „true“.
- *--vc-clustered-events-threshold*

  Legt den Schwellenwert für Clusterereignisse fest. Der Standardwert ist 3.
- *--vc-enable-triallelic-filter*

  Aktiviert den Filter für mehrere Allele. Der Standardwert ist „true“.

Somatischer Modus	Filter-ID	Beschreibung
Tumor-Only & Tumor-Normal	clustered_events	In einer bestimmten aktiven Region wurden Clusterereignisse (≥ 3) beobachtet. Der Schwellenwert für Clusterereignisse kann konfiguriert werden (Standardwert: >= 3).
Tumor-Only & Tumor-Normal	t_lod	Die Variante entspricht nicht dem Wahrscheinlichkeitsschwellenwert (t_lod < 6,5).
Tumor-Only & Tumor-Normal	multiallelic	Gefilterte Stelle, wenn an dieser Position im Tumor mindestens zwei ALT-Allele vorhanden sind.
Tumor-Only & Tumor-Normal	str_contraction	Vermuteter PCR-Fehler, bei dem das ALT-Allel eine Repeat-Einheit weniger als die Referenz aufweist, z. B. ACTACTACT -> ACTACT. Nur aktiv, wenn --vc-enable-gatk-acceleration=true verwendet wird.
Tumor-Only & Tumor-Normal	base_quality	Median der Basenqualität von ALT-Reads an diesem Locus: < 20.
Tumor-Only & Tumor-Normal	mapping_quality	Median der Mappingqualität von ALT-Reads an diesem Locus: < 30.
Tumor-Only & Tumor-Normal	fragment_length	Absolute Differenz zwischen dem Median der Fragmentlänge von ALT-Reads und dem Median der Fragmentlänge von Referenz-Reads an einem bestimmten Locus > 10.000.
Tumor-Only & Tumor-Normal	read_position	Median der Abstände zwischen Anfang/Ende des Reads und einem bestimmten Locus > 5. (Die Variante befindet sich zu nah am Rand aller Reads.)
Tumor-Only & Tumor-Normal	panel_of_normals	In mindestens einer Probe in der Normalgruppen-VCF festgestellt.

Filterung nach somatischem Calling bei Vorhandensein einer übereinstimmenden Normalkontrollprobe

Somatischer Modus	Filter-ID	Beschreibung
Tumor-Normal	germline_risk	Wahrscheinlichkeit für das Vorhandensein eines Allels in der Normalprobe > 0,025.
Tumor-Normal	artifact_in_normal	TLOD des Normal-Read-Satzes (Normal-Artefakt-LOD) > 0,0. Wird nicht als Normal-Artefakt klassifiziert, wenn die Allelfraktion in der Normalprobe wesentlich kleiner ist als die Allelfraktion in der Tumorprobe (normalAlleleFraction < 0,1 * tumorAlleleFraction).

QUAL wird nicht in den Datensätzen zu somatischen Varianten ausgegeben. Stattdessen erhält der Zuverlässigkeits-Score den Wert INFO/TLOD.

```
##INFO=<ID=TLOD,Number=A,Type=String,Description="Tumor LOD score">
```

Quantifiziert die Evidenz des Vorhandenseins eines ALT-Allels an einem bestimmten Locus in der Tumorprobe.

Wenn zusätzlich eine Normalprobe vorhanden ist, gibt DRAGEN den Zuverlässigkeits-Score ebenfalls als INFO/NLOD aus.

```
##INFO=<ID=NLOD,Number=A,Type=String,Description="Normal LOD score">
```

Quantifiziert die Evidenz dafür, dass die Normalprobe an einem bestimmten Locus eine homozygote Referenz bildet.

GQ wird nicht in den Datensätzen zu somatischen Varianten ausgegeben, da DRAGEN nicht auf mehrere diploide Genotypkandidaten testet. Stattdessen wird ein ALT-Allel als Kandidat für eine somatische Variante gewertet und wenn INFO/TLOD > vc-tlod-call-threshold (Standardwert = 3), wird das Feld FORMAT/GT für die Tumorprobe auf „0/1“ hartcodiert und im Feld FORMAT/AF wird eine Prognose für die Allelfrequenz der somatischen Variante mit einem Wert innerhalb von [0,1] angegeben.

- Wenn INFO/TLOD < vc-tlod-call-threshold, wird die Variante nicht in die VCF aufgenommen.
- Wenn INFO/TLOD > vc-tlod-call-threshold und INFO/TLOD < vc-tlod-filter-threshold, wird die Variante in die VCF aufgenommen, jedoch gilt auch FILTER=t_lod.
- Wenn INFO/TLOD > vc-tlod-call-threshold und INFO/TLOD > vc-tlod-filter-threshold, wird die Variante in die VCF aufgenommen und es gilt FILTER=PASS.
- Der Standardwert für vc-tlod-filter-threshold ist 6.5.

Im Folgenden ist ein Beispiel für einen somatischen T/N-VCF-Datensatz dargestellt. INFO/TLOD > vc-tlod-call-threshold und INFO/TLOD < vc-tlod-filter-threshold, daher ist der FILTER mit „t_lod“ gekennzeichnet.

```
2 593701 . G A . t_lod
   DP=97;MQ=48.74;TLOD=3.86;NLOD=9.83;FractionInformativeReads=1.000
   GT:AD:AF:F1R2:F2R1:DP:SB:MB 0/0:33,0:0.000:14,0:19,0:33
   0/1:61,3:0.047:29,2:32,1:64:35,26,0,3:39,22,1,2
```

gVCF, Combine gVCF und Joint VCF

Beim Joint-Calling handelt es sich um einen Modus, bei dem ein gemeinsamer Analyseschritt die Sensitivität der Variantenbestimmung mithilfe populationsweiter Informationen aus einer Kohorte mit mehreren Proben erhöht. Außerdem ermöglicht das Joint-Calling mit einer Stammbaumdatei das De-novo-Trio-Calling für den Probanden anhand von Familiendaten. Joint-Calling steht nur für die Keimbahn-Pipeline zur Verfügung.

Allgemeine Beschreibung des Joint-Callings kleiner Varianten:

Wie auch bei CNV und SV handelt es sich beim Joint-Calling kleiner Varianten um einen Vorgang in zwei Schritten:

1 Generieren Sie eine Einzelproben-gVCF für alle Proben in der Kohorte.

2 Führen Sie den Joint-Genotyper mit *--enable-joint-genotyping true* und den drei einzelnen gVCF-Dateien als Eingabe aus. Die Ausgabedatei ist eine gemeinsame VCF-Datei.

Allgemeine Beschreibung des De-novo-Joint-Callings kleiner Varianten:

Wie auch bei CNV und SV handelt es sich beim De-novo-Trio-Calling kleiner Varianten um einen Vorgang in zwei Schritten:

3 Generieren Sie eine Einzelproben-gVCF für die drei Proben des Trios.

Es wird empfohlen, die Option *--vc-emit-ref-confidence* gemeinsam mit der Banding-Option zu verwenden.

BP RESOLUTION führt zu wesentlich längeren Laufzeiten sowie zu einer vollen Basenpaarauflösung (große Dateien).

Der Banding-Vorgang optimiert sowohl die Laufzeit als auch die Dateigröße bei minimaler Beeinträchtigung der Genauigkeit. Das Banding ist standardmäßig aktiviert.

1 Führen Sie den Joint-Genotyper mit *--enable-joint-genotyping true --pedigree-file <pedigree.ped>* und den drei einzelnen gVCF-Dateien als Eingabe aus. Die Ausgabedatei ist eine gemeinsame VCF-Datei, in der dem Probanden De-novo-Calls und ein zugehöriger DQ-Score zugeordnet sind.

CNV, SV und small SV lassen sich mit DRAGEN folgendermaßen parallel für ein Trio ausführen:

- Führen Sie in derselben DRAGEN-Befehlszeile einen Einzelprobenlauf für small VC (im gVCF-Modus), CNV und SV für jede Probe aus.
- Führen Sie das De-novo-Joint-Calling für kleine Varianten aus.
- Führen Sie das De-novo-Joint-Calling für CNV aus.
- Führen Sie das De-novo-Joint-Calling für SV aus.

Insgesamt sechs DRAGEN-Befehlszeilen ergeben drei gemeinsame De-novo-VCFs als Ausgabe (small VC, CNV und SV).

Der Schritt combine gVCF ist optional. Er ist nur erforderlich, wenn eine gemeinsame gVCF-Datei als Ausgabe des Joint-Genotypers benötigt wird. Wenn der Joint-Genotyper lediglich eine gemeinsame VCF-Datei ausgeben muss, kann dieser Schritt übersprungen werden.

Der einzige Unterschied zwischen einer gemeinsamen gVCF-Datei und einer gemeinsamen VCF-Datei besteht darin, dass die gemeinsame gVCF-Datei die NICHT_REFERENZ-Blöcke aus den Einzelproben-gVCF-Dateien enthält. Wenn der Schritt combine gVCF erforderlich ist, muss er zwischen den Befehlszeilen für die Einzelproben-gVCF und der Joint-Genotyper-Befehlszeile ausgeführt werden, in der der Joint-Genotyper die kombinierte gVCF-Datei als Eingabe erhält.

Diese Schritte werden im Folgenden beschrieben:

gVCF, combine gVCF und Joint-Calling erfolgen in einem mehrstufigen Prozess. Zunächst wird für jede Person eine gVCF-Datei erstellt. Bei der gVCF handelt es sich um eine erweiterte VCF, die nicht nur Informationen zu den Positionen mit Varianten enthält, sondern auch zu Positionen, bei denen es sich um eine homozygote Referenz handelt. Die gVCF ist u. U. jedoch nicht fortlaufend, d. h. es bestehen Lücken zwischen den in der gVCF aufgeführten Regionen. Es stehen zusätzliche Informationen zur Verfügung, die die Evidenz (Reads) für das Nichtvorhandensein von Varianten (Referenz) oder alternativen Allelen angeben.

Die DRAGEN-gVCF ist nicht notwendigerweise fortlaufend und kann Lücken enthalten, die nicht callfähigen Regionen entsprechen. Diese Regionen werden nicht in der gVCF ausgegeben.

Eine Region ist nur callfähig, wenn sie das folgende Kriterium erfüllt:

- Mindestens ein Read ist mit MAPQ>0 gemappt.

Beispiele für nicht callfähige Regionen:

- Es sind keine Reads auf die Region gemappt.
- Es sind mehrere Reads auf die Regionen gemappt, jedoch gilt für alle Reads MAPQ=0.

DRAGEN ermittelt callfähige Regionen vor dem Ausführen des Varianten-Callers. Daher sind Lücken in den callfähigen Regionen auch in der gVCF vorhanden, da nur callfähige Regionen den VC durchlaufen.

Wenn beim Joint-Calling eine Position für einige Proben in einer gVCF-Lücke liegt und in einem homref-Block für andere Proben, wird die Position in der gemeinsamen gVCF als homref-Block klassifiziert, jedoch enthält das Feld FORMAT von Proben mit Lücken ausschließlich Punkte („./.:.:.“), d. h. keinen Call. Diese Position wird nicht in die gemeinsame VCF aufgenommen, da für keine der Personen Varianten-Calls vorliegen.

Im Folgenden finden Sie ein Beispiel für eine Position in der gemeinsamen VCF, bei der für eine Person eine Variante vorliegt und sich bei den beiden anderen Proben dieselbe Position in einer gVCF-Lücke befindet (das Feld FORMAT enthält ausschließlich Punkte):

```
1       605262  .       G       A       13.41   DRAGENHardQUAL
AC=2;AF=1.000;AN=2;DP=2;FS=0.000;MQ=14.00;QD=6.70;SOR=0.693
GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GP       ./.:.:.:.:.:LowDepth
1/1:0,2:1.000:2:4:PASS:0,0:0,2:50,6,0:1.383e+01,4.943e+00,1.951e+00
./.:.:.:.:.:LowDepth
```

Für den nächsten Schritt können Sie eine der folgenden Optionen auswählen:

- Combine gVCF verwendet mehrere gVCF-Dateien als Eingabe und generiert eine kombinierte gVCF-Datei, die Daten aller Eingabeproben enthält. Die kombinierte gVCF-Datei kann an den Joint-Caller gesendet werden.
- Führen Sie den Joint-Caller direkt mit mehreren gVCF-Dateien als Eingabe aus. Die gVCF-Dateien werden gemeinsam genotypisiert, um eine einzige VCF zu generieren. Die gemeinsame VCF enthält mehrere Genotypspalten, für jede Probe in der Kohorte jeweils eine.

gVCF-Optionen

Zusätzlich zu den Standardparametern für die Varianten-Caller-Phase der DRAGEN-Hostsoftware stehen für die gVCF-Erstellung folgende Parameter zur Verfügung:

- *--vc-emit-ref-confidence*

  Zur Aktivierung der gVCF-Erstellung bei der Basenpaarauflösung legen Sie die Option *--vc-emit-ref-confidence* auf BP_RESOLUTION fest. Zur Aktivierung von gVCF-Erstellung mit Banding auf GVCF festlegen.

- *--vc-gvcf-gq-bands*

  Die Option *--vc-gvcf-gq-bands* kann optional für die Definition von GQ-Folgen für die gVCF-Ausgabe verwendet werden. Der Standardwert lautet 10,20,30,40,60,80.

- *--vc-max-alternate-alleles*

  Über die Option *--vc-max-alternate-alleles* wird die maximale Anzahl von ALT-Allelen festgelegt, die in einer VCF- oder gVCF-Datei ausgegeben werden. Der Standardwert ist 6.

Optionen für Combine gVCF

Combine gVCF kann nach dem Erstellen der gVCF-Dateien aufgerufen werden. Das Tool Combine gVCF dient zum Zusammenführen aller gVCF-Eingabedateien in einer gVCF-Datei.

Für Combine gVCF erfordert die DRAGEN-Hostsoftware folgende Optionen.

- *--enable-combinegvcfs*

  Legen Sie diese Option auf „true“ fest, wenn Sie gVCF-Dateien zusammenführen möchten.

- *--output-directory*

  Gibt das Ausgabeverzeichnis an.

- *--output-file-prefix*

  Gib das Präfix an, mit dem alle Ausgabedateien des Laufs gekennzeichnet werden.

- *-r*

  Gibt das Verzeichnis an, in dem die Hashtabelle gespeichert wird.

- *--variant*, *--variant-list*

  Gibt den Pfad zu einer einzelnen gVCF-Datei an. In der Befehlszeile können mehrere *--variant*-Optionen verwendet werden, eine für jede gVCF-Datei. Es werden bis zu 500 gVCF-Dateien unterstützt.
  Mit der Option *--variant-list* kann ein Pfad zu einer Datei angegeben werden, die eine Liste der zusammenzuführenden gVCF-Eingabedateien enthält, jeweils eine Variantendatei pro Zeile.

Joint-Calling

Das Joint-Calling kann aufgerufen werden, nachdem die erforderlichen gVCF-Dateien in der Keimbahn-Pipeline erstellt wurden. Die DRAGEN-Hostsoftware erfordert folgende Optionen für das Joint-Calling.

- *--enable-joint-genotyping*

  Legen Sie diese Option auf „true“ fest, wenn Sie gVCF-Dateien zusammenführen möchten.

- *--output-directory*

  Gibt das Ausgabeverzeichnis an.

- *--output-file-prefix*

  Gib das Präfix an, mit dem alle Ausgabedateien des Laufs gekennzeichnet werden.

- *-r*

  Gibt das Verzeichnis an, in dem die Hashtabelle gespeichert wird.

- *--variant* oder *--variant-list*

  Gibt den Pfad zu einer einzelnen gVCF-Datei an. In der Befehlszeile können mehrere *--variant*-Optionen verwendet werden, eine für jede gVCF-Datei. Es werden bis zu 500 gVCF-Dateien unterstützt. Mit der Option *--variant-list* kann eine Datei angegeben werden, die eine Liste der zusammenzuführenden gVCF-Eingabedateien enthält, jeweils eine Variantendatei pro Zeile.

Die folgenden Optionen sind optional:

- *--pedigree-file*

  Die Option *--pedigree-file* gibt den Pfad zu einer PED-Stammbaumdatei an, die eine strukturierte Beschreibung der familiären Beziehungen zwischen den Proben enthält. Mithilfe dieser Option kann der gemeinsame Caller die Stammbaumdaten in die Analyse aufnehmen. Derzeit werden nur Stammbaumdateien mit Trios (Mutter, Vater, Kind) unterstützt. Mehrere Sätze Trios werden unterstützt, z. B. mehrere Kinder. Die Stammbaumdatei muss tabulatorgetrennt sein (leerzeichengetrennte Dateien werden nicht unterstützt) und die Erweiterung .ped aufweisen.

  Beispiel:

```
#Family_ID Individual_ID   Paternal_ID    Maternal_ID    Sex Phenotype
FAM001     NA12877_Father  0              0              1   1
FAM001     NA12878_Mother  0              0              2   1
FAM001     NA12882_Proband NA12877_Father NA12878_Mother 2   2
FAM001     NA12883_Proband NA12877_Father NA12878_Mother 1   0
```

Die einzelnen Spalten enthalten Folgendes:

- Spalte 1: Stammbaum-ID
- Spalte 2: ID der Person
- Spalte 3: Vater der Person (0 = Gründer)
- Spalte 4: Mutter der Person (0 = Gründer)
- Spalte 5: Geschlecht (1 = männlich, 2 = weiblich)
- Spalte 6: genetische Daten (0 = unbekannt, 1 = nicht betroffen, 2 = betroffen)

- *--enable-multi-sample-gvcf*

  Legen Sie *--enable-multi-sample-gvcf* auf „true“ fest, wenn der gemeinsame Caller eine Mehrproben-gVCF-Datei ausgeben soll. Bei dieser Option muss die Eingabe als kombinierte gVCF-Datei erfolgen.

Tabelle 2 Modi für das Joint-Calling, Eingabedateien und Befehlszeilenoptionen

Zu generierende Datei	Mehrproben-gVCF mit Joint-Calling (Bevölkerungsgruppe)	Mehrproben-gVCF mit Joint-Calling (Familie)	Mehrproben-VCF mit Joint-Calling (Bevölkerungsgruppe)	Mehrproben-VCF mit Joint-Calling (Familie)
Eingabedatei	kombinierte Mehrproben-gVCF-Datei	kombinierte Mehrproben-gVCF-Datei	kombinierte Mehrproben-gVCF-Datei oder X einzelne gVCF-Dateien	kombinierte Mehrproben-gVCF-Datei oder X einzelne gVCF-Dateien
Stammbaumdatei verwenden	Nein	Ja	Nein	Ja
Befehlszeilenoptionen	--enable-joint-genotyping true --enable-multi-sample-gvcf TRUE	--enable-joint-genotyping true --enable-multi-sample-gvcf TRUE --pedigree-file file.ped	--enable-joint-genotyping true	--enable-joint-genotyping true --pedigree-file file.ped

In den gemeinsamen VCF-Ausgabedatensätzen weisen Proben, für die FORMAT GT = „0/0“ gilt, keine Variante an dieser Position auf und werden daher möglicherweise mit benachbarten Positionen in einer Folge zusammengefasst, bei denen es sich ebenfalls um eine homozygote Referenz handelt (oder die Evidenz zu gering für das Calling einer Variante ist).

Bei zusammengefassten Positionen haben FORMAT/AD, FORMAT/AF und FORMAT/DP folgende spezielle Bedeutungen:

- FORMAT/DP für die Folge wird als minimale DP sämtlicher Positionen in der Folge berechnet.
- Die FORMAT/AD-Werte für die Folge sind AD-Werte, die an der Position in der Folge ausgewählt wurden, für die gilt: DP = mittlerer DP.
- FORMAT/AF wird anhand von FORMAT/AD berechnet.

Beispiel:

```
chr1      10230   .             AC         A            66.08     PASS
 AC=2;AF=0.333;AN=6;DP=767;FS=2.949;MQ=30.17;MQRankSum=-
0.808;QD=0.19;ReadPosRankSum=-1.942;SOR=0.866
GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP
0/1:108,65:0.376:173:19:PASS:63,35:45,30:50,0,45:-5.006,0,-4.51:1.911e+01,5.366e-
02,4.815e+01
0/1:114,70:0.380:184:18:PASS:68,35:46,35:48,0,45:-4.825,0,-
```

```
4.486:1.831e+01,6.462e-02,4.792e+01
0/0:223,74:0.249:297:0:LowGQ:.:.:0,0,3318:.:.
```

De-novo-Joint-Calling

Wenn in der DRAGEN-Befehlszeile zum Joint-Calling eine Stammbaumdatei verwendet wird (*--pedigree-file file.ped*), die die familiären Beziehungen zwischen den Proben angibt, bestimmt DRAGEN beim Probanden alle Varianten mit Vererbungskonflikt und berechnet einen zugehörigen De-novo-Qualitäts-Score (DQ), der die Wahrscheinlichkeit dafür angibt, dass es sich bei der Variante um eine De-novo-Mutation handelt. Je höher der Score, desto höher ist die Wahrscheinlichkeit für das Vorliegen einer De-novo-Variante. Für den Fall, dass in der Stammbaumdatei mehrere Trios angegeben sind (z. B. ein Stammbaum mit mehreren Generationen), erkennt DRAGEN die Trios automatisch und bestimmt die De-novo-Varianten in der Probandenprobe für jedes Trio.

Beim Ausgabeformat für die De-novo-Informationen gibt es geringfügige Abweichungen zwischen der Joint-VCF vor Filterung und der Joint-VCF nach Filterung.

Joint-VCF vor Filterung

In der Joint-VCF vor Filterung werden sämtliche Probanden-Varianten mit Vererbungskonflikt mit FORMAT/DN als De-novo-Varianten gekennzeichnet und erhalten einen FORMAT/DQ-Score. Sämtliche normal von den Eltern geerbten Probanden-Varianten werden im Feld FORMAT/DN als „Inherited“ gekennzeichnet.

Das Feld FORMAT/FT erhält den Wert „PASS“ oder eine Annotation, abhängig davon, ob die Einzelprobenvariante als „PASS“ gilt oder nicht. Der FILTER-Status wird mit „.“ gekennzeichnet.

Die folgenden Beispiele zeigen einen Variantendatensatz aus einer Joint-VCF vor Filterung, bei der das Calling mithilfe einer Stammbaumdatei erfolgt ist. Die Probanden-Variante ist als De-novo-Variante gekennzeichnet. In diesem Fall sind beide Elternteile eine homozygote Referenz mit hoher Zuverlässigkeit und das Kind ist heterozygot ALT. Die FORMAT-Reihenfolge ist Proband/Vater/Mutter.

```
chr1    861154  .       T       C       27.62   .
       AC=1;AF=0.167;AN=6;DP=61;FS=0.000;MQ=41.44;MQRankSum=-
0.617;QD=0.84;ReadPosRankSum=3.574;SOR=0.850
       GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
       0/1:21,12:0.364:33:31:PASS:11,5:10,7:66,0,45:-6.635,0,-4.45:3.159e+01,3.092e-
03,4.750e+01:24,0,83:6.8575e-04:DeNovo
       0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:0,26,253
       0/0:16,1:0.059:17:6:PASS:.:.:0,6,592:.:.:22,0,254


chr1    234710899       .       T       C       44.74   .

AC=1;AF=0.167;AN=6;DP=73;FS=4.720;MQ=250.00;MQRankSum=5.310;QD=1.15;ReadPosRankS
um=1.366;SOR=0.251
       GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
0/1:21,18:0.462:39:48:PASS:14,10:7,8:84,0,50:-8.427,0,-5:4.950e+01,7.041e-
05,5.300e+01:15,0,120:3.2280e-01:DeNovo
       0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:10,0,227
       0/0:25,0:0.000:22:60:PASS:.:.:0,60,899:.:.:0,33,227
```

Joint-VCF nach Filterung

In der Joint-VCF nach Filterung erhalten sämtliche Probanden-Varianten im Vererbungskonflikt einen FORMAT/DQ-Score. Alle derartige Varianten mit FORMAT/DQ >= DQ-Schwellenwert werden im Feld FORMAT/DN mit „DeNovo“ gekennzeichnet. Die Varianten mit FORMAT/DQ < DQ-Schwellenwert werden im Feld FORMAT/DN mit „LowDQ“ gekennzeichnet. Sämtliche normal von den Eltern geerbten Probanden-Varianten werden im Feld FORMAT/DN als „Inherited“ gekennzeichnet.

Das Feld FORMAT/FT erhält den Wert „PASS“ oder eine Annotation, abhängig davon, ob die Einzelprobenvariante als „PASS“ gilt oder nicht. Der FILTER-Status erhält den Wert „PASS“ oder eine Annotation, abhängig davon, ob die Joint-QUAL größer ist als ein Schwellenwert oder nicht.

Im Folgenden finden Sie die Datensätze für die Joint-VCF nach Filterung. Diese wurden anhand derselben Beispiele erstellt wie die Datensätze für die Joint-VCF vor Filterung. In diesem Fall wird der erste Datensatz statt mit „DeNovo“ mit „LowDQ“ gekennzeichnet, da FORMAT/DQ unter dem Schwellenwert liegt. Die FORMAT-Reihenfolge ist Proband/Vater/Mutter.

```
chr1    861154  .       T       C       27.62   DRAGENHardQUAL
       AC=1;AF=0.167;AN=6;DP=61;FS=0.000;MQ=41.44;MQRankSum=-
0.617;QD=0.84;ReadPosRankSum=3.574;SOR=0.850
       GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
       0/1:21,12:0.364:33:31:PASS:11,5:10,7:66,0,45:-6.635,0,-4.45:3.159e+01,3.092e-
03,4.750e+01:24,0,83:6.8575e-04:LOWDQ
       0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:0,26,253
       0/0:16,1:0.059:17:6:PASS:.:.:0,6,592:.:.:22,0,254


chr1    234710899       .       T       C       44.74   PASS

AC=1;AF=0.167;AN=6;DP=73;FS=4.720;MQ=250.00;MQRankSum=5.310;QD=1.15;ReadPosRankS
um=1.366;SOR=0.251
       GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
       0/1:21,18:0.462:39:48:PASS:14,10:7,8:84,0,50:-8.427,0,-5:4.950e+01,7.041e-
05,5.300e+01:15,0,120:3.2280e-01:DeNovo
       0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:10,0,227
       0/0:25,0:0.000:22:60:PASS:.:.:0,60,899:.:.:0,33,227
```

De-novo-Metriken

Der Bericht mit den VC-Metriken enthält eine Berechnung der Anzahl aller De-novo-Varianten (SNP und INDEL) über einem standardmäßigen DQ-Schwellenwert. Diese Schwellenwerte können mit folgenden Optionen festgelegt werden:

- *--qc-snp-DeNovo-quality-threshold* <Wert>, Standardwert = 0.05
- *--qc-indel-DeNovo-quality-threshold* <Wert>, Standardwert = 0.02

Diese Schwellenwerte wirken sich auf die in den VC-Metriken enthaltene Anzahl an De-novo-Varianten sowie den Inhalt der Joint-VCF nach Filterung aus. (Nur Probanden-Varianten mit Vererbungskonflikt und FORMAT/DQ > DQ-Schwellenwert werden im Feld FORMAT/DN mit „DeNovo“ gekennzeichnet.) Die Standardschwellenwerte liefern ein ausgewogenes Verhältnis zwischen Spezifität und Sensitivität, können bei Bedarf jedoch geändert werden.

Für den Fall, dass in der Stammbaumdatei mehrere Trios angegeben sind (z. B. ein Stammbaum mit mehreren Generationen), erkennt DRAGEN die Trios automatisch und bestimmt die De-novo-Varianten in der Probandenprobe für jedes Trio.

De-novo-Variantenfilterung

Der Filterungsschritt identifiziert De-novo-Varianten-Calls des Joint-Calling-Workflows in Regionen mit Änderungen des Ploidiegrads. Da das De-novo-Calling in Regionen, in denen mindestens ein Stammbaummitglied nicht diploide Genotypen aufweist, eine reduzierte Spezifität haben kann, kennzeichnet die De-novo-Variantenfilterung relevante Varianten und kann somit die Spezifität des Call-Satzes erhöhen.

Das FORMAT/DN-Feld in der Probandenvariante wird auf Grundlage der strukturellen und Kopienzahlvarianten-Calls des Stammbaums vom ursprünglichen DeNovo-Wert in „DeNovoSV“ oder „DeNovoCNV“ geändert, sofern die De-novo-Variante mit einer den Ploidiegrad verändernden SV oder CNV überlappt. Alle anderen Variantendetails bleiben unverändert. Alle Varianten der Eingabe-VCF sind auch in der gefilterten Ausgabe-VCF vorhanden. Strukturelle oder Kopienzahlvarianten, die den Ploidiegrad nicht verändern (z. B. Inversionen), werden bei der Filterung nicht berücksichtigt. Im folgenden Beispiel ist ein De-novo-SNV-Call in der Eingabe-VCF aufgeführt:

```
chr1 234710899 . T C 44.74 PASS
AC=1;AF=0.167;AN=6;DP=73;FS=4.720;MQ=250.00;MQRankSum=5.310;QD=1.15;Rea
dPosRankSum=1.366;SOR=0.251
GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
0/1:21,18:0.462:39:48:PASS:14,10:7,8:84,0,50:-8.427,0,-
5:4.950e+01,7.041e-05,5.300e+01:15,0,120:3.2280e-01:DeNovo
0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:10,0,227
0/0:25,0:0.000:22:60:PASS:.:.:0,60,899:.:.:0,33,227
```

Bei Überlappung mit einem SV-Duplikat in der Probandenvariante werden Mutter oder Vater in der gefilterten Ausgabe-VCF wie folgt dargestellt:

```
chr1 234710899 . T C 44.74 PASS
AC=1;AF=0.167;AN=6;DP=73;FS=4.720;MQ=250.00;MQRankSum=5.310;QD=1.15;Rea
dPosRankSum=1.366;SOR=0.251
GT:AD:AF:DP:GQ:FT:F1R2:F2R1:PL:GL:GP:PP:DQ:DN
0/1:21,18:0.462:39:48:PASS:14,10:7,8:84,0,50:-8.427,0,-
5:4.950e+01,7.041e-05,5.300e+01:15,0,120:3.2280e-01:DeNovoSV
0/0:13,0:0.000:11:30:PASS:.:.:0,30,450:.:.:10,0,227
0/0:25,0:0.000:22:60:PASS:.:.:0,60,899:.:.:0,33,227
```

Im folgenden Beispiel einer Befehlszeile wird die De-novo-Filterung basierend auf den Dateien ausgeführt, die von den Joint-Calling-Workflows zurückgegeben werden:

```
dragen \
   --dn-enable-denovo-filtering true \
   --dn-input-joint-vcf <JOINT_KLEINE_VARIANTEN_VCF> \
   --dn-output-joint-vcf <AUSGABE-VCF> \
   --dn-sv-vcf <JOINT_SV-VCF> \
   --dn-cnv-vcf <JOINT_CNV_VCF> \
   --enable-map-align false
```

Optionen für die De-novo-Variantenfilterung

Folgende Optionen werden für die De-novo-Variantenfilterung verwendet:

- *--dn-input-vcf*: zu filternde Joint-VCF mit kleinen Varianten aus dem De-novo-Calling-Schritt.
- *--dn-output-vcf*: Dateispeicherort, an dem die gefilterte VCF gespeichert werden soll. Wenn nicht angegeben, wird die VCF-Eingabedatei überschrieben.
- *--dn-sv-vcf*: Joint-VCF mit strukturellen Varianten aus dem SV-Calling-Schritt. Wenn ausgelassen, werden Prüfungen mit überlappenden strukturellen Varianten übersprungen.
- *--dn-cnv-vcf*: Joint-VCF mit strukturellen Varianten aus dem CNV-Calling-Schritt. Wenn ausgelassen, werden Prüfungen mit überlappenden Kopienzahlvarianten übersprungen.

Harte Keimbahnvariantenfilterung

DRAGEN bietet eine Post-VCF-Variantenfilterung anhand von Annotationen in den VCF-Datensätzen. Im Folgenden werden die standardmäßige und nichtstandardmäßige harte Variantenfilterung beschrieben. Aufgrund der Beschaffenheit der DRAGEN-Algorithmen, die die Hypothese korrelierter Fehler aus dem Varianten-Caller berücksichtigen, wurde jedoch die Varianten-Rausch-Unterscheidung in der Pipeline deutlich verbessert, was die Abhängigkeit von der Post-VCF-Filterung erheblich verringert. Daher ist die standardmäßige Post-VCF-Filterung in DRAGEN extrem einfach.

Standardmäßige harte Variantenfilterung

Die Standardfilter in der Keimbahn-Pipeline sind folgende:

- ##FILTER=<ID=DRAGENHardQUAL,Description="Set if true:QUAL < 10.4139">
- ##FILTER=<ID=PloidyConflict,Description="Genotype call from variant caller not consistent with chromosome ploidy">
- ##FILTER=<ID=lod_fstar,Description="Variant does not meet likelihood threshold (default threshold is 6.3)">
- DRAGENHardQUAL: Für sämtliche Contigs außer dem mitochondrialen Contig besteht die standardmäßige harte Filterung ausschließlich aus der Schwellenwertbildung anhand des QUAL-Werts.
- lod_fstar: Für das mitochondriale Contig besteht die standardmäßige harte Filterung ausschließlich aus der Schwellenwertbildung anhand des LOD-Scores.
- PloidyConflict: Dieser Filter wird auf alle Varianten-Calls auf ChrY weiblicher Proben angewendet, wenn in der DRAGEN-Befehlszeile „female“ (weiblich) angegeben wird.

Nichtstandardmäßige harte Variantenfilterung

DRAGEN unterstützt die grundlegende Filterung von Varianten-Calls, wie im VCF-Standard beschrieben. Mit der Option *--vc-hard-filter* lässt sich eine beliebige Anzahl von Filtern in einer durch Semikolons getrennten Liste von Ausdrücken angeben:

<Filter-ID>:<snp|indel|all>:<Kritierienliste>,

wobei die Kriterienliste selbst eine Liste von Ausdrücken ist, die im folgenden Format durch den Operator || (OR) getrennt wird:

<Annotations-ID> <Vergleichsoperator> <Wert>

Die Elemente der Ausdrücke haben dabei folgende Bedeutung:

- Filter-ID: Der Name des Filters, der in der Spalte FILTER der VCF-Datei für Calls angegeben wird, die anhand dieses Ausdrucks gefiltert werden.

- **snp/indel/all**: Die Untergruppe der Varianten-Calls, auf die der Ausdruck angewendet werden soll.
- **Annotations-ID**: Die Annotation zum Varianten-Call-Datensatz, für die Werte für den Filter überprüft werden sollen. Unterstützte Annotationen sind FS, MQ, MQRankSum, QD und ReadPosRankSum.
- **Vergleichsoperator**: Der numerische Operator für den Vergleich mit dem angegebenen Filterwert. Unterstützte Operatoren sind <, ≤, =, ≠, ≥ und >.

Beispielsweise markiert der folgende Ausdruck mit der Kennzeichnung „SPN filter“ alle SNPs mit FS < 2,1 oder mit MQ < 100 und mit „indel filter“ alle Datensätze mit FS < 2,2 oder mit MQ < 110:

```
--vc-hard-filter="SNP filter:snp:FS < 2.1 || MQ < 100; indel
   filter:indel:FS < 2.2 || MQ < 110"
```

Dieses Beispiel dient lediglich zu Illustrationszwecken. Die Verwendung für die DRAGEN V3-Ausgabe wird NICHT empfohlen. Illumina empfiehlt die Verwendung der standardmäßigen harten Filter.

Wertvergleiche können ausschließlich mit OR kombiniert werden. Arithmetische Kombinationen mehrerer Annotationen werden nicht unterstützt. In Zukunft werden möglicherweise auch komplexere Ausdrücke unterstützt.

Ausrichtungsverzerrungsfilter

Mit dem Ausrichtungsverzerrungsfilter wird das Rauschen reduziert, das typischerweise in folgenden Situationen auftritt:

- Artefakte, die vor der Adapterligation auftreten und während der Vorbereitung der Genbibliothek eingeführt werden (eine Kombination aus Wärme, Schneiden und Kontaminierung durch Metalle kann zu einer Paarung der 8-Oxoguanin-Basen mit Cytosin oder Adenin und somit während der PCR-Amplifikation zu G→T-Transversionsmutationen führen.)
- FFPE-Artefakt (formalinfixiert, in Paraffin eingebettet). FFPE-Artefakte entstehen durch Desaminierung von Cytosinen unter Einwirkung von Formaldehyd, was zu C→T-Transitionsmutationen führt.

Der Ausrichtungsverzerrungsfilter kann nur für somatische Pipelines verwendet werden. Durch Festlegen der Option *--vc-enable-orientation-bias-filter* auf „true“ wird der Filter aktiviert. Die Standardeinstellung ist „false“.

Der zu filternde Artefakttyp lässt sich über die Option *--vc-orientation-bias-filter-artifacts* festlegen. Die Standardeinstellung lautet „C/T,G/T“ und entspricht OxoG- und FFPE-Artefakten. Die gültigen Werte umfassen „C/T“ oder „G/T“ oder „C/T,G/T,C/A“.

Ein Artefakt (bzw. ein Artefakt und sein umgekehrtes Komplement) kann nur einmal aufgeführt werden. „C/T,G/A“ ist beispielsweise nicht gültig, da es sich bei C→G und T→A um umgekehrte Komplemente handelt.

dbSNP-Annotation

In den Modi „Germline“ (Keimbahn), „Tumor-Normal somatic“ (Tumor-Normal-somatisch) oder „Tumor-Only somatic“ (Tumor-Only-somatisch) kann DRAGEN nach Varianten-Calls in einer dbSNP-Datenbank suchen und Annotationen für alle dort gefundenen Übereinstimmungen hinzufügen. Legen Sie zur Aktivierung der dbSNP-Datenbanksuche die Option *--dbsnp* auf den vollständigen Pfad zur dbSNP-Datenbank-VCF- oder -.vcf.gz-Datei fest, sortiert in der Referenzreihenfolge.

Für jeden Varianten-Call in der Ausgabe-VCF gilt Folgendes: Wenn der Call mit einem Datenbankeintrag für CHROM, POS, REF und mindestens einem ALT übereinstimmt, dann wird die rsID für den übereinstimmenden Datenbankeintrag in die ID-Spalte für diesen Call in der Ausgabe-VCF kopiert. Zusätzlich fügt DRAGEN für in der Datenbank gefundene Calls im Feld INFO eine DB-Annotation hinzu.

DRAGEN gleicht die Varianten-Calls anhand der Bezeichnung für Referenzsequenz bzw. -Contig ab, es besteht jedoch keine zusätzliche Möglichkeit, um zu überprüfen, ob für die Erstellung der dbSNP die gleiche Referenz verwendet wird wie für das Alignment und das Varianten-Calling. Stellen Sie sicher, dass die Contigs in der ausgewählten Annotationsdatenbank mit denen in der Alignment-/Varianten-Calling-Referenz übereinstimmen.

PON-VCF (Normalgruppen-VCF-Datei)

Wenn DRAGEN im somatischen Modus Tumor-Normal oder Tumor-Only ausgeführt wird, sucht die Software in der Normalgruppen-VCF-Datei (PON-VCF) nach Varianten-Calls. Die PON-VCF-Datei muss vorab erstellt werden. Sie stellt einen Satz an Varianten dar, die durch die somatische DRAGEN-Pipeline bei der Ausführung einer Gruppe an Normalproben festgestellt wurden (diese müssen nicht notwendigerweise mit den Probanden übereinstimmen, von denen die Tumorprobe entnommen wurde). Idealerweise sollte die PON-VCF-Datei jedoch anhand von Normalproben erstellt werden, die mit dem gleichen Gerät zur Bibliotheksvorbereitung/-sequenzierung erfasst wurden. Auf diese Weise werden systematische Fehler, die während der Bibliotheksvorbereitung/-sequenzierung auftreten, in der PON-VCF-Datei berücksichtigt.

- *--panel-of-normals*

  Gibt eine PON-VCF-Datei an. Wenn eine PON-VCF-Datei als Eingabe verwendet und in mindestens einer Probe in der Datei (die mehrere Dutzend Proben enthalten kann) eine somatische Variante gefunden wird, wird sie in der Spalte FILTER der VCF-Ausgabedatei als „panel_of_normals“ gekennzeichnet.

Automatisch erstellte MD5SUM für VCF-Dateien

Für VCF-Ausgabedateien wird automatisch eine MD5SUM-Datei erstellt. Diese Datei wird im gleichen Ausgabeverzeichnis wie die VCF-Datei gespeichert und trägt deren Namen, ergänzt um die Endung .md5sum. Beispiel: whole_genome_run_123.vcf.md5sum. Bei MD5SUM-Dateien handelt es sich um einzeilige Textdateien mit der md5sum-Angabe aus der VCF-Ausgabedatei. Diese md5sum-Angabe deckt sich mit dem Wert, der bei Eingabe des Linux-Befehls „md5sum“ angezeigt wird.

Kopienzahlvarianten-Calling

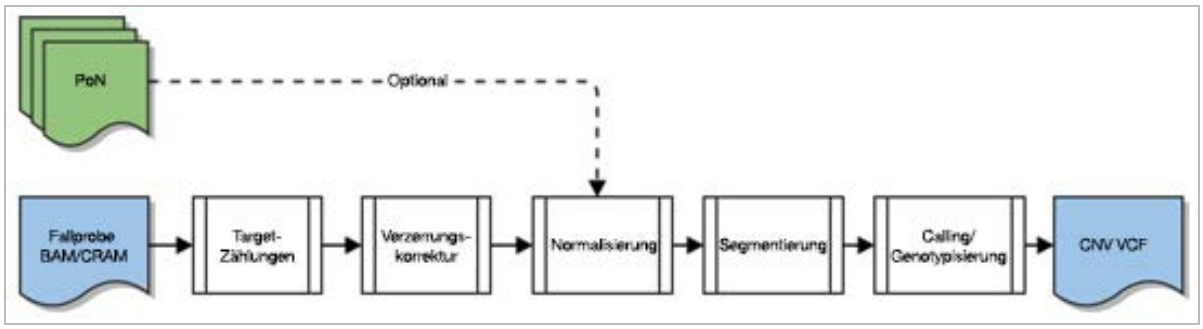
Die DRAGEN-Pipeline für Kopienzahlvarianten (Copy Number Variant, CNV) kann das Calling von CNV-Ereignissen anhand von NGS-Daten (Next-Generation Sequencing, Sequenzierung der nächsten Generation) ausführen. Über die DRAGEN-Hostsoftware und eine zentrale Oberfläche unterstützt diese Pipeline mehrere Anwendungen, darunter die Verarbeitung von Gesamtgenom-Sequenzierungsdaten und Gesamtexom-Sequenzierungsdaten für die Keimbahn-Analyse.

Die DRAGEN-CNV-Pipeline unterstützt zwei Betriebsmodi für die Normalisierung. Die Modi verwenden jeweils unterschiedliche Normalisierungsverfahren für den Umgang mit Verzerrungen, die sich je nach Anwendung (z. B. WGS im Vergleich zu WES) unterscheiden. Während die Standardoption auf den optimalen Kompromiss hinsichtlich Geschwindigkeit und Genauigkeit setzt, ist für bestimmte Workflows möglicherweise eine Feinabstimmung der Optionen erforderlich.

CNV-Workflow

Der Workflow der DRAGEN-CNV-Pipeline sieht aus wie in der folgenden Abbildung dargestellt.

Abbildung 3 Workflow für die DRAGEN-CNV-Pipeline



Diese Pipeline nutzt zahlreiche Eigenschaften der DRAGEN-Plattform, die in anderen Pipelines zur Verfügung stehen, beispielsweise die Hardwarebeschleunigung und die effiziente E/A-Verarbeitung. Legen Sie die Befehlszeilenoption *--enable-cnv* auf „true“ fest, um die CNV-Verarbeitung in der DRAGEN-Hostsoftware zu aktivieren.

Die CNV-Pipeline umfasst die folgenden Verarbeitungsmodule:

- Target-Zählungen: Klasseneinteilung von Read-Zählungen und anderen Signalen aus Alignments.
- Abweichungskorrektur: Korrektur immanenter Systemabweichungen.
- Normalisierung: Bestimmung der normalen Ploidiegrade und Normalisierung der Fallprobe.
- Segmentierung: Unterbrechungspunktbestimmung durch Segmentierung des normalisierten Signals.
- Calling/Genotypisierung: Schwellenwertbildung, Scoring, Qualifikation und Filterung von putativen Ereignissen als Kopienzahlvarianten.

Wahlweise kann in das Normalisierungsmodul eine Normalgruppe (Panel of Normals, PoN) geladen werden. Diese wird verwendet, wenn Kohorten- oder Populationsproben verfügbar sind. Alle anderen Module werden von den unterschiedlichen CNV-Algorithmen unterschiedlich behandelt.

Signalflussanalyse

Die folgenden Abbildungen bieten eine umfassende Übersicht über die Schritte in der DRAGEN-CNV-Pipeline, während das Signal die unterschiedlichen Phasen durchquert. Es handelt sich um Beispielabbildungen, die mit den von der DRAGEN-CNV-Pipeline generierten Diagrammen nicht identisch sind.

Der erste Schritt in der DRAGEN-CNV-Pipeline ist die Target-Zählung. In dieser Phase werden Signale wie Read-Anzahl und nicht zusammengehörige Paare extrahiert und in Target-Intervalle eingeteilt.

Abbildung 4 Signal für Read-Anzahl

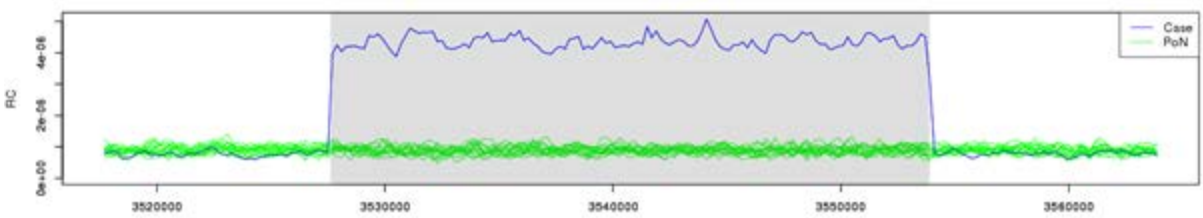
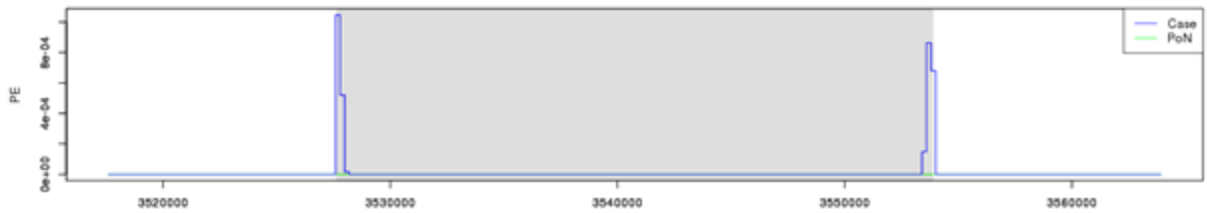
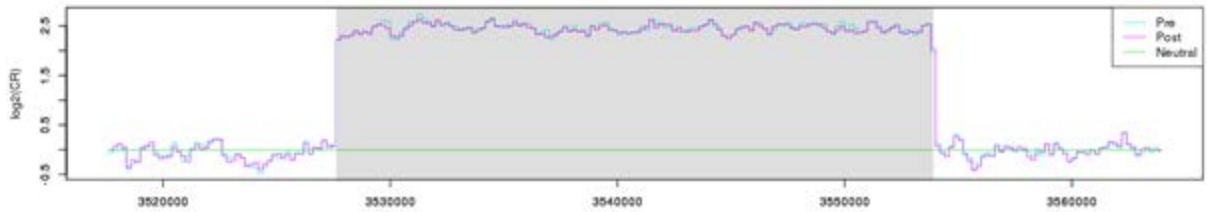


Abbildung 5 Signal für nicht zusammengehörige Paare



Im nächsten Schritt wird die Fallprobe in Relation zur Normalgruppe oder zum geschätzten normalen Ploidiegrad normiert und etwaige andere Verzerrungen werden aus dem Signal herausgerechnet, um alle Signale auf Ereignisebene zu amplifizieren.

Abbildung 6 Prä/post tangentiale Normalisierung



Das normierte Signal wird dann mithilfe eines der verfügbaren Segmentierungsalgorithmen segmentiert. Das Calling der Ereignisse erfolgt über diese Abschnitte.

Abbildung 7 Segmente

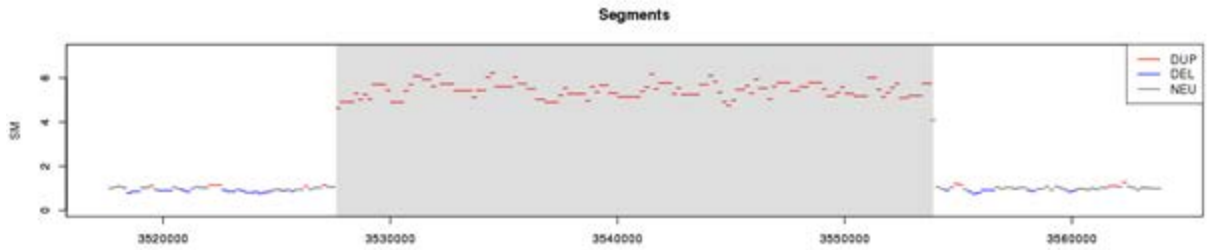


Abbildung 8 Ereignisse mit Call

Die Ereignisse erhalten dann einen Score und werden in der VCF-Ausgabedatei ausgegeben.

Optionen für die CNV-Pipeline

Im Folgenden sind die Optionen höchster Ebene aufgeführt, die zur Steuerung der CNV-Pipeline verwendet werden und auch in der DRAGEN-Hostsoftware anwendbar sind. Die Eingabe in das DRAGEN CNV kann über eine BAM- oder CRAM-Datei erfolgen. Mit dem DRAGEN-Mapper und -Aligner können auch FASTQ-Dateien verwendet werden.

- *--bam-input*: Die zu verarbeitende BAM-Datei.
- *--cram-input*: Die zu verarbeitende CRAM-Datei.
- *--enable-cnv*: Aktiviert/deaktiviert die CNV-Verarbeitung. Legen Sie diese Option auf „true“ fest, um die CNV-Verarbeitung zu aktivieren.
- *--enable-map-align*: Aktiviert/deaktiviert das Mapper- und Aligner-Modul. Die Standardeinstellung ist „true“. Alle Eingabe-Reads werden neu gemappt und aligniert, sofern die Option nicht auf „false“ festgelegt wird.
- *--fastq-file1*, *--fastq-file2*: Die zu verarbeitende(n) FASTQ-Datei(en).
- *--output-directory*: Ausgabeverzeichnis, in dem alle Ergebnisse gespeichert werden.
- *--output-file-prefix*: Präfix der Ausgabedatei, das allen Ergebnis-Dateinamen vorangestellt wird.
- *--ref-dir*: Hashtabellen-Verzeichnis des DRAGEN-Referenzgenoms.

Eingabe für die CNV-Pipeline

Die DRAGEN-CNV-Pipeline unterstützt mehrere Eingabeformate. Das häufigste Format ist eine bereits gemappte und alignierte BAM- oder CRAM-Datei. Wenn Ihre Daten noch nicht gemappt und aligniert wurden, finden Sie unter *Erstellen einer Alignment-Datei* auf Seite 56 weitere Informationen.

Informationen zum direkten Ausführen der DRAGEN-CNV-Pipeline mit einer FASTQ-Eingabe ohne Erstellen einer BAM- oder CRAM-Datei finden Sie unter *Weitergeben von Alignments* auf Seite 56. In diesem Abschnitt werden Schritte zum Weitergeben von Alignment-Datensätzen direkt aus der Mapping-Alignment-Phase von DRAGEN beschrieben.

Referenz-Hashtabelle

Zusätzlich zu weiteren Optionen, die auch bei anderen Pipelines erforderlich sind, muss für die DRAGEN-CNV-Pipeline die Hashtabelle mit der Option *--enable-cnv* und der Einstellung „true“ generiert werden. Wenn *--enable-cnv* auf „true“ festgelegt ist, generiert *dragen* eine zusätzliche k-mer-Uniqueness-Map, mit der der CNV-Algorithmus Verzerrungen der Mappingfähigkeit entgegenwirkt. Die Datei mit der k-mer-Uniqueness-Map muss nur einmal pro Referenz-Hashtabelle generiert werden. Dies nimmt pro humanem Gesamtgenom jeweils ca. 1,5 Stunden in Anspruch.

Bei der Referenz-Hashtabelle handelt es sich um eine vorab generierte Binärdarstellung des Referenzgenoms. Informationen zum Erstellen einer Hashtabelle finden Sie unter *Vorbereiten eines Referenzgenoms* auf Seite 129.

Der Befehl im folgenden Beispiel generiert eine Hashtabelle.

```
dragen \
   --build-hash-table true \
   --ht-reference <FASTA> \
   --output-directory <AUSGABE> \
   --enable-cnv true \
   --enable-rna true
```

Erstellen einer Alignment-Datei

Die folgenden Beispiele für Befehlszeilen zeigen, wie die DRAGEN-Mapping-Alignment-Pipeline je nach Eingabetyp ausgeführt wird. Die Mapping-Alignment-Pipeline erstellt eine Alignment-Datei in Form einer BAM- oder CRAM-Datei, die dann in der DRAGEN-CNV-Pipeline verwendet werden kann.

Sie müssen Alignment-Dateien für alle Proben erstellen, für die noch kein Mapping und Alignment durchgeführt wurde, einschließlich aller Proben, die als Referenzen für die Normalisierung verwendet werden sollen. Jede Probe muss über einen eindeutigen Probenbezeichner verfügen, der mit der Option *--RGSM* angegeben wird. Bei BAM- und CRAM-Eingabedateien wird der Probenbezeichner aus der Datei entnommen. Daher ist die Option *--RGSM* nicht erforderlich.

Beispielbefehl für das Mapping/Alignment einer FASTQ-Datei:

```
dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
   -2 <FASTQ2> \
   --RGSM <PROBE> \
   --RGID <RGID> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true
```

Beispielbefehl für das Mapping/Alignment einer vorhandenen BAM-Datei:

```
dragen \
   -r <HASHTABELLE> \
   --bam-input <BAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true
```

Beispielbefehl für das Mapping/Alignment einer vorhandenen CRAM-Datei:

```
dragen \
   -r <HASHTABELLE> \
   --cram-input <CRAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true
```

Weitergeben von Alignments

DRAGEN kann FASTQ-Proben mappen und alignieren und diese im Anschluss direkt an die nachgeschalteten Caller weitergeben. Nachgeschaltete Caller sind beispielsweise der CNV-Caller und der Haplotyp-Varianten-Caller. Bei dieser Verarbeitung können Sie die Erstellung einer BAM- oder CRAM-Datei überspringen und müssen keine zusätzlichen Dateien speichern.

Wenn Sie Alignments direkt an die DRAGEN-CNV-Pipeline weitergeben möchten, führen Sie die FASTQ-Probe in einem regulären DRAGEN-Mapping-Alignment-Workflow aus und geben Sie dann zusätzliche Argumente für eine CNV-Aktivierung an. Im folgenden Beispiel ist eine Befehlszeile zum Mappen/Alignieren einer FASTQ-Datei dargestellt, die dann an die CNV-Pipeline gesendet wird.

```
dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
```

```
-2 <FASTQ2> \
--RGSM <PROBE> \
--RGID <RGID> \
--output-directory <AUSGABE> \
--output-file-prefix <PROBE> \
--enable-map-align true \
--enable-cnv true \
--cnv-enable-self-normalization true
```

Informationen zum gleichzeitigen Ausführen des CNV- und des Haplotyp-Varianten-Callers finden Sie unter *Gleichzeitiges CNV- und Haplotyp-Varianten-Calling* auf Seite 68.

Target-Zählungen

Die Target-Zählung ist die erste Verarbeitungsphase in der DRAGEN-CNV-Pipeline. In diesem Schritt werden die Alignments in Intervalle gruppiert. Das primäre Analyseformat für die CNV-Verarbeitung ist die Target-Zählungsdatei mit Merkmalsignalen, die aus den Alignments extrahiert werden und in der nachfolgenden Verarbeitung verwendet werden können. Gruppierungsstrategie, Intervallgrößen sowie deren Grenzwerte werden über die Optionen für die Generierung der Target-Zählung und die verwendete Normalisierungsmethode gesteuert.

Bei Gesamtgenom-Sequenzdaten werden die Intervalle für die Referenz-Hashtabelle automatisch generiert. Nur die primären Contigs der Referenz-Hashtabelle werden für die Gruppierung berücksichtigt. Sie können mithilfe der Option *--cnv-skip-contig-list* zusätzliche Contigs festlegen, die übersprungen werden sollen.

Bei Gesamtexom-Sequenzdaten werden die zu analysierenden Intervalle anhand der über die Option *--cnv-target-bed* bereitgestellten BED-Zieldatei bestimmt.

Die Target-Zählung generiert eine .target.counts-Datei. Diese kann später anstelle einer beliebigen BAM- oder CRAM-Datei verwendet werden, indem sie mithilfe der Option *--cnv-input* für die Normalisierungsphase angegeben wird. Die .target.counts-Datei ist eine vorläufige Datei der DRAGEN-CNV-Pipeline und sollte nicht geändert werden.

Die .target.counts-Datei ist eine tabulatorgetrennte Textdatei mit folgenden Spalten:

- Contig-Bezeichner
- Startposition
- Endposition
- Bezeichnung des Target-Intervalls
- Anzahl der Alignments in diesem Intervall
- Anzahl der nicht zusammengehörigen Alignments in diesem Intervall

Im folgenden Beispiel ist eine *.target.counts-Datei aufgeführt.

```
contig  start   stop    name                SampleName  improper_pairs
1       565480  565959  target-wgs-1-565480  7          6
1       566837  567182  target-wgs-1-566837  9          0
1       713984  714455  target-wgs-1-713984  34         4
1       721116  721593  target-wgs-1-721116  47         1
1       724219  724547  target-wgs-1-724219  24         21
1       725166  725544  target-wgs-1-725166  43         12
1       726381  726817  target-wgs-1-726381  47         14
1       753243  753655  target-wgs-1-753243  31         2
```

```
1       754322  754594  target-wgs-1-754322  27          0
1       754594  755052  target-wgs-1-754594  41          0
```

Gesamtgenom

Wenn es sich bei den Proben um Gesamtgenomproben handelt, wird die effektive Breite der Target-Intervalle mit der Option *--cnv-interval-width* festgelegt. Je höher die Coverage einer Probe, desto höher die Auflösung, die erkannt werden kann. Diese Option ist bei der Ausführung mit einer Normalgruppe wichtig, da alle Proben über übereinstimmende Intervalle verfügen müssen. Bei der Selbstnormalisierung kann die effektive Breite größer sein als der festgelegte Wert.

WGS-Coverage	Empfohlene Auflösung*
5x	1.000 bp
10x	1.000 bp
20x	500 bp
30x	250 bp
50x	250 bp

* Sie können auch zwischen Auflösung und Geschwindigkeit abwägen.

Die Intervalle werden für jedes Contig in der Referenz automatisch generiert. Sie können mithilfe der Option *--cnv-skip-contig-list* eine Liste mit Contigs festlegen, die übersprungen werden sollen. Für diese Option ist eine kommagetrennte Liste mit Contig-Bezeichnern erforderlich. Die Contig-Bezeichner müssen mit der verwendeten Referenz-Hashtabelle übereinstimmen. Standardmäßig werden nur die mitochondrialen Chromosomen übersprungen. Nicht primäre Contigs werden nicht verarbeitet.

Mit der folgenden Option können Sie z. B. die Chromosomen M, X und Y überspringen:

```
--cnv-skip-contig-list "chrM,chrX,chrY"
```

Gesamtexom

Wenn es sich bei den Proben um Gesamtexomproben handelt, sollte mit der Option *--cnv-target-bed $TARGET_BED* eine Target-BED-Datei bereitgestellt werden.

Die Target-BED-Datei erfordert eine Kopfzeile und eine vierte Spalte, die die Target-Bezeichnung angibt. Im Folgenden finden Sie einen einfachen awk-Befehl, der die Target-BED-Eingabedatei (ohne Kopf- oder Befehlszeilen) in eine für CNV geeignete Datei übersetzt.

```
cat <(echo -e "contig\tstart\tstop\tname") \
<(awk '{print $1"\t"$2"\t"$3"\ttarget-"NR}' $ORIGINAL_BED)
```

Eine normale BED-Datei ohne Kopfzeile ist ebenfalls zulässig. In diesem Fall werden die Target-Bezeichnungen in der vierten Spalte vom CNV-Algorithmus während der Target-Zählung automatisch generiert. Stellen Sie zur Verwendung einer normalen BED-Datei sicher, dass diese keine Kopfzeile enthält. In diesem Fall werden wie beim DRAGEN-Varianten-Caller alle Spalten ab der dritten Spalte ignoriert.

Optionen für die Target-Zählung

Die folgenden Optionen steuern die Generierung von Target-Zählungen.

- *--cnv-counts-method*: Gibt die Zählungsmethode für ein Alignment an, für das eine Zählung in einer Target-Klasse erstellt werden soll. Mögliche Werte sind „midpoint“, „start“ und „overlap“. Der

Standardwert bei Verwendung des Normalgruppenverfahrens ist „overlap“. Hierbei wird ein Alignment, bei dem eine Überlappung mit einem Teil der Target-Klasse vorhanden ist, für diese Klasse gezählt. Im Selbstnormalisierungsmodus ist die Standardzählmethode „start“.

- *--cnv-min-mapq*: Gibt die minimale MAPQ für einen während der Generierung der Target-Zählung zu zählenden Read an. Beachten Sie, dass Reads mit der MAPQ 0 immer gezählt werden, unabhängig von dieser Einstellung. Der Standardwert ist 20.
- *--cnv-target-bed*: Gibt eine korrekt formatierte BED-Datei an, die die Target-Intervalle für das Coverage-Sampling enthält. Wird für die WES-Analyse verwendet.
- *--cnv-interval-width*: Gibt die Breite des Sampling-Intervalls für die CNV-WGS-Verarbeitung an. Diese Option steuert die effektive Fenstergröße. Der Standardwert ist 1000.
- *--cnv-skip-contig-list*: Gibt eine kommagetrennte Liste mit Contig-Bezeichnern an, die beim Generieren von Intervallen für die WGS-Analyse übersprungen werden. Wenn nicht anders angegeben, werden standardmäßig die Contigs „chrM“, „MT“, „m“ und „chrm“ übersprungen.

Korrektur der GC-Verzerrung

Ein typischer NGS-Workflow führt zu Verzerrungen, die das Calling von CNV-Ereignissen erschweren. Diese Verzerrungen lassen sich auf die Bibliotheksvorbereitung, Capture-Kits, Sequenzierer-Unterschiede und auch Mapping-Verzerrungen zurückführen. Die DRAGEN-CNV-Pipeline korrigiert diese Verzerrungen in den unterschiedlichen Verarbeitungsphasen.

Das Modul für die Korrektur der GC-Verzerrung wird unmittelbar nach der Target-Zählung ausgeführt und bezieht sich auf die .target.count-Datei. Die Korrektur der GC-Verzerrung generiert eine Dateiversion mit korrigierter GC-Verzerrung. Der Dateiname enthält die Erweiterung .target.counts.gc-corrected. Versionen mit korrigierter GC-Verzerrung werden für die nachgeschaltete Verarbeitung von WGS-Daten empfohlen. Wenn für die Gesamtexom-Sequenzierung (Whole Exome Sequencing, WES) genügend Zielregionen vorhanden sind, können ebenfalls die Werte der korrigierten GC-Verzerrung verwendet werden.

Typische Capture-Kits verfügen über mehr als 200.000 Targets in den Regionen von Interesse. Wenn Ihre BED-Datei über weniger als 200.000 Targets verfügt oder wenn sich die Zielregionen in einer bestimmten Genomregion befinden (und dadurch möglicherweise eine unzutreffende GC-Verzerrungsstatistik verursachen), sollte die Korrektur der GC-Verzerrung deaktiviert werden.

Mithilfe der folgenden Optionen kann das Modul für die Korrektur der GC-Verzerrung gesteuert werden.

- *--cnv-enable-gcbias-correction*: Aktiviert/deaktiviert die Korrektur der GC-Verzerrung bei der Generierung der Target-Zählung. Die Standardeinstellung ist „true“.
- *--cnv-enable-gcbias-smoothing*: Aktiviert/deaktiviert eine Glättung der Korrektur der GC-Verzerrung über benachbarte GC-Klassen mit exponentiellem Kernel. Die Standardeinstellung ist „true“.
- *--cnv-num-gc-bins*: Gibt die Anzahl der Klassen für die Korrektur der GC-Verzerrung an. Jede Klasse repräsentiert den Prozentsatz des GC-Gehalts. Zulässige Werte sind 10, 20, 25, 50 oder 100. Der Standardwert ist 25.

Normalisierung

Die DRAGEN-CNV-Pipeline unterstützt zwei Normalisierungsalgorithmen:

- Selbstnormalisierung: Anhand von Statistiken für die analysierte Probe wird das Ploidiegrad-Basisniveau bestimmt.

- Normalgruppe (Panel of Normals, PON): Referenz-basierter Normalisierungsalgorithmus, der anhand von zusätzlichen übereinstimmenden Normalproben ein Basisniveau für das Calling von CNV-Ereignissen bestimmt. Die übereinstimmenden Normalproben bedeuten in diesem Fall, dass Bibliotheksvorbereitung und Sequenzierungsworkflow von der Fallprobe übernommen wurden.

Der zu verwendende Algorithmus ist von den verfügbaren Daten und der Anwendung abhängig. Wählen Sie den Modus der Normalisierung anhand der folgenden Richtlinien aus.

Selbstnormalisierung

- Gesamtgenom-Sequenzierung
- Einzelprobenanalyse
- Keine zusätzlichen übereinstimmenden Proben verfügbar
- Einfacherer Workflow über Einzelaufruf

Normalgruppe

- Gesamtgenom-Sequenzierung
- Gesamtexom-Sequenzierung
- Gezielte Panels
- Zusätzliche übereinstimmende Proben verfügbar
- Tumor-/Matched-Normal-Analyse

Selbstnormalisierung

Die DRAGEN-CNV-Pipeline bietet einen Selbstnormalisierungsmodus, für den keine Referenzprobe oder Normalgruppe erforderlich ist. Aktivieren Sie diesen Modus, indem Sie *--cnv-enable-self-normalization* auf „true“ festlegen. Dieser Betriebsmodus ist weniger zeitaufwendig, da nicht zwei Phasen ausgeführt werden müssen. Das Basisniveau für den Call wird anhand der in der Fallprobe vorliegenden Statistiken festgelegt.

Da für die Selbstnormalisierung die Statistiken in der Fallprobe verwendet werden, wird dieser Modus aufgrund der Möglichkeit unzureichender Daten nicht für WES- oder zielgerichtete Sequenzierungsanalysen empfohlen.

Der Selbstnormalisierungsmodus wird für die Gesamtgenomsequenzierung mit Einzelprobenverarbeitung empfohlen. Die Pipeline wird bis zur Segmentierungs- und Calling-Phase ausgeführt, wobei die endgültigen Calling-Ereignisse erstellt werden.

```
dragen \
   -r <HASHTABELLE> \
   --bam-input <BAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align false \
   --enable-cnv true \
   --cnv-enable-self-normalization true
```

Wenn Sie einen Lauf von einer FASTQ-Probe ausführen, ist die Selbstnormalisierung der Standardbetriebsmodus.

Im Selbstnormalisierungsmodus legt die Option *--cnv-interval-width*, die während der Target-Zählung verwendet wird, die effektive Intervallbreite basierend auf der Anzahl eindeutiger k-mer-Positionen fest. In der Regel müssen Sie diese Option nicht ändern.

Normalgruppe

Bei der Normalgruppenmethode (Panel of Normals, PON) wird anhand von einem Satz übereinstimmender Normalproben das Basisniveau für das Calling von CNV-Ereignissen bestimmt. Diese übereinstimmenden Normalproben müssen aus dem gleichen Bibliotheksvorbereitungs- und Sequenzierungsworkflow stammen, der auch für die Fallprobe verwendet wurde. Dies ermöglicht dem Algorithmus, Abweichungen auf Systemebene herauszurechnen, die nicht probenspezifisch sind.

In diesem Betriebsmodus ist die DRAGEN-CNV-Pipeline in zwei voneinander getrennte Phasen unterteilt. Zur Klassifizierung der Alignments durchlaufen alle Proben, Fälle und Normalgruppen die Target-Zählungsphase. Anschließend durchläuft die Fallprobe die Normalisierungs- und Call-Erkennungsphase und wird dabei zur Bestimmung der Ereignisse mit der Normalgruppe abgeglichen.

Target-Zählungsphase

Target-Zählungen sollten für alle Proben durchgeführt werden, egal, ob sie als Referenzen verwendet werden sollen oder die zu untersuchenden Fallproben sind. Die Fallprobe und alle Proben, die als eine Normalgruppenprobe verwendet werden sollen, müssen identische Intervalle aufweisen und sollten deshalb mit identischen Einstellungen erstellt werden. In der Target-Zählungsphase wird außerdem auch eine Korrektur der GC-Verzerrung durchgeführt. Diese ist standardmäßig aktiviert.

Die folgenden Beispiele beziehen sich auf die WGS-Verarbeitung. Mehr zur Exomverarbeitung finden Sie unter *Gesamtexom* auf Seite 58.

Im Folgenden finden Sie einen Beispielbefehl für die Verarbeitung einer BAM-Datei.

```
dragen \
   -r <HASHTABELLE> \
   --bam-input <BAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align false \
   --enable-cnv true
```

Im Folgenden finden Sie einen Beispielbefehl für die Verarbeitung einer CRAM-Datei.

```
dragen \
   -r <HASHTABELLE> \
   --cram-input <CRAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align false \
   --enable-cnv true
```

Normalisierungs- und Call-Erkennungsphase

Als nächster Schritt in der CNV-Pipeline bei Verwendung einer Normalgruppe werden die Normalisierung und die Calls durchgeführt. Dazu gehört die Auswahl einer Normalgruppe, die aus einer Liste mit Target-Zählungsdateien besteht, die für die referenzbasierte Mediannormalisierung erforderlich ist.

Sie können die Analyse in anderen Workflow-Kombinationen ausführen. Bitte beachten: Das Calling der CNV-Ereignisse erfolgt für die verwendeten Referenzproben. Idealerweise werden für die Normalgruppenproben Bibliotheksvorbereitungs- und Sequenzierungsworkflows ausgeführt, die mit den Workflows für die zu analysierende Fallprobe übereinstimmen. Für das Calling von Geschlechtschromosomen wird die Verwendung von dem Geschlecht entsprechenden Proben in der Gruppe empfohlen. Da die Normalisierung zielbasiert anhand des Normalgruppenmedians erfolgt, sind dem Geschlecht entsprechende Referenzen für die Erkennung von Kopienzahlereignissen auf den Geschlechtschromosomen erforderlich.

Erstellen Sie für die Generierung einer Normalgruppe (Panel of Normals, PON) eine Textdatei, in der in jeder Zeile ein Pfad zu einer während der Target-Zählungsphase erstellten target.counts-Datei vorhanden ist. Relative Pfade können angegeben werden, sofern sich diese auf das aktuelle Arbeitsverzeichnis beziehen. Die Angabe vollständiger Pfade wird empfohlen, wenn der Workflow später verwendet oder für andere Benutzer freigegeben wird.

Im Folgenden finden Sie eine PON-Beispieldatei, in der eine Untergruppe der Dateien mit GC-Korrektur aus der Target-Zählungsphase verwendet wird.

```
/data/output_trio1/sample1.target.counts.gc-corrected
/data/output_trio1/sample2.target.counts.gc-corrected
/data/output_trio2/sample4.target.counts.gc-corrected
/data/output_trio2/sample5.target.counts.gc-corrected
/data/output_trio3/sample7.target.counts.gc-corrected
/data/output_trio3/sample8.target.counts.gc-corrected
```

Alternativ können die in der Normalgruppe verwendeten Dateien mit der Option *--cnv-normals-file* festgelegt werden. Mit dieser Option kann ein einziger Dateiname angegeben werden, die Option kann mehrfach festgelegt werden.

Nach Erstellen einer PON-Datei können Sie den Caller durch Festlegen Ihrer Fallprobe über die Option *--cnv-input* und durch Festlegen der PON-Datei mit der Option *--cnv-normals-list* ausführen. Da empfohlen wird, die Zählungen mit korrigierter GC-Verzerrung zu verwenden, ist das nochmalige Ausführen der GC-Verzerrungskorrektur nicht erforderlich. Durch Festlegen der Option *--cnv-enable-gcbias-correction* auf „false“ kann die Korrektur der GC-Verzerrung deaktiviert werden. Beispiel:

```
dragen \
   -r <HASHTABELLE> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align false \
   --enable-cnv true \
   --cnv-input <FALLZAHL> \
   --cnv-normals-list <NORMALWERTE> \
   --cnv-enable-gcbias-correction false
```

Mit diesem Befehl wird die Fallprobe anhand der Normalgruppe normalisiert. Anschließend wird die Segmentierungs- und Calling-Phase ausgeführt.

Normalisierungsoptionen

Diese Optionen steuern die Präkonditionierung der Normalgruppe und die Normalisierung der Fallprobe.

- *--cnv-enable-self-normalization*: aktiviert/deaktiviert den Selbstnormalisierungsmodus, für den keine Normalgruppe erforderlich ist.
- 9-*-cnv-extreme-percentile*: gibt den Extremwert der mittleren Perzentile an, ab dem Proben herausgefiltert werden. Der Standardwert ist 2.5.

- *--cnv-input*: gibt eine Target-Zählungsdatei für die untersuchte Fallprobe bei Verwendung einer Normalgruppe an.
- *--cnv-matched-normal*: gibt die Target-Zählungsdatei der zugeordneten Normalprobe an.
- *--cnv-normals-file*: gibt eine target.counts-Datei zur Verwendung in der Normalgruppe an. Diese Option kann mehrmals angegeben werden, für jede Datei einmal.
- *--cnv-normals-list*: gibt eine Textdatei mit den Pfaden zur Referenzliste der Target-Zählungsdateien an, die als Normalgruppe verwendet werden. Die Angabe vollständiger Pfade wird empfohlen, wenn der Workflow später verwendet oder für andere Benutzer freigegeben wird. Relative Pfade können angegeben werden, sofern sich diese auf das aktuelle Arbeitsverzeichnis beziehen.
- *--cnv-max-percent-zero-samples*: definiert einen Schwellenwert, ab dem Targets mit zu vielen Proben herausgefiltert werden, die eine Coverage von null aufweisen. Der Standardwert ist 5%.
- *--cnv-max-percent-zero-targets*: definiert einen Schwellenwert, ab dem Proben mit zu vielen Targets herausgefiltert werden, die eine Coverage von null aufweisen. Der Standardwert ist 2.5%.
- *--cnv-target-factor-threshold*: definiert einen Prozentsatz des mittleren Target-Faktor-Schwellenwerts, ab dem verwendbare Targets herausgefiltert werden. Der Standardwert ist 1% für die Gesamtgenomverarbeitung und 10% für die gezielte Sequenzierungsverarbeitung.
- *--cnv-truncate-threshold*: definiert einen Prozentsatz-Schwellenwert, ab dem extreme Ausreißer gekürzt werden. Der Standardwert ist 0.1%.

Segmentierung

Nach Normalisierung einer Fallprobe durchläuft die Probe eine Segmentierungsphase. In DRAGEN sind mehrere Segmentierungsalgorithmen implementiert, z. B.:

- CBS (Circular Binary Segmentation)
- SLM (Shifting Level Models)
- FPOP (Functional Pruning Optimal Partitioning)

Für den SLM-Algorithmus gibt es zwei Varianten: SLM und HSLM. HSLM (heterogenes SLM) wird für die Exomanalyse verwendet und bearbeitet Target-Capture-Kits mit ungleichen Abständen.

Der verwendete Standardalgorithmus für die Segmentierung in der Gesamtgenomverarbeitung ist SLM, für die Gesamtexom-Verarbeitung ist der Algorithmus CBS.

- *--cnv-segmentation-mode*: Legt den auszuführenden Segmentierungsalgorithmus fest. Der Wert lautet „cbs“, „slm“, „hslm“ oder „fpop“. Der Standardwert ist „slm“ oder „cbs“, abhängig davon, ob es sich bei den Intervallen um Gesamtgenomintervalle oder gezielte Sequenzierungsintervalle handelt.
- *--cnv-merge-distance*: Legt die Mindestanzahl an Basenpaaren zwischen zwei Segmenten fest, ab der eine Zusammenfassung zulässig ist. Der Standardwert ist 0, d. h., es muss sich um direkt benachbarte Segmente handeln.
- *--cnv-merge-threshold*: Legt den maximalen Unterschied zwischen den Segmentmittelwerten fest, bis zu dem eine Zusammenfassung von zwei benachbarten Segmenten zulässig ist. Der Segmentmittelwert wird als linearer Kopienverhältniswert dargestellt. Der Standardwert ist 0.2. Legen Sie den Wert auf 0 fest, wenn Sie die Zusammenfassung deaktivieren möchten.

Externe R-Pakete

Zur Verwendung der Segmentierungsalgorithmen SLM oder FPOP müssen externe R-Pakete installiert werden. R-Pakete werden außerhalb der DRAGEN-Hostsoftware ausgeführt. DRAGEN enthält unter /opt/edico/R/install_R_packages.R ein einfaches Installationsskript. Bei diesen R-Paketen handelt es sich

um zusätzliche Segmentierungsverfahren. Sie sind nicht offizieller Teil der DRAGEN-Hostsoftware. Wenn Ihr System die Installation von R- oder Drittanbieterbibliotheken nicht zulässt, können Sie auf CBS (Circular Binary Segmentation) zurückgreifen.

Circular Binary Segmentation

Circular Binary Segmentation ist direkt in DRAGEN implementiert und basiert auf der Veröffentlichung *A faster circular binary segmentation for the analysis of array CGH data*. Durch diesen Algorithmus wird die Empfindlichkeit für NGS-Daten verbessert.

Mit folgenden Optionen wird Circular Binary Segmentation gesteuert.

- *--c-alpha*: Legt das Signifikanzniveau für den Test hinsichtlich der Akzeptanz von Changepoints fest. Der Standardwert ist 0.01.
- *--cnv-cbs-eta*: Legt bei Verwendung des Permutationsverfahrens die Typ-1-Fehlerrate des sequenziellen Grenzwerts für ein frühzeitiges Beenden fest. Der Standardwert ist 0.05.
- *--cnv-cbs-kmax*: Legt die maximale Breite des kleineren Segments für die Permutation fest. Der Standardwert ist 25.
- *--cnv-cbs-min-width*: Legt die Mindestanzahl von Markern für ein geändertes Segment fest. Der Standardwert ist 2.
- *--cnv-cbs-nmin*: Legt die Mindestdatenlänge für eine maximale statistische Näherung fest. Der Standardwert ist 200.
- *--cnv-cbs-nperm*: Legt die Anzahl an Permutationen für die Berechnung des p-Werts fest. Der Standardwert ist 10000.
- *--cnv-cbs-trim*: Legt den Anteil der Daten fest, die für die Varianzberechnungen gekürzt werden müssen. Der Standardwert ist 0.025.

Shifting Level Models-Segmentierung

Die Shifting Level Models(SLM)-Segmentierung folgt der R-Implementierung wie in SLMSuite: a suite of algorithms for segmenting genomic profiles dargestellt.

- *--cnv-slm-eta*: Ausgangswahrscheinlichkeit für eine Änderung des Mittelwertprozess-Werts. Der Standardwert ist 1e-5.
- *--cnv-slm-fw*: Minimale Anzahl von Datenpunkten für die Ausgabe einer CNV. Die Standardeinstellung ist 0, d. h., Segmente mit einer Designsonde können ausgegeben werden.
- *--cnv-slm-omega*: Skalierungsparameter für die relative Gewichtung von experimenteller/biologischer Varianz. Der Standardwert ist 0.3.
- *--cnv-slm-stepeta*: Parameter für die Distanznormalisierung. Der Standardwert ist 10000. Diese Option wird nur für HSLM verwendet.

Functional Pruning Optimal Partitioning(FPOP)-Segmentierung

Der FPOP-Segmentierungsmodus folgt dem Vorgehen der R-Implementierung von On Optimal Multiple Changepoint Algorithms for Large Data.

- *--cnv-fpop-penalty*: Abzugsoption für die Changepoint-Bestimmung. Der Standardwert ist 0.03.

Qualitätsbewertung

Qualitäts-Scores werden anhand eines Wahrscheinlichkeitsmodells berechnet, das eine Mischung aus Wahrscheinlichkeitsverteilungen mit vielen Nachkommastellen (eine pro ganzzahliger Kopienzahl) und einer Ereignislängengewichtung verwendet. Die Rauschvarianz wird geschätzt. Die Ausgabe-VCF enthält eine in der Phred-Skala angegebene Metrik, die die Zuverlässigkeit der ermittelten Amplifikations- (CN > 2 für Diploid-Locus), Deletions- (CN < 2 für Diploid-Locus) oder kopieneutralen (CN=2 für Diploid-Locus) Ereignisse angibt.

Außerdem berechnet der Scoring-Algorithmus Qualitäts-Scores für exact-copy-number, die als Eingaben für die De-novo-CNV-Erkennungspipeline verwendet werden.

Ausgabedateien

Die DRAGEN-Hostsoftware generiert zahlreiche vorläufige Dateien. Die endgültige Call-Datei mit den Amplifikations- und Deletionsereignissen ist die Datei *.seg.called.merged.

Zusätzlich zu der Segmentdatei gibt DRAGEN die Calls im VCF-Standardformat aus. Die VCF-Datei enthält standardmäßig nur Kopienzahlzunahmeereignisse und -abnahmeereignisse. Segmente ohne Einfluss auf die Kopienzahl sind der Datei *.seg.called.merged zu entnehmen. Legen Sie *--cnv-enable-ref-calls* auf „true“ fest, um Calls ohne Einfluss auf die Kopienzahl (REF) in die Ausgabe-VCF aufzunehmen.

Weitere Informationen zur .seg.called.merged-Datei sowie zur Verwendung der Ausgabedateien bei Debugging und Analyse finden Sie unter *Signalflussanalyse* auf Seite 53.

CNV-VCF-Datei

Die CNV-VCF-Datei entspricht dem VCF-Standardformat. Aufgrund der Darstellungsweise von CNV-Ereignissen im Vergleich zu strukturellen Varianten stehen nicht alle Felder zur Verfügung. Stehen mehr Informationen zu einem Ereignis zur Verfügung, werden diese in der Regel annotiert. Einige Felder der DRAGEN-CNV-VCF-Datei sind spezifisch für CNVs.

Im Folgenden finden Sie ein Beispiel für die CNV-spezifischen Kopfzeilen.

```
##fileformat=VCFv4.1
##ALT=<ID=CNV,Description="Copy number variant region">
##ALT=<ID=DEL,Description="Deletion relative to the reference">
##ALT=<ID=DUP,Description="Region of elevated copy number relative to the
reference">
##contig=<ID=1,length=249250621>
##contig=<ID=2,length=243199373>
##contig=<ID=3,length=198022430>
##contig=<ID=4,length=191154276>
##contig=<ID=5,length=180915260>
…
##reference=file:///reference_genomes/Hsapiens/hs37d5/DRAGEN
##INFO=<ID=REFLEN,Number=1,Type=Integer,Description="Number of REF positions
included in this record">
##INFO=<ID=SVTYPE,Number=1,Type=String,Description="Type of structural variant">
##INFO=<ID=END,Number=1,Type=Integer,Description="End position of the variant
described in this record">
##INFO=<ID=CIPOS,Number=2,Type=Integer,Description="Confidence interval around
POS">
##INFO=<ID=CIEND,Number=2,Type=Integer,Description="Confidence interval around
```

```
END">
##FILTER=<ID=cnvQual,Description="CNV with quality below 10">
##FILTER=<ID=cnvCopyRatio,Description="CNV with copy ratio within +/- 0.2 of
1.0">
##FORMAT=<ID=SM,Number=1,Type=Float,Description="Linear copy ratio of the segment
mean">
##FORMAT=<ID=CN,Number=1,Type=Integer,Description="Estimated copy number">
##FORMAT=<ID=BC,Number=1,Type=Integer,Description="Number of bins in the region">
##FORMAT=<ID=PE,Number=2,Type=Integer,Description="Number of improperly paired
end reads at start and stop breakpoints">
```

Die Spalte ID wird zur Darstellung des Ereignisses verwendet.

In der Spalte REF sind alle CNV-Ereignisse mit einem „N“ aufgeführt.

In der Spalte ALT wird die Art des CNV-Ereignisses angegeben. Da in der VCF-Datei ausschließlich CNV-Ereignisse angegeben sind, wird nur der Eintrag DEL oder DUP verwendet.

Die Spalte QUAL beinhaltet einen geschätzten Qualitäts-Score für das CNV-Ereignis, der für die harte Filterung verwendet wird.

In der Spalte FILTER wird „PASS“ angegeben, wenn das CNV-Ereignis alle Filter passiert hat. Andernfalls enthält diese Spalte den Namen des fehlgeschlagenen Filters.

Die Spalte INFO enthält Informationen zum Ereignis, die größtenteils den Angaben in Spalte ID entsprechen. Der Eintrag REFLEN gibt die Ereignislänge wieder. Der Eintrag SVTYPE lautet stets „CNV“. Der Eintrag END gibt die Endposition des Ereignisses wieder. Die Einträge „CIPOS und CIEND werden derzeit nicht verwendet.

Die FORMAT-Felder werden in der Kopfzeile beschrieben.

- SM: lineares Kopienverhältnis des Segmentmittelwerts
- CN: geschätzte Kopienzahl
- BC: Anzahl der Klassen in der Region
- PE: Anzahl nicht zusammengehöriger Paired-End-Reads an Start- und Stopp-Unterbrechungspunkten

Visualisierung und BigWig-Dateien

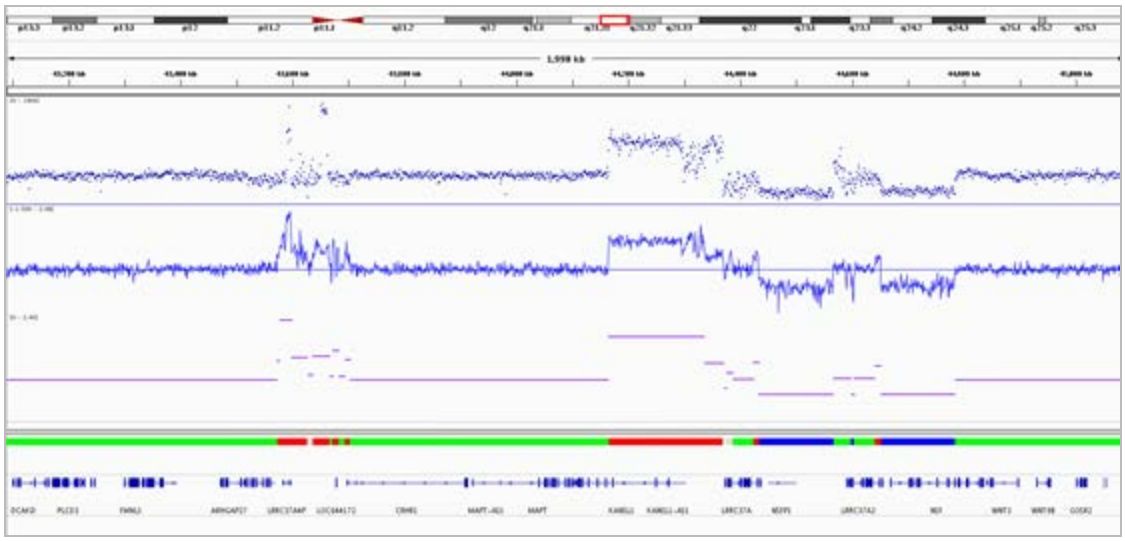
Zum Durchführen einer Analyse mit einem bekannten Referenzsatz können Sie die vorläufigen Ausgabedateien aus den Pipelinephasen verwenden. Eine Analyse dieser Dateien unterstützt Sie bei der Feinabstimmung der Optionen.

Alle Dateien verfügen über eine Struktur, die der einer BED-Datei ähnelt, sowie eine optionale Kopfzeile.

- **.target.counts*: Enthält die Anzahl der Read-Zählungen pro Target-Intervall. Es handelt sich um das Rohsignal, extrahiert aus den Alignments der BAM- oder CRAM-Datei. Das Format ist für die Fallprobe und beliebige Normalgruppen-Proben identisch. Es wird auch eine BigWig-Darstellung einer target.counts.diploid-Datei bereitgestellt. Sie ist auf den normalen Ploidiegrad von 2 normalisiert anstatt auf die Rohzählungen.
- **.tn.tsv*: Das tangential normalisierte Signal der Fallprobe pro Target-Intervall. Diese Datei enthält das logarithmisch normalisierte Signal. Eine starke Signalabweichung von 0,0 deutet auf ein mögliches CNV-Ereignis hin.
- **.seg.called.merged*: Enthält die vom Segmentierungsalgorithmus generierten Segmente.
- **.cnv.vcf*: CNV-VCF-Ausgabedatei mit Hinweis auf Ereignisse.

Legen Sie die Option *--enable-cnv-tracks* auf „true“ fest, um weitere vergleichbare bigwig- und gff-Dateien zu generieren. Diese Dateien können neben anderen verfügbaren Aufzeichnungen wie RefSeq-Genen in IGV geladen werden. Mithilfe dieser und anderer öffentlich verfügbarer Aufzeichnungen ist eine einfachere Interpretation von Calls möglich. In der folgenden Abbildung ist ein Beispiel aufgeführt.

Abbildung 9 IGV-Beispiel



Ausgabe- und Filteroptionen

Mithilfe der Ausgabe- und Filteroptionen lassen sich die CNV-Ausgabedateien steuern.

- *--cnv-blacklist-bed*: Legt eine BED-Datei mit Intervallen fest, die aus der endgültigen CNV-VCF-Ausgabe ausgeschlossen werden sollen. Ein Call wird unterdrückt, wenn dieser mit einem Intervall aus der Blacklist-BED-Datei um mindestens 50 % überlappt.
- *--cnv-enable-plots*: Generiert Diagramme im Rahmen der CNV-Pipeline. Die Standardeinstellung ist „false“.
- Bei einer WGS-CNV-Analyse mit hochauflösenden Intervallen (weniger als 1.000 bp) kann die Diagrammerstellung längere Zeit in Anspruch nehmen. Illumina empfiehlt die Verwendung der Standardeinstellung (deaktiviert).
- *--cnv-enable-ref-calls*: Nimmt Calls ohne Einfluss auf die Kopienzahl (REF) in die VCF-Ausgabedatei auf. Die Standardeinstellung ist „false“.
- *--cnv-enable-tracks*: Erstellt Verfolgungsdateien, die zur Anzeige in IGV importiert werden können. Bei Aktivierung dieser Option werden eine *.gff-Datei für die Ausgabe-Varianten-Calls sowie *.bw-Dateien für das tangential normalisierte Signal generiert. Die Standardeinstellung ist „true“.
- *--cnv-filter-bin-support-ratio*: Filtert ein Kandidatenereignis heraus, wenn die Anzahl unterstützender Klassen weniger als das angegebene Verhältnis in Bezug auf die Ereignisgesamtlänge beträgt. Das Standardverhältnis ist 0.2 (20 % Unterstützung). Beispiel: Wenn das Calling eines Ereignisses mit einer Länge von 100.000 bp erfolgt, die den Call unterstützenden Target-Intervall-Klassen jedoch nur insgesamt 15.000 bp umfassen (15.000/100.000 = 0,15), wird das Ereignis herausgefiltert.
- *--cnv-filter-copy-ratio*: Gibt den Schwellenwert für das über 1,0 gemittelte Kopienverhältnis an, ab dem ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 0.2. Das Ergebnis sind Calls mit einem Wert von unter CR=0.8 oder über CR=1.2.

- *--cnv-filter-length*: Gibt die minimale Ereignislänge in Basen an, ab der ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 10000.
- *--cnv-filter-qual*: Gibt den QUAL-Wert an, ab dem ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 10.
- *--cnv-min-qual*: Gibt den kleinsten gemeldeten QUAL-Wert an. Der Standardwert ist 3.
- *--cnv-max-qual*: Gibt den größten gemeldeten QUAL-Wert an. Der Standardwert ist 200.
- *--cnv-ploidy*: Gibt den normalen Ploidiewert an. Diese Option wird nur zur Schätzung des Kopienzahlwerts verwendet, der in der VCF-Ausgabedatei ausgegeben wird. Der Standardwert ist 2.
- *--cnv-qual-length-scale*: Gibt den Gewichtungsfaktor der Verzerrung an, um QUAL-Schätzungen für längere Segmente anzupassen. Diese erweiterte Option sollte nicht geändert werden. Der Standardwert ist 0.9303 (2-0.1).
- *--cnv-qual-noise-scale*: Gibt den Gewichtungsfaktor der Verzerrung an, um QUAL-Schätzungen auf Grundlage der Probenvarianz anzupassen. Diese erweiterte Option sollte nicht geändert werden. Der Standardwert ist 1.0.

Gleichzeitiges CNV- und Haplotyp-Varianten-Calling

DRAGEN kann FASTQ-Proben mappen und alignieren und die Daten im Anschluss direkt an die nachgeschalteten Caller weitergeben. Eine einzelne Probe kann sowohl das CNV- als auch das Haplotyp-VC durchlaufen, wenn die Eingabe als FASTQ-Probe erfolgt. In diesem Fall erfolgt standardmäßig eine Selbstnormalisierung.

Führen Sie die FASTQ-Probe in einem regulären DRAGEN-Mapping-Alignment-Workflow aus und geben Sie dann zusätzliche Argumente ein, um CNV, VC oder beide Optionen zu aktivieren. Die für eigenständige CNV-Workflows geeigneten Optionen können auch hier verwendet werden.

In den folgenden Beispielen sind weitere Befehle aufgeführt.

FASTQ mit CNV mappen/alignieren

```
dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
   -2 <FASTQ2> \
   --RGSM <PROBE> \
   --RGID <RGID> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true \
   --enable-cnv true \
   --cnv-enable-self-normalization true
```

FASTQ mit VC mappen/alignieren

```
dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
   -2 <FASTQ2> \
   --RGSM <PROBE> \
   --RGID <RGID> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true \
   --enable-variant-caller true
```

FASTQ mit CNV und VC mappen/alignieren

```
dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
   -2 <FASTQ2> \
   --RGSM <PROBE> \
   --RGID <RGID> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align true \
   --enable-cnv true \
   --cnv-enable-self-normalization true \
   --enable-variant-caller true
```

BAM-Eingabe für CNV und VC

```
dragen \
   -r <HASHTABELLE> \
   --bam-input <BAM> \
   --output-directory <AUSGABE> \
   --output-file-prefix <PROBE> \
   --enable-map-align false \
   --enable-cnv true \
   --cnv-enable-self-normalization true \
   --enable-variant-caller true
```

Korrelation und Geschlechtsgenotypisierung von Proben

Beim Ausführen der Target-Zählung oder Normalisierung stellt die DRAGEN-CNV-Pipeline auch die folgenden Informationen über die Proben im Lauf bereit.

- Eine Korrelationsmetrik des Read-Anzahlprofils zwischen der Fallprobe und beliebigen Normalgruppenproben. Für eine zuverlässige Analyse wird eine Korrelationsmetrik größer als 0.90 empfohlen, von der Software wird jedoch keine strikte Einschränkung erzwungen.

- Das prognostizierte Geschlecht jeder Probe im Lauf. Das Geschlecht wird auf Grundlage der Informationen über die Read-Anzahl in den Geschlechtschromosomen und den autosomalen Chromosomen prognostiziert. Der Median für die Zählungen wird auf dem Bildschirm für die autosomalen Chromosomen, das X-Chromosom und das Y-Chromosom angezeigt.

Die Ergebnisse werden beim Ausführen der Pipeline auf dem Bildschirm angezeigt. Beispiel:

```
==========================================
Correlation Table
==========================================
Correlation of case sample PlatinumGenomes_50X_NA12877 against
PlatinumGenomes_50X_NA12878: 0.984092

Sex Genotyper
==========================================
Predicted sex of samples
PlatinumGenomes_50X_NA12877: MALE XY 0.99737
PlatinumGenomes_50X_NA12878: FEMALE XX 0.968929
```

Um die Analyse der Geschlechtschromosomen mithilfe einer Normalgruppe durchzuführen, wird die Verwendung von dem Geschlecht entsprechenden Proben in der Normalgruppe empfohlen.

Mit der Option *–sample-sex* können Sie das Geschlecht der Probe überschreiben.

Mehrproben-CNV-Calling

Das Mehrproben-CNV-Calling ist ausgehend von tangential normalisierten Zählungsdateien (*.tn.tsv) möglich, die mit der Option *--cnv-input* angegeben werden (eine pro Probe). Die Mehrproben-CNV-Analyse profitiert von der Verwendung gemeinsamer Segmentierung, um die Sensitivität für die Erkennung von Kopienzahl-Variablensegmenten zu erhöhen. Für jedes identifizierte Kopienzahl-Variablensegment wird der Genotyp der Kopienzahl der jeweiligen Probe in einem einzigen VCF-Eintrag ausgegeben, um Annotation und Interpretation zu erleichtern.

Das folgende Befehlszeilenbeispiel bezieht sich auf die Durchführung einer Trio-Analyse:

```
dragen \
-r <HASHTABELLE> \
--output-directory <AUSGABE> \
--output-file-prefix <PROBE> \
--enable-cnv true \
--cnv-input <VATER_TN_TSV> \
--cnv-input <MUTTER_TN_TSV> \
--cnv-input <PROBAND_TN_TSV> \
--pedigree-file <STAMMBAUMDATEI>
```

Optionen für das De-novo-CNV-Calling

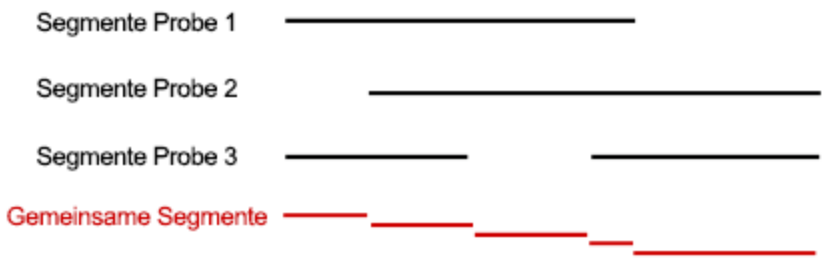
Die folgenden Optionen werden für das De-novo-CNV-Calling verwendet:

- *--cnv-input*: Beim De-novo-CNV-Calling gibt diese Option die tangential normalisierten Signaleingangsdateien (*.tn.tsv) aus den Einzelprobenläufen an. Diese Option kann mehrmals angegeben werden, für jede Eingangsprobe einmal.
- *--cnv-filter-de-novo-qual*: Phred-skalierter Schwellenwert, bei dem ein putatives Ereignis in der Probandenprobe mit „DeNovo“ gekennzeichnet wird. Der Standardwert ist 0.10.

- *--pedigree-file*: Stammbaumdatei, die die Beziehung zwischen den Eingangsproben angibt.

Gemeinsame Segmentierung

Im ersten Schritt erfolgt das CNV-Calling separat für die einzelnen Proben. Anschließend wird bei der gemeinsamen Segmentierung anhand der Kopienzahlvariablen-Segmente aus den einzelnen Probenanalysen eine Gruppe gemeinsamer Kopienzahlvariablen-Segmente ermittelt. Diese Gruppe wird einfach aus der Schnittmenge aller Unterbrechungspunkte aus den Kopienzahlvariablen-Segmenten sämtlicher Proben gebildet. Dadurch werden alle zwischen den unterschiedlichen Proben teilweise überlappenden Segmente getrennt. Beispiel:



Nach der gemeinsamen Segmentierung wird das Kopienzahl-Calling anhand der gemeinsamen Segmente erneut separat für jede einzelne Probe durchgeführt. Die Segmente können wie bei der Analyse einer Einzelprobe zusammengefasst werden, jedoch werden alle gemeinsamen Segmente als einzelner Eintrag in die Mehrproben-VCF ausgegeben. Der Qualitäts-Score (QS in der VCF) des zusammengefassten Probensegments wird ggf. für die Call-Filterung verwendet. Proben-Calls werden anhand des FT-Felds der Probe in der Mehrproben-VCF gefiltert. Die QUAL-Spalte der Mehrproben-VCF enthält in keinem Fall Werte (ist also „.“). Die FILTER-Spalte der Mehrproben-VCF enthält den Wert „SampleFT“, wenn keines der FT-Felder der Probe „PASS“ enthält, und enthält „PASS“, wenn mindestens eines der FT-Felder der Probe „PASS“ enthält.

De-novo-Calling-Phase

Ein De-novo-Ereignis ist definiert als das Vorhandensein eines Genotyps an einem bestimmte Locus im Genom eines Probanden, wobei dieser Genotyp ohne Mendelsche Vererbung der Eltern entstanden ist. In der De-novo-Calling-Phase werden putative De-novo-Ereignisse in der Probandenprobe von jedem Trio einer Mehrproben-Analyse identifiziert. In einigen Fällen sind diese putativen De-novo-Ereignisse echt, sie können jedoch auch von Sequenzierungs- oder Analyseartefakten stammen. Daher wird jedem putativen De-novo-Ereignis ein De-novo-Qualitäts-Score zugewiesen, um De-novo-Ereignisse mit niedriger Qualität herausfiltern zu können. Trios werden durch Angabe einer .ped-Datei mit der Option *--pedigree-file* spezifiziert. Es können mehrere Trios angegeben werden (z. B. 4er-Analyse). Alle gültigen Trios werden verarbeitet.

Bei jedem gemeinsamen Segment in einem Trio bestimmt der De-novo-Caller, ob für die aufgerufenen Kopienzahl-Genotypen ein Mendelscher Vererbungskonflikt vorliegt. Der CNV-Caller ermittelt nicht die Kopienzahl sämtlicher Allele in einem gegebenen diploiden Segment, d. h., es werden nur Annahmen über die mögliche Allelzusammensetzung der Eltern-Genotypen getroffen.

Angenommen wird, dass das Allel mit der Kopienzahl 0 für diploide Regionen eines elterlichen Genoms (geschlechtsspezifisch) nicht vorhanden ist, sofern die zugewiesene Kopienzahl größer oder gleich 2 ist.

Dies führt zu folgenden Vereinfachungen:

Kopienzahl-Genotyp des Elternteils	Mögliche Kopienzahl-Allele	Angenommene mögliche Kopienzahl-Allele
2	0/2, 1/1	1/1
3	0/3, 1/2	1/2
4	0/4, 1/3, 2/2	1/3, 2/2
N	x/(N-x) für x <= N/2	x/(N-x) für 1 <= x <= N/2

Im Folgenden sind Beispiele mit konsistenten und inkonsistenten Kopienzahl-Genotypen für diploide Regionen unter diesen Annahmen aufgeführt:

Kopienzahl der Mutter	Kopienzahl des Vaters	Kopienzahl des Probanden	Konsistent mit Mendelscher Vererbung?
2	2	2	Ja
2	2	1	Nein
3	2	4	Nein
3	2	2	Ja

Liegt ein Mendelscher Vererbungskonflikt vor, wird ein Phred-skalierter De-novo-Qualitäts-Score (Feld DQ in der VCF-Datei) mithilfe der Wahrscheinlichkeit für jeden Kopienzahlstatus (siehe Abschnitt „Qualitätsbewertung“) jeder Probe im Trio in Kombination mit einer A-priori-Wahrscheinlichkeit für die Trio-Genotypen berechnet:

```
DQ = -10log(1-Sum over conflicting genotypes(p(CNm|data)*p(CNf|data)*p
   (CNp|data)*p(CNm,CNf,CNp))/Sum over all genotypes(p(CNm|data)*p
   (CNf|data)*p(CNp|data)*p(CNm,CNf,CNp)))
```

Wobei gilt:

- CNm = Kopienzahl der Mutter
- CNf = Kopienzahl des Vaters
- CNp = Kopienzahl des Probanden
- p(CNm,CNf,CNp) = A-priori-Wahrscheinlichkeit für den Trio-Genotyp

Das Feld DN in der VCF-Datei verweist auf den De-novo-Status des jeweiligen Segments. Mögliche Werte:

- Inherited: aufgerufener Trio-Genotyp ist mit der Mendelschen Vererbung konsistent
- LowDQ: aufgerufener Trio-Genotyp ist mit der Mendelschen Vererbung nicht konsistent und DQ liegt unter dem De-novo-Qualitätsschwellenwert (Standard: 0.1)
- DeNovo: aufgerufener Trio-Genotyp ist mit der Mendelschen Vererbung nicht konsistent und DQ liegt über dem De-novo-Qualitätsschwellenwert (Standard: 0.1)

Mehrproben-CNV-VCF-Ausgabe

Die Datensätze einer Mehrproben-CNV-VCF unterscheiden sich leicht von den Datensätzen im Fall einer Einzelprobe. Die Hauptunterschiede sind folgende:

- Die Einträge pro Datensatz werden den Unterbrechungspunkten in der Gesamtmenge aller eingegebenen Proben entsprechend in Segmente aufgeteilt. Die VCF enthält demnach insgesamt mehr Einträge.

- Die QUAL-Spalte wird nicht verwendet, ihr Wert ist „.“. Die Qualität pro Probe wird mit dem QS-Tag in die SAMPLE-Spalten übertragen.
- Die FILTER-Spalte zeigt PASS an, wenn eine der einzelnen SAMPLE-Spalten den Wert PASS aufweist. Trifft dies nicht zu, zeigt sie SampleFT an.
- Die Annotationen pro Probe werden aus ihren jeweiligen Herkunfts-Calls übertragen. Die Einzelprobenfilter werden auf der Probenebene angewendet und in der FT-Annotation ausgegeben.

Bei Verwendung einer gültigen Stammbaumdatei wird zudem ein De-novo-Calling durchgeführt. Der Probandenprobe werden dabei die folgenden zwei Annotationen hinzugefügt:

```
##FORMAT=<ID=DQ,Number=1,Type=Float,Description="De novo quality">
##FORMAT=<ID=DN,Number=1,Type=String,Description="Possible values are
   'Inherited', 'DeNovo' or 'LowDQ'. Threshold for a passing de novo call
   is DQ > 0.100000">
```

Die VCF enthält viele Einträge. Aufgrund der gemeinsamen Segmentierung lässt sich die Anzahl an De-novo-Ereignissen jedoch durch das Extrahieren von Einträgen mit DN- und DQ-Annotation ermitteln. Diese Datensätze werden ebenfalls extrahiert und im Fall von De-novo-Calls ins GFF3-Format konvertiert.

Repeat-Expansion-Bestimmung mit Expansion Hunter

STRs (Short Tandem Repeats) sind Regionen im Genom, die aus Wiederholungen kurzer DNA-Segmente bestehen, die als Repeat-Einheiten bezeichnet werden. STRs können die normale Länge überschreiten und dadurch Mutationen verursachen, die als Repeat-Expansionen bezeichnet werden. Repeat-Expansionen sind die Ursache zahlreicher Erkrankungen, darunter Fragiles-X-Syndrom, amyotrophe Lateralsklerose und Chorea Huntington.

DRAGEN enthält ein Verfahren zur Bestimmung von Read-Expansionen, das als ExpansionHunter bezeichnet wird. Dieses Verfahren erfolgt anhand eines genauen Realignments der Reads in und um jedes Target-Repeat anhand des Sequenzdiagramms. Anschließend erfolgt die Genotypisierung der Repeat-Länge in den einzelnen Allelen anhand dieser Diagramm-Alignments. Weitere Informationen und Analysen sind folgenden Dokumenten zu ExpansionHunter zu entnehmen:

- ExpansionHunter (ursprüngliche Version)
- Graph ExpansionHunter (neue, in DRAGEN integrierte Version)

Beachten Sie bitte, dass diese Verfahren nur für Humangesamtgenom-Proben angewendet werden können, die mit Verfahren ohne PCR generiert wurden. Die Genotypisierung von Repeats erfolgt nur bei mindestens 10-facher Coverage am Locus.

Optionen für die Repeat-Expansion-Bestimmung

Mit folgenden Befehlszeilenoptionen lässt sich die Repeat-Expansion-Bestimmung von DRAGEN aktivieren.

- --repeat-genotype-enable = true
- --repeat-genotype-specs=<Pfad zur Spezifikationsdatei>

Zusätzlich lässt sich mit der Option *--sample-sex* das Geschlecht der Probe festlegen.

Die folgenden Optionen sind optional.

- --repeat-genotype-region-extension-length=<Länge der um das Repeat zu untersuchenden Region> (Standardwert: 1000 bp)
- --repeat-genotype-min-baseq=<Minimale Basenqualität für Basen mit hoher Konfidenz> (Standardwert: 20)

Weitere Informationen zur Spezifikationsdatei, die mit der Option *--repeat-genotype-specs* angegeben wird, finden Sie unter *Spezifikationsdateien für Repeat-Expansionen* auf Seite 74.

Bei der Hauptausgabe der Repeat-Expansionserkennung handelt es sich um eine VCF-Datei, die die mit dieser Analyse ermittelten Varianten enthält.

Spezifikationsdateien für Repeat-Expansionen

Die JSON-Datei mit der Repeat-Spezifikation (auch als Variantenkatalog bezeichnet) legt die zu analysierenden Repeat-Regionen für ExpansionHunter fest. Die Standard-Repeat-Spezifikation für bestimmte pathogene Repeats befindet sich im Verzeichnis /opt/edico/repeat-specs/ (abhängig vom für DRAGEN verwendeten Referenzgenom).

Sie können Spezifikationsdateien für neue Repeat-Regionen mithilfe einer der bereitgestellten Spezifikationsdateien als Vorlage erstellen. Ausführliche Informationen zum Format finden Sie in der Dokumentation zu ExpansionHunter.

Ausgabedateien für die Repeat-Expansion-Bestimmung

VCF-Ausgabedatei

Die Ergebnisse von Repeat-Genotypisierungen werden als einzelne VCF-Dateien ausgegeben, die die Länge jedes Allels bei jedem callfähigen Repeat gemäß Definition in der Katalogdatei repeat-specification enthalten. Der Dateiname lautet <Ausgabepräfix>.repeats.vcf (.gz).

Die VCF-Ausgabedatei beginnt mit den folgenden Feldern.

Tabelle 3 VCF-Kernfelder

Feld	Beschreibung
CHROM	Identifikator für Chromosomen
POS	Position der ersten Base vor der Repeat-Region in der Referenz
ID	Immer „.“
REF	Die Referenzbase mit der Position POS
ALT	Liste der Repeat-Allele im Format <STRn>, wobei n für die Anzahl an Repeat-Einheiten steht
QUAL	Immer „.“
FILTER	Immer PASS

Tabelle 4 Zusätzliche INFO-Felder

Feld	Beschreibung
SVTYPE	Immer STR
END	Position der letzten Base der Repeat-Region in der Referenz
REF	Anzahl der Repeat-Einheiten innerhalb des Repeats in der Referenz
RL	Referenzlänge in bp
RU	Repeat-Einheit in der Referenzausrichtung
REPID	Repeat-ID aus der Datei repeat-specification

Tabelle 5 GENOTYP-Felder (je Probe)

Feld	Beschreibung
GT	Genotyp
SO	Das Allel unterstützende Read-Typen; mögliche Typen sind SPANNING, FLANKING oder INREPEAT, d. h., die Reads umspannen das Repeat, grenzen an ihn an oder sind vollständig darin enthalten
CI	Konfidenzintervall bzw. Repeat-Länge eines Allels
AD_SP	Anzahl an umspannenden Reads, die dem Allel entsprechen
AD_FL	Anzahl an angrenzenden Reads, die dem Allel entsprechen
AD_IR	Anzahl an Reads innerhalb des Repeats, die dem Allel entsprechen

Die folgende VCF-Angabe beschreibt z. B. das Repeat C9orf72 in einer Probe mit der ID LP6005616-DNA_A03.

```
QUAL    FILTER  INFO    FORMAT  LP6005616-DNA_A03
chr9    27573526        .       C       <STR2>,<STR349> .       PASS
SVTYPE=STR;END=27573544;REF=3;RL=18;RU=GGCCCC;REPID=ALS GT:SO:CN:CI:AD_SP:AD_
FL:AD_IR
1/2:SPANNING/INREPEAT:2/349:2-2/323-376:19/0:3/6:0/459
```

In diesem Beispiel umspannt das erste Allel 2 Repeat-Einheiten, das zweite Allel umspannt 349 Repeat-Einheiten. Die Repeat-Einheit lautet GGCCCC (Feld RU INFO), die Sequenz des ersten Allels ist demnach GGCCCCGGCCCC und die Sequenz des zweiten Allels ist GGCCCC x 349. Das Repeat umspannt drei Repeat-Einheiten in der Referenz (Feld REF INFO).

Die Länge des kurzen Allels wurde aus den umspannenden Reads ermittelt (SPANNING), die Länge des erweiterten Allels wurde aus Reads innerhalb des Repeats ermittelt (INREPEAT). Das Konfidenzintervall für die Größe des erweiterten Allels ist (323,376). Es wurden 19 umspannende und 3 angrenzende Reads ermittelt, die dem Repeat-Allel der Größe 2 entsprechen (d. h., 19 Reads enthalten das Repeat der Größe 2 vollständig und 2 angrenzende Reads überschneiden sich mit höchstens 2 Repeat-Einheiten). Zudem wurden 6 angrenzende Reads und 459 Reads innerhalb des Repeats ermittelt, die dem Repeat-Allel der Größe 349 entsprechen.

Zusätzliche Ausgabedateien

Die neuen Sequenzdiagramm-Alignments von Reads in den Repeat-Zielregionen werden in einer BAM-Datei ausgegeben. Die Alignment-Position eines Reads wird auf die Position in der Repeat-Region festgelegt, auf die dieser neu ausgerichtet wurde. Das vollständige Alignment (CIGAR) in Bezug auf das Sequenzdiagramm wird in einem benutzerdefinierten XG-Tag mit dem Format <LocusName>,<StartPosition>,<DiagrammCIGAR> angegeben.

StartPosition stellt die erste Alignment-Position des Reads im ersten Knoten dar, DiagrammCIGAR beschreibt das Alignment in Bezug auf das Diagramm von diesem Punkt aus. Weitere Informationen finden Sie unter https://git.illumina.com/Bioinformatics/graph-tools/blob/master/docs/alignment.md. Qualitäts-Scores in dieser BAM-Datei werden in eine Binärdarstellung überführt. Zur Genotypisierung mithilfe eines Diagramms werden ein hoher Score (40) und ein niedriger Score (0) verwendet.

Calling für spinale Muskelatrophie

Ursache der spinalen Muskelatrophie (SMA) ist eine Disruption aller Kopien des SMN1-Gens. Von dieser progressiven Erkrankung ist etwa eine von 10.000 Lebendgeburten betroffen. SMN1 verfügt mit SMN2 über ein Paralog mit sehr hoher Identität mit einer Abweichung von nur etwa 10 SNVs und kleinen Indels. Eine

Kopie (hg19 chr5:70247773 C->T) wirkt sich auf das Spleißen aus und unterbricht die Produktion des funktionalen SMN-Proteins aus SMN2. Bei der WGS-Standardanalyse wird kein vollständiges Ergebnis des Varianten-Callings für SMN erstellt. Dies liegt an der hohen Ähnlichkeit der Duplikation in Kombination mit häufigen Kopienzahlvarianten. Schätzungsweise können jedoch 95 % der SMA-Fälle erkannt werden, indem das Fehlen des funktionalen C (SMN1)-Allels in einer beliebigen Kopie von SMN bestimmt wird.

DRAGEN verwendet das Sequenzdiagramm-Realignment (wie beim Repeat-Expansion-Calling), um Reads mit einer einzelnen Referenz, die SMN1 und SMN2 angibt, zu alignieren. Zusätzlich zum standardmäßigen diploiden Genotyp-Calling prüft das Programm anhand eines direkten statistischen Tests das Vorhandensein von C-Allelen. Wird kein C-Allel erkannt, ist die Probe betroffen. Andernfalls ist die Probe nicht betroffen.

SMA-Calling wird nur für Humangesamtgenom-Sequenzierungsproben mit Bibliotheken ohne PCR unterstützt.

Verwendung

Die SMA-Calling-Implementierung erfolgt zusammen mit der Repeat-Expansion-Erkennung. Weitere Informationen zum Diagramm-Alignment und entsprechende Optionen finden Sie unter *Repeat-Expansion-Bestimmung mit Expansion Hunter* auf Seite 73.

Das SMA-Calling wird zusammen mit der Repeat-Expansion-Bestimmung aktiviert, indem die Option *--repeat-genotype-enable* auf „true“ festgelegt wird. Die Datei mit dem Variantenspezifizierungskatalog muss zur Aktivierung von SMA-Calling eine Beschreibung der betreffenden SMN1/2-Variante enthalten. Beispieldateien sind im Ordner /opt/edico/repeat-specs/experimental verfügbar.

SMN-Ausgabe und betroffene Repeats sind in der Datei <Ausgabepräfix>.repeat.vcf enthalten. Die SMN-Ausgabe ist ein einzelner SNV-Call an der Schlüsselposition (mit Auswirkung auf das Spleißen) in SMN1, wobei der SMA-Status in benutzerdefinierten Feldern enthalten ist:

Tabelle 6 SMA-Ergebnis in repeat.vcf-Ausgabe

Feld	Beschreibung
VARID	SMN kennzeichnet den SMN-Call.
GT	Genotyp-Call an dieser Position unter Verwendung eines normalen (diploiden) Genotypmodells.
DST	SMA-Status-Call: + steht für erkannt, - steht für nicht erkannt, ? steht für unbestimmt.
AD	Gesamtzahl der Reads, die das C- und T-Allel unterstützen.
RPL	Log10-Likelihood-Quotient zwischen den betroffenen und nicht betroffenen Modellen. Positive Scores zählen zu den nicht betroffenen Modellen.

Calling struktureller Varianten

In DRAGEN sind die Verfahren des Manta Structural Variant Caller integriert. Eine Beschreibung dieser Verfahren finden Sie in der Dokumentation zu Manta.

Das Calling struktureller Varianten (SVs) und Indels erfolgt anhand von gemappten Paired-End-Sequenzierungs-Reads. Der SV-Caller ist für die Analyse von Keimbahnvarianten in kleinen Personengruppen und somatischen Varianten in Tumor-Normal-Probenpaaren optimiert.

Der SV-Caller leistet Folgendes:

- Bestimmung, Assemblierung und Scoring umfangreicher SVs, mittelgroßer Indels und großer Insertionen in einem einzigen effizienten Workflow.
- Kombination von Paired- und Split-Read-Evidenz während SV-Bestimmung und -Scoring zur Optimierung der Genauigkeit. Zur Meldung von Varianten sind keine Split-Reads oder erfolgreichen Unterbrechungspunkt-Assemblierungen erforderlich, wenn die sonstige Evidenz hoch ist.

- Stellt Scoring-Modelle für Keimbahnvarianten in kleinen Gruppen von Diploid-Proben und somatische Varianten in übereinstimmenden Tumor-Normal-Probenpaaren bereit.

Zusätzlich ist eine experimentelle Unterstützung für die Analyse nicht übereinstimmender Tumorproben vorhanden. Alle SV- und Indel-Bestimmungen werden im Format VCF 4.1 ausgegeben.

Überblick über Manta

Bei Manta erfolgt die SV- und Indel-Erkennung in zwei primären Schritten: (1) Scan des Genoms zum Auffinden von SV-zugeordneten Regionen (2) Analyse, Scoring und Ausgabe von in solchen Regionen gefundenen SVs.

1 Erstellen des Bruchenden-Assoziationsdiagramms: Das gesamte Genom wird gescannt, um mögliche SVs und große Indels nachzuweisen. Die Nachweise werden in einem Diagramm aufgeführt, dessen Ränder alle Genomregionen mit möglichen Bruchendenassoziationen verknüpfen. Ränder können dabei zwei verschiedene Genomregionen verknüpfen (Beleg für eine Long-Range-Assoziation) oder ein Rand verknüpft eine Region mit sich selbst, um eine lokale Indel-/kleine SV-Assoziation zu erfassen. Diese Assoziationen sind allgemeiner als eine spezifische SV-Hypothese, in der möglicherweise viele potenzielle Bruchenden an einem Rand gefunden werden, obwohl es in der Regel nur ein oder zwei pro Rand sind.

2 Analysieren der Diagrammränder zum Auffinden von SVs: Anhand einer Analyse von einzelnen Diagrammrändern oder Gruppen hochgradig verknüpfter Ränder können Sie den Rändern zugeordnete SVs entdecken und bewerten. Die Unterschritte dieses Prozesses umfassen Folgendes:
   - Bestimmung von potenziellen SVs, die dem Rand zugeordnet sind.
   - Versuchte Assemblierung der SV-Bruchenden.
   - Scoring/Genotypisierung und Filterung von SVs anhand verschiedener biologischer Modelle (derzeit: diploid, Keimbahn und somatisch).
   - Ausgabe als VCF.

Funktionen von Manta

Manta kann sämtliche Typen struktureller Varianten erkennen, die sich ohne Kopienzahlanalyse und umfangreiche De-novo-Assemblierung bestimmen lassen. Weitere Informationen zu bestimmbaren Typen finden Sie unter *Erkannte Variantenklassen* auf Seite 78.

Für jede strukturelle Variante und jedes Indel versucht Manta, die Bruchenden zu Basenpaaren zu assemblieren und die nach links versetzte Bruchendenkoordinate (gemäß VCF 4.1 SV-Berichtsrichtlinien) gemeinsam mit allen Bruchendenhomologiesequenzen und/oder zwischen Bruchenden eingefügten Sequenzen zu melden. In vielen Fällen lassen sich Daten mit der Assemblierung nicht zuverlässig bestimmen. In diesem Fall wird die Variante als IMPRECISE angegeben und erhält den Score ausschließlich anhand der Evidenz aus dem Paired-End-Read.

Die als Eingabe in Manta bereitgestellten Sequenzierungs-Reads müssen aus einem Paired-End-Sequenzierungs-Assay mit einer „innie“-Ausrichtung zwischen zwei Reads jedes Sequenzfragments stammen, wobei dieser jeweils einen Read vom äußeren Rand des Fragment-Inserts nach innen darstellt.

Manta wurde hauptsächlich hinsichtlich Gesamtgenom- und Gesamtexom-Sequenzierungs-Assays (bzw. Sequenzierungs-Assays mit sonstiger Target-Anreicherung) in DNA getestet. Für diese Assays werden folgende Anwendungen unterstützt:

- Gemeinsame Analyse kleiner Diploid-Personengruppen (klein bedeutet hierbei auf Familienebene mit ca. 10 oder weniger Proben)
- Subtraktive Analyse eines übereinstimmenden Tumor-Normal-Probenpaares
- Analyse einer individuellen Tumorprobe

Es gibt keine spezifischen Einschränkungen hinsichtlich des Einsatzes von Manta für die gemeinsame Analyse größerer Kohorten. Dies wurde jedoch nicht ausgiebig getestet. Probleme mit der Stabilität und der Call-Qualität sind daher nicht auszuschließen.

Tumorproben können ohne übereinstimmende Normalprobe analysiert werden. In diesem Fall steht keine Scoring-Funktion zur Verfügung, die zugrunde liegende Evidenz-Zählung und zahlreiche Filter lassen sich jedoch trotzdem sinnvoll einsetzen.

Erkannte Variantenklassen

Manta kann alle Variationsklassen erkennen, die als neue DNA-Adjazenzen im Genom erklärt werden können. Einfache Insertions-/Deletionsereignisse können bis zu einem Cutoff mit konfigurierbarer Mindestgröße erkannt werden (Standard 8). Alle neuen DNA-Adjazenzen werden basierend auf dem Bruchendenmuster in folgende Kategorien eingestuft:

- Deletionen
- Insertionen
  - Vollständig assemblierte Insertionen
  - Teilweise assemblierte (d. h. bestimmte) Insertionen
- Tandem-Duplikationen
- Nicht eingestufte Bruchendenpaare, die Intra- und Inter-Chromosomentranslokationen entsprechen, bzw. komplexe strukturelle Varianten.

Bekannte Einschränkungen

Manta kann folgende Variantentypen nicht bestimmen:

- Vereinzelte Duplikationen.
- Die meisten Expansions-/Kontraktionsvarianten von Referenztandem-Repeats.
- Bruchenden in Zusammenhang mit kleinen Inversionen.
  - Der Grenzwert wurde nicht ermittelt, jedoch nimmt theoretisch die Erfassungsleistung unterhalb von ca. 200 Basen ab. Sogenannte Mikroinversionen lassen sich möglicherweise indirekt als kombinierte Insertions-/Deletionsvarianten bestimmen.
- Vollständig assemblierte große Insertionen.
  - Die maximale Größe vollständig assemblierter Insertionen kann ungefähr der doppelten Read-Paarfragmentgröße entsprechen, jedoch fällt die Fähigkeit zur vollständigen Assemblierung der Insertion bereits unterhalb dieser Größe auf unzulängliche Werte.
  - Manta erfasst und meldet extrem große Insertionen, wenn die Bruchendsignatur eines derartigen Ereignisses ermittelt wird, auch wenn die eingefügte Sequenz nicht vollständig assembliert werden kann.

Weitere Repeat-abhängige Einschränkungen gelten für alle Variantentypen:

- Die Fähigkeit zur Assemblierung von Varianten in der Bruchendenauflösung tendiert mit Anwachsen der Bruchenden-Repeat-Länge auf die Read-Größe gegen null.
- Die Fähigkeit zur Bestimmung von Bruchenden fällt mit Anwachsen der Bruchendwiederholungslänge auf die Fragmentgröße (praktisch) auf null.

Manta klassifiziert zwar einige neue DNA-Adjazenzen, bestimmt jedoch keine Konstrukte höherer Ordnung in dieser Klassifikation. Beispielsweise weist eine von Manta als Deletion gekennzeichnete Variante auf eine

intrachromosomale Translokation mit einem einer Deletion entsprechenden Bruchendmuster hin, jedoch werden Tiefe, B-Allelfrequenz oder kreuzende Adjazenzen nicht zur direkten Bestimmung des SV-Typs getestet.

Eingabeanforderungen

Wenn Manta im eigenständigen Modus ausgeführt wird, müssen für die Eingabe bereitgestellte Sequenzierungs-Reads aus einem Paired-End-Sequenzierungs-Assay mit einer „innie"-Orientierung zwischen den zwei Reads des jeweiligen DNA-Fragments stammen, wobei diese jeweils einen Read vom äußeren Rand des Fragment-Inserts nach innen darstellen.

Manta akzeptiert nicht gepaarte Reads in der Eingabe, solange ausreichend Paired-End-Reads vorhanden sind, um die Größenverteilung der gepaarten Fragmente zu bestimmen. Nicht gepaarte Reads werden bei Bestimmung, Assemblierung und Split-Read-Scoring weiterhin verwendetet, wenn deren Alignments (oder Split-Alignments mit SA-Tag) große Indels bzw. SV unterstützen oder Nichtübereinstimmung/Clipping auf ein mögliches Bruchende hinweist.

Manta erfordert, dass Eingabesequenzierungs-Reads von einem externen Tool gemappt und im BAM- oder CRAM-Format als Eingabe bereitgestellt werden. Jede Datei muss nach Koordinaten geordnet und indiziert sein, damit ein Index in einem samtools-/htslib-Format in einer Datei angelegt werden kann, deren Name dem der BAM- bzw. CRAM-Datei mit der zusätzlichen Erweiterung „.bai", „.crai" oder „.csi" entspricht.

Bei der Konfiguration muss mindestens eine BAM- oder CRAM-Datei für die Normal- oder die Tumorprobe angegeben werden. Außerdem kann ein entsprechendes Tumor-Normal-Probenpaar bereitgestellt werden. Wenn mehrere Eingabedateien für die Normalprobe bereitgestellt werden, werden die Dateien als Einzelproben einer gemeinsamen Analyse diploider Proben behandelt.

Für die für Manta bereitgestellten BAM- und CRAM-Eingabedateien gelten die folgenden Einschränkungen:

- Alignments dürfen keine unbekannte Read-Sequenz (SEQ=„*") aufweisen.
- Alignments dürfen im Feld „SEQ" nicht das Zeichen „=" enthalten.
- Alignments dürfen nicht die Sequenzübereinstimmung/-nichtübereinstimmung verwenden („="/„X"). RG-Tags (Read-Gruppe) der CIGAR-Notation im Alignment-Datensatz werden ignoriert. Einzelne Dateien werden als Einzelproben behandelt.
- Alignments mit Basecall-Qualitätswerten über 70 werden zurückgewiesen. (Diese werden als Versatzfehler gewertet und daher nicht unterstützt.)

Außerdem erfordert Manta eine Referenzsequenz im FASTA-Format. Hierbei muss es sich um dieselbe Referenz handeln, die für das Mapping der Eingabe-Alignment-Dateien verwendet wird. Die Referenz muss einen Index in einem samtools-/htslib-Format in einer Datei enthalten, deren Name dem der Eingabe-FASTA mit der zusätzlichen Erweiterung .fai entspricht.

Optionen für das Calling struktureller Varianten

Für den Structural Variant Caller werden folgende Befehlszeilenoptionen unterstützt:

- *--enable-sv*: Aktiviert/deaktiviert den Structural Variant Caller. Die Standardeinstellung ist „false".
- *--sv-reference*: Gibt eine Referenzdatei im FASTA-Format an.
- *--sv-call-regions-bed*: Gibt eine BED-Datei an, die den Satz mit Regionen für das Calling enthält. Die Datei muss bgzip-komprimiert und tabix-indiziert sein.
- *--sv-region*: Schränkt für das Debugging die Analyse auf eine festgelegte Region des Genoms ein. Diese Option kann wiederholt angegeben werden, um eine Liste mit Regionen zu erstellen. Der Wert muss das Format „chr:startPos-endPos" aufweisen.

- *--sv-exome*: Wenn dieser Wert auf „true“ festgelegt ist, wird der Varianten-Caller für gezielte Sequenzierungseingaben konfiguriert. Filter mit großer Tiefe werden deaktiviert. Die Standardeinstellung ist „false“.
- *--sv-output-contigs*: Bei Festlegung auf „true“ werden assemblierte Contig-Sequenzen in einer VCF-Datei ausgegeben. Die Standardeinstellung ist „false“.
- *--sv-quiet*: Bei Festlegung auf „true“ wird die Protokollausgabe auf stderr verhindert. (Es wird jedoch weiterhin eine entsprechende Protokolldatei erstellt.) Die Standardeinstellung ist „true“.

Betriebsmodi

Das Calling struktureller Varianten kann in den folgenden Modi erfolgen:

- Standalone (Eigenständig): Verwendet gemappte BAM-/CRAM-Eingabedateien. Für diesen Modus sind die folgenden Optionen erforderlich:
  - *--enable-map-align false*
  - *--enable-sv true*
- Integrated (Integriert): Wird automatisch mit der Ausgabe des DRAGEN-Mappers/-Aligners durchgeführt. Für diesen Modus sind die folgenden Optionen erforderlich:
  - *--enable-map-align true*
  - *--enable-sv true*
  - *--enable-map-align-output true*
  - *--output-format bam*

Das Calling struktureller Varianten kann auch zusammen mit einem beliebigen anderen Caller aktiviert werden.

Im Folgenden finden Sie eine Beispielbefehlszeile für den Integrated-Modus:

```
dragen -f \
   --sv-reference <FASTA mit .fai-Datei> \
   --ref-dir=<HASHTABELLE> \
   --enable-map-align true \
   --enable-map-align-output true \
   --output-format BAM \
   --enable-sv true \
   --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <PRÄFIX> \
   --RGID Illumina_RGID \
   --RGSM <Probenname> \
   -1 <FASTQ1> \
   -2 <FASTQ2>
```

Im Folgenden finden Sie eine Beispielbefehlszeile für das Joint Diploid-Calling im Standalone-Modus:

```
dragen -f \
   --sv-reference <FASTA mit .fai-Datei> \
   --ref-dir <HASHTABELLE> \
   --bam-input <BAM1> \
   --bam-input <BAM2> \
   --bam-input <BAM3> \
   --enable-map-align false \
```

```
--enable-sv true \
--output-directory <AUSGABEVERZEICHNIS> \
--output-file-prefix <PRÄFIX>
```

VCF-Ausgabe für strukturelle Varianten

Die VCF-Ausgabedatei für strukturelle Varianten steht im Ausgabeverzeichnis zur Verfügung. Die Datei ist mit *<Präfix der Ausgabedatei>.sv.vcf.gz* bezeichnet.

Der Structural Variant Caller generiert eine zusätzliche Ausgabe im Verzeichnis <Ausgabeverzeichnis>/sv/. Der Ordner <Ausgabeverzeichnis>/sv/results enthält zusätzliche Varianten- und Statistikausgabedateien. Weitere Unterverzeichnisse enthalten Protokolle und vorübergehende Ausgaben des Varianten-Callings.

Prognose struktureller Varianten

Unter <Ausgabeverzeichnis>/sv/results/variants gibt Manta eine Reihe von VCF 4.1-Dateien aus. Derzeit werden zwei VCF-Dateien für eine Keimbahn-Analyse sowie eine zusätzliche somatische VCF-Datei für eine Tumor-Normal-Subtraktion erstellt. Es handelt sich um folgende Dateien:

- diploidSV.vcf.gz

  SVs und Indels, für die Scores und Genotypen nach einem diploiden Modell für die Probenreihe in einer gemeinsamen Diploid-Probenanalyse oder für die Normalprobe in einer Tumor-Normal-Subtraktionsanalyse generiert wurden. Im Fall einer Tumor-Normal-Subtraktion enthalten die Scores in dieser Datei keine Informationen aus der Tumorprobe.

- somaticSV.vcf.gz

  SVs und Indels, für die Scores nach einem Modell für somatische Varianten generiert wurden. Diese Datei wird nur erstellt, wenn während der Konfiguration eine Alignment-Datei für eine Tumorprobe bereitgestellt wird.

- candidateSV.vcf.gz

  SV- und Indel-Kandidaten, für die keine Scores generiert wurden. Es ist nur ein minimaler Beleg erforderlich, damit eine SV als Kandidat in diese Datei aufgenommen wird. Eine SV oder ein Indel kann nur als Kandidat einen Score erhalten. Daher kann eine SV nur in anderen VCF-Ausgaben enthalten sein, wenn sie auch in dieser Datei vorhanden ist. Standardmäßig sind in dieser Datei Indels mit einer Größe ab 8 enthalten. Die kleinsten Indels in dieser Reihe werden an einen Caller für kleine Varianten ohne Scoring von Manta weitergegeben (der Manta-Standardscore beginnt bei einer Größe von 50).

Bei Tumor-Only-Analysen wird von Manta eine zusätzliche VCF-Ausgabedatei erstellt:

- tumorSV.vcf.gz

  Teilmenge der Datei candidateSV.vcf.gz nach dem Entfernen redundanter Kandidaten und kleiner Indels, die unter der Mindestgröße für Varianten-Scores liegen (der Standardwert ist 50). Für die SVs wurden keine Scores generiert, jedoch enthalten sie folgende zusätzliche Informationen: (1) Anzahl der Belege aus Paired- und Split-Reads für jedes Allel, (2) eine Untergruppe der Filter aus dem Tumor-Normal-Modell mit Scores, für höhere Präzision auf einen einzelnen Tumorfall angewendet.

Manta VCF-Ausgabe

Die Manta VCF-Ausgabe entspricht der Spezifikation VCF 4.1 für die Beschreibung struktureller Varianten. Für die Ausgabe werden, wenn möglich, Standardfeldnamen verwendet. Alle benutzerdefinierten Felder werden in der Kopfzeile der VCF-Datei beschrieben. In den folgenden Abschnitten finden Sie ausführliche Informationen zur Variantendarstellung sowie zu den primären VCF-Feldwerten.

VCF-Probennamen

Die Probennamen in der VCF-Ausgabe werden aus den einzelnen Alignierungs-Eingabedateien des ersten Read-Gruppendatensatzes (@RG) abgerufen, der in der Kopfzeile aufgeführt ist. Leerzeichen im Namen werden durch Unterstriche ersetzt. Wird kein Probenname gefunden, wird eine Standardkennzeichnung (SAMPLE1, SAMPLE2 usw.) verwendet.

Kleines Indel

Alle Varianten werden in der VCF-Datei mittels symbolischer Allele protokolliert, sofern sie nicht als kleines Indel klassifiziert sind. In diesem Fall werden für die Allel-Felder REF und ALT der VCF-Datei vollständige Sequenzen bereitgestellt. Eine Variante wird als kleines Indel klassifiziert, wenn alle der folgenden Bedingungen erfüllt sind:

- Die Variante kann vollständig als Kombination einer Insertions- und Deletionssequenz ausgedrückt werden.
- Die Länge der Deletion oder Insertion ist kleiner als 1.000.
- Die Variantenbruchenden und/oder die eingefügte Sequenz sind präzise.

Wenn VCF-Datensätze im Format für kleine Indels ausgegeben werden, beinhalten sie auch das CIGAR INFO-Tag mit einer Beschreibung des kombinierten Insertions- und Deletionsereignisses.

Insertionen mit unvollständiger Insertsequenz-Assemblierung

Große Insertionen werden in bestimmten Fällen gemeldet, selbst wenn die Insertsequenz nicht vollständig assembliert werden kann. In diesem Fall meldet Manta die Insertion mithilfe des symbolischen Allels <INS> und gibt in den betreffenden INFO-Feldern LEFT_SVINSSEQ und RIGHT_SVINSSEQ an, womit das assemblierte linke und rechte Ende der Insertsequenz beschrieben werden. Im Folgenden ist ein Beispiel für einen solchen Datensatz aus der Joint Diploid-Analyse von NA12878, NA12891 und NA12892 mit einem Mapping auf hg19 aufgeführt:

```
chr1 11830208 MantaINS:1577:0:0:0:3:0 T <INS> 999 PASS
END=11830208;SVTYPE=INS;CIPOS=0,12;CIEND=0,12;HOMLEN=12;HOMSEQ=TAAATTTT
TCTT;LEFT_
SVINSSEQ=TAAATTTTCTTTTTTCTTTTTTTTTTAAATTTATTTTTTTATTGATAATTCTTGGGTGTTT
CTCACAGAGGGGGATTTGGCAGGGTCACGGGACAACAGTGGAGGGAAGGTCAGCAGACAAACAAGTGAACA
AAGGTCTCTGGTTTTCCCAGGCAGAGGACCCTGCGGCCTTCCGCAGTGTTCGTGTCCCTGATTACCTGAGA
TTAGGGATTTGTGATGACTCCCAACGAGCATGCTGCCTTCAAGCATCTGTTCAACAAAGCACATCTTGCAC
TGCCCTTAATTCATTTAACCCCGAGTGGACACAGCACATGTTTCAAAGAG;RIGHT_
SVINSSEQ=GGGGCAGAGGCGCTCCCCACATCTCAGATGATGGGCGGCCAGGCAGAGACGCTCCTCACTTC
CTAGATGTGATGGCGGCTGGGAAGAGGCGCTCCTCACTTCCTAGATGGGACGGCGGCCGGGCGGAGACGCT
CCTCACTTTCCAGACTGGGCAGCCAGGCAGAGGGGCTCCTCACATCCCAGACGATGGGCGGCCAGGCAGAG
ACACTCCCCACTTCCCAGACGGGGTGGCGGCCGGGCAGAGGCTGCAATCTCGGCACTTTGGGAGGCCAAGG
CAGGCGGCTGCTCCTTGCCCTCGGGCCCCGCGGGGCCCGTCCGCTCCTCCAGCCGCTGCCTCC
GT:FT:GQ:PL:PR:SR 0/1:PASS:999:999,0,999:22,24:22,32
0/1:PASS:999:999,0,999:18,25:24,20 0/0:PASS:230:0,180,999:39,0:34,0
```

Inversionen

Inversionen werden standardmäßig als Bruchenden aufgeführt. Für eine einfache reziproke Inversion werden vier Bruchenden mit demselben EVENT INFO-Tag aufgeführt. Im Folgenden finden Sie ein Beispiel für eine reziproke Inversion:

```
chr1 17124941 MantaBND:1445:0:1:1:3:0:0 T [chr1:234919886[T 999 PASS
SVTYPE=BND;MATEID=MantaBND:1445:0:1:1:3:0:1;CIPOS=0,1;HOMLEN=1;
HOMSEQ=T;INV5;EVENT=MantaBND:1445:0:1:0:0:0:0;JUNCTION_QUAL=254;BND_
DEPTH=107;
MATE_BND_DEPTH=100 GT:FT:GQ:PL:PR:SR 0/1:PASS:999:999,0,999:65,8:15,51
chr1 17124948 MantaBND:1445:0:1:0:0:0:0 T T]chr1:234919824] 999 PASS
SVTYPE=BND;MATEID=MantaBND:1445:0:1:0:0:0:1;INV3;EVENT=MantaBND:1445:0:
1:0:0:0:0;
JUNCTION_QUAL=999;BND_DEPTH=109;MATE_BND_DEPTH=83 GT:FT:GQ:PL:PR:SR
0/1:PASS:999:999,0,999:60,2:0,46
chr1 234919824 MantaBND:1445:0:1:0:0:0:1 G G]chr1:17124948] 999 PASS
SVTYPE=BND;MATEID=MantaBND:1445:0:1:0:0:0:0;INV3;EVENT=MantaBND:1445:0:
1:0:0:0:0;
JUNCTION_QUAL=999;BND_DEPTH=83;MATE_BND_DEPTH=109 GT:FT:GQ:PL:PR:SR
0/1:PASS:999:999,0,999:60,2:0,46
chr1 234919885 MantaBND:1445:0:1:1:3:0:1 A [chr1:17124942[A 999 PASS
SVTYPE=BND;MATEID=MantaBND:1445:0:1:1:3:0:0;CIPOS=0,1;HOMLEN=1;
HOMSEQ=A;INV5;EVENT=MantaBND:1445:0:1:0:0:0:0;JUNCTION_QUAL=254;BND_
DEPTH=100;
MATE_BND_DEPTH=107 GT:FT:GQ:PL:PR:SR 0/1:PASS:999:999,0,999:65,8:15,51
```

VCF INFO-Felder

ID	Beschreibung
IMPRECISE	Kennzeichnung, die auf eine nicht präzise strukturelle Variante hinweist, d. h., der genaue Unterbrechungspunkt wurde nicht gefunden
SVTYPE	Typ der strukturellen Variante
SVLEN	Längendifferenz zwischen REF- und ALT-Allelen
END	Endposition der in diesem Datensatz beschriebenen Variante
CIPOS	Konfidenzintervall um POS
CIEND	Konfidenzintervall um END
CIGAR	CIGAR-Alignment für jedes zweite Indel-Allel
MATEID	ID des Mate-Bruchendes
EVENT	ID des zum Bruchende gehörigen Ereignisses
HOMLEN	Länge der identischen Basenpaar-Homologie an Ereignisunterbrechungspunkten
HOMSEQ	Sequenz der identischen Basenpaar-Homologie an Ereignisunterbrechungspunkten
SVINSLEN	Länge der Insertion
SVINSSEQ	Sequenz der Insertion
LEFT_SVINSSEQ	Bekanntes linkes Ende einer Insertion bei einer Insertion unbekannter Länge
RIGHT_SVINSSEQ	Bekanntes rechtes Ende einer Insertion bei einer Insertion unbekannter Länge

ID	Beschreibung
BND_DEPTH	Read-Tiefe am lokalen Bruchende der Translokation
MATE_BND_DEPTH	Read-Tiefe am remoten Mate-Bruchende der Translokation
JUNCTION_QUAL	Wenn die SV-Stelle zu einem EVENT gehört (d. h. eine Variante mit mehreren Adjazenzen), enthält dieses Feld den QUAL-Wert ausschließlich für die betreffende Adjazenz
SOMATIC	Kennzeichnung für eine somatische Variante
SOMATICSCORE	Qualitäts-Score der somatischen Variante
JUNCTION_ SOMATICSCORE	Wenn die SV-Stelle zu einem EVENT gehört (d. h. eine Variante mit mehreren Adjazenzen), enthält dieses Feld den SOMATICSCORE-Wert ausschließlich für die betreffende Adjazenz
CONTIG	Assemblierte Contig-Sequenz, wenn die Variante nicht unpräzise ist (mit *--outputContig*)

VCF FORMAT-Felder

ID	Beschreibung
GT	Genotyp
FT	Probenfilter, „PASS“ zeigt an, dass bei dieser Probe alle Filter passiert wurden
GQ	Genotypqualität
PL	Normalisierte, Phred-skalierte Wahrscheinlichkeiten für Genotypen gemäß der Definition in der VCF-Spezifikation
PR	Anzahl der umspannenden Read-Paare, die die REF- oder ALT-Allele stark (Q30) unterstützen
SR	Anzahl der Split-Reads, die die REF- oder ALT-Allele stark (Q30) unterstützen

VCF FILTER-Felder

ID	Stufe	Beschreibung
MinQUAL	Datensatz	QUAL-Score unter 20
MinGQ	Probe	GQ-Score unter 15
MinSomaticScore	Datensatz	SOMATICSCORE unter 30
Ploidy	Datensatz	Für DEL- und DUP-Varianten: Genotypen überlappender Varianten (mit vergleichbarer Größe) stimmen nicht mit der Diploid-Prognose überein
MaxDepth	Datensatz	Tiefe in der Nähe eines oder beider Variantenbruchenden ist höher als die dreifache mittlere Chromosomentiefe
MaxMQ0Frac	Datensatz	Für eine kleine Variante (< 1.000 Basen): Anteil von Reads in allen Proben mit MAPQ0 an einem der Bruchenden überschreitet 0,4
NoPairSupport	Datensatz	Für Varianten, die die Paired-Read-Fragmentgröße deutlich überschreiten: keiner der Paired-Reads unterstützt das alternative Allel in einer der Proben
SampleFT	Datensatz	Keine Probe durchläuft die Filter auf Probenebene
HomRef	Probe	Homozygoter Referenz-Call

Interpretation von VCF-Dateien

Es gibt zwei Filterebenen: die Datensatzebene (FILTER) und die Probenebene (FORMAT/FT). Prinzipiell sind Filter auf Datensatzebene unabhängig von Filtern auf Probenebene. Sollte jedoch keine der Proben alle Filter auf Probenebene passieren, wird der SampleFT-Filter auf Datensatzebene angewendet.

Interpretation des Felds INFO/EVENT

Einige in der VCF-Datei aufgeführte strukturelle Varianten wie beispielsweise Translokationen werden als einzelne neue Sequenzverknüpfung in der Probe behandelt. Manta gibt im Feld INFO/EVENT an, dass bei mindestens zwei solcher Verknüpfungen davon ausgegangen wird, dass diese als Teil einer einzelnen Variantenereignisses gemeinsam auftreten. Alle einzelnen Variantendatensätze, die zum gleichen Ereignis gehören, weisen die gleiche INFO/EVENT-Zeichenfolge auf. Obwohl eine solche Schlussfolgerung auch nach dem SV-Calling durch eine Analyse der relativen Entfernung und Ausrichtung der jeweiligen Varianten-Unterbrechungspunkte möglich ist, nimmt Manta diesen Ereignismechanismus in den Calling-Prozess auf, um die Sensitivität für solche größeren Ereignisse zu erhöhen. Vorausgesetzt, dass mindestens eine Ereignisverknüpfung bereits die Schwellenwerte für Standardvariantenkandidaten überschritten hat, wird die Sensitivität durch das Herabsetzen der Evidenzschwellenwerte für zusätzliche Verknüpfungen in einem Muster, das mit einem Ereignis mit mehreren Verknüpfungen (z. B. ein reziprokes Translokationspaar) konsistent ist, verbessert.

Obwohl dieser Mechanismus allgemein für Ereignisse angewendet werden kann, die eine beliebige Anzahl an Verknüpfungen aufweisen, ist er derzeit auf zwei beschränkt. Derzeit ist der Mechanismus insbesondere zur Identifizierung und Verbesserung der Sensitivität für reziproke Translokationspaare hilfreich.

Feld VCF ID

Das Feld VCF ID (Bezeichner) kann für Annotationen verwendet werden. Bei BND(Bruchenden)-Datensätzen für Translokationen werden mit dem ID-Wert Bruchenden-Mates oder -Partner verknüpft. Im Folgenden finden Sie ein Beispiel einer Manta VCF-ID.

```
MantaINS:1577:0:0:0:3:0
```

Der im Feld ID bereitgestellte Wert bezeichnet den Rand/die Ränder des SV-Assoziationsdiagramms, in dem die SV oder das Indel entdeckt wurde. Der von Manta bereitgestellte ID-Wert ist vor allem für die interne Nutzung durch Manta-Entwickler vorgesehen. Der Wert ist innerhalb jeder von Manta erstellten VCF-Datei eindeutig. Diese ID-Werte können für die Verknüpfung zugehöriger Datensätze mithilfe des VCF MATEID-Standardschlüssels verwendet werden. Möglicherweise wird die Struktur dieser ID in Zukunft geändert. Der gesamte Wert kann problemlos als eindeutiger Schlüssel verwendet werden. Eine Analyse dieses Werts kann jedoch zu Kompatibilitätsproblemen mit zukünftigen Updates führen.

Konvertieren von Manta-VCF-Dateien in das BEDPE-Format

Manchmal ist es praktisch, für strukturelle Varianten das BEDPE-Format zu nutzen. Für solche Anwendungen empfehlen wir das Skript vcfToBedpe, das unter folgendem Link verfügbar ist:

```
https://github.com/ctsa/svtools
```

Dieses Repository geht auf @hall-lab zurück und unterstützt dank Modifikationen das VCF 4.1 SV-Format sowie die Portabilitätseinschränkungen von Manta.

Im BEDPE-Format sind im Vergleich zur VCF-Ausgabe von Manta deutlich weniger Informationen zu strukturellen Varianten vorhanden. Vor allem Ausrichtung und Homologie der Bruchenden sowie die Insertionssequenz fehlen. Außerdem können Felder nicht für spezifische Locus- und Probeninformationen definiert werden. Aus diesem Grund empfiehlt Illumina, BEDPE nur als vorübergehendes Ausgabeformat für Anwendungen zu verwenden, die auf dieses Format angewiesen sind.

Statistik-Ausgabedatei

Weitere sekundäre Ausgabewerte werden in den Dateien unter <Ausgabeverzeichnis>/sv/results/stats bereitgestellt.

- alignmentStatsSummary.txt

  Fragmentlängenquantile für jede Eingabe-Alignment-Datei.

- svLocusGraphStats.tsv

  Statistiken und Laufzeitinformationen in Bezug auf das SV-Locus-Diagramm.

- svCandidateGenerationStats.tsv

  Statistiken und Laufzeitinformationen in Bezug auf die SV-Kandidatengenerierung.

- svCandidateGenerationStats.xml

  XML-Daten zur Unterstützung des Berichts svCandidateGenerationStats.tsv.

- diploidSV.sv_metrics.csv

  Die Anzahl der erfolgreichen SV-Calls im Rahmen des diploiden Modells. Diese Datei wird nur bei der Keimbahn-Analyse oder der Tumor-Normal-Analyse generiert.

- somaticSV.sv_metrics.csv

  Die Anzahl der erfolgreichen SV-Calls im Rahmen des Modells für somatische Varianten. Diese Datei wird nur bei der Tumor-Normal-Analyse generiert.

- tumorSV.sv_metrics.csv

  Die Anzahl der erfolgreichen SV-Calls bei der Tumor-Only-Analyse. Diese Datei wird nur bei der Tumor-Only-Analyse generiert.

De-novo-Qualitäts-Scoring struktureller Varianten

Das De-novo-Qualitäts-Scoring lässt sich für das Joint Diploid-Calling struktureller Varianten aktivieren, indem *--sv-denovo-scoring* auf „true“ festgelegt und eine Stammbaumdatei bereitgestellt wird. Dies fügt der VCF-Ausgabedatei die Felder FORMAT/DQ und FORMAT/DN hinzu, die einen De-novo-Qualitäts-Score und einen zugehörigen De-novo-Call aufnehmen können.

Das folgende Beispiel zeigt eine Befehlszeile zur Aktivierung des De-novo-Qualitäts-Scorings für einen Joint Diploid-Lauf.

```
dragen -f
   --sv-reference <FASTA> \
   --ref-dir <HASHTABELLE> \
   --bam-input <BAM1> \
   --bam-input <BAM2> \
   --bam-input <BAM3> \
   --enable-map align=false \
   --enable-sv=true \
   --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <PRÄFIX> \
   --sv-denovo-scoring true \
   --RGID DRAGEN_RGID \
   --RGSM <Probenname>
   --pedigree-file <PED-DATEI>
```

Dragen kann auch mit einer vorhandenen Ausgabe-VCF mit strukturellen Varianten ausgeführt werden, die mehrere Proben (z. B. ein Trio mit Proband und Eltern) enthält, um eine modifizierte VCF-Datei mit den Feldern FORMAT/DQ und FORMAT/DN zu generieren. (Die Originaldatei wird nicht geändert.)

Im folgenden Beispiel finden Sie eine Befehlszeile zur Ableitung eines De-novo-Qualitäts-Scores aus einem vorhandenen SV-Trio.

```
dragen -f \
   --variant <TRIO_VCF-DATEI> \
   --pedigree-file <PED-DATEI> \
   --enable-map-align false \
   --sv-denovo-scoring true \
   --output-directory <AUSGABEVERZEICHNIS> \
   --output-file-prefix <PRÄFIX>
```

Das Feld DQ ist folgendermaßen definiert:

```
##FORMAT=<ID=DQ,Number=1,Type=Float,Description="Denovo quality">
```

Das Feld DQ enthält einen Score für die A-posteriori-Wahrscheinlichkeit, dass es sich bei der Probandenvariante um eine De-novo-Variante handelt. Wenn sich dieser berechnen lässt, wird der Score in der Phred-Skala zum Probanden hinzugefügt und die übrigen Proben werden mit einem Punkt (.) gekennzeichnet, der angibt, dass dieser Score fehlt.

Beispielsweise entsprechen DQ-Scores von 13 und 20 einer A-posteriori-Wahrscheinlichkeit für eine De-novo-Variante von 0,95 bzw. 0,99.

Das Feld DN ist folgendermaßen definiert:

```
##FORMAT=<ID=DN,Number=1,Type=String,Description="Possible values are
   'DeNovo' or 'LowDQ'. Threshold for a passing de novo call is DQ >=
   20">
```

DRAGEN vergleicht gültige (> 0) DQ-Scores mit einem Schwellenwert mit einem Standard-Score von 20. Bei einem Score größer oder gleich dem Schwellenwert erhält das Feld DN den Wert „DeNovo“. Bei einem Score unterhalb des Schwellenwerts wird der Wert „LowDQ“ zugewiesen. Wenn kein gültiger DQ-Score vorhanden ist (z. B. DQ = „0“ oder „.“), wird das Feld DN auf „.“ festgelegt.

Der Schwellenwert kann mit der Befehlszeilenoption *--sv-denovo-threshold* geändert werden. Fügen Sie beispielsweise *--sv-denovo-threshold 10* zur Befehlszeile von DRAGEN hinzu, wenn der Schwellenwert auf 10 begrenzt werden soll.

Die Eingaben für diese Funktion sind die VCF-Datei und die Stammbaumdatei, die angibt, welche Probe des Trios zum Probanden, zur Mutter bzw. zum Vater gehört. Für den Fall, dass in der Stammbaumdatei mehrere Trios angegeben sind (z. B. ein Stammbaum mit mehreren Generationen), erkennt DRAGEN die Trios automatisch und bestimmt die De-novo-Varianten in der Probandenprobe für jedes Trio.

Qualitätssicherungsmetriken und Berichte zur Coverage/Callfähigkeit

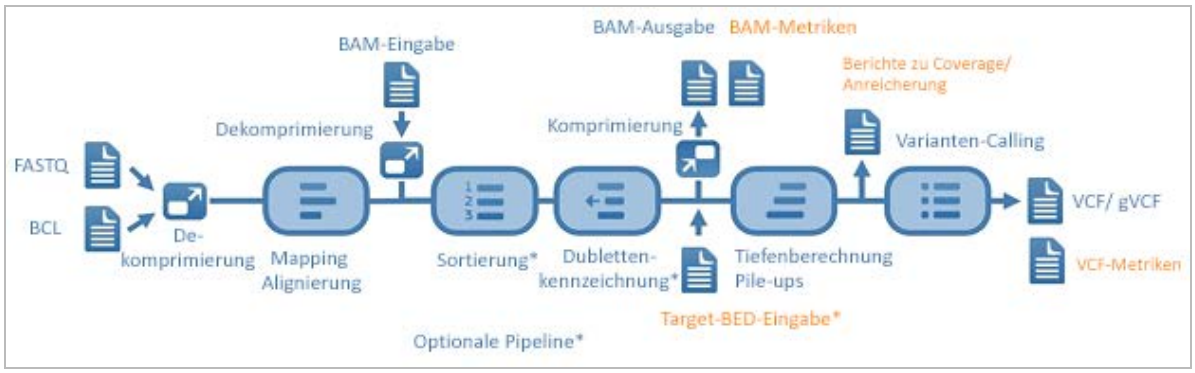
Bei jedem Lauf werden Coverage-Berichte mit pipelinespezifischen Metriken generiert. In den verschiedenen Phasen der Pipeline werden vier unterschiedliche Gruppen von Metriken generiert:

- Mapping- und Alignment-Metriken
- VCF-Metriken
- Metriken für die Dauer (oder Laufzeit)
- Metriken und Berichte für die Coverage (oder Anreicherung)

Mapping-Alignment-Metriken, VCF-Metriken, Metriken für die Dauer und eine Untergruppe an verfügbaren Coverage-Berichten werden automatisch generiert und erfordern keine Aktivierung oder spezifischen Befehle. Zusätzliche Coverage-Metriken können aktiviert und zusätzliche Coverage-Regionen spezifiziert werden.

Die Berechnung der Metriken wird während der Analyse durchgeführt, sodass die Laufzeit von DRAGEN nicht beeinträchtigt wird.

Abbildung 10 Generierung von Metriken und Berichten



Ausgabeformat für Qualitätssicherungsmetriken

Die Qualitätssicherungsmetriken werden in einem lesbaren Format über den Standardausgang ausgegeben, CSV-Dateien werden im Laufausgangsverzeichnis gespeichert.

- <Ausgabepräfix>.mapping_metrics.csv
- <Ausgabepräfix>.vc_metrics.csv
- <Ausgabepräfix>.time_metrics.csv
- <Ausgabepräfix>.<Coverage-Regionspräfix>_coverage_metrics.csv
- <Ausgabepräfix>.<sonstige Coverage Berichte>.csv

Die einzelnen Zeilen sind selbsterklärend und damit leicht lesbar.

Abschnitt	RG/Probe	**Metrik**	Anzahl/Verhältnis/Zeit	Prozentsatz/Sekunden
MAPPING/ALIGNING SUMMARY		Eingabe-Reads insgesamt	816360354	
MAPPING/ALIGNING SUMMARY		Anzahl der doppelten Reads (markiert, nicht entfernt)	15779031	1.93
…				
MAPPING/ALIGNING PER RG	RGID_1	Reads in RG insgesamt	816360354	100
MAPPING/ALIGNING PER RG	RGID_1	Anzahl der doppelten Reads (markiert)	15779031	1.93
…				
VARIANT CALLER SUMMARY		Anzahl der Proben	1	
VARIANT CALLER SUMMARY		Verarbeitete Reads	738031938	

…

Abschnitt	RG/Probe	Metrik	Anzahl/Verhältnis/Zeit	Prozentsatz/Sekunden
VARIANT CALLER PREFILTER	SAMPLE_1	Gesamt	4918287	100
VARIANT CALLER PREFILTER	SAMPLE_1	Biallelisch	4856654	98.75
...				
RUN TIME		Referenzladedauer	00:18.6	18.65
RUN TIME		Dauer der Read-Alignierung	19:24.4	1164.42

Mapping- und Alignment-Metriken

Mapping- und Alignment-Metriken stehen wie die über den Befehl samtools flagstat berechneten Metriken auf einer aggregierten Ebene (alle Eingabedaten) und auf einer Read-Gruppenebene zur Verfügung. Sofern nicht ausdrücklich angegeben, beziehen sich die Einheiten der Metriken auf Reads (und nicht auf Paare oder Alignments).

- Eingabe-Reads insgesamt: Gesamtzahl der Reads in den FASTQ-Eingabedateien.
- Anzahl der als Dublette gekennzeichneten Reads: Anzahl der aufgrund der Verwendung der Option *--enable-duplicate-marking* als Dublette gekennzeichneten Reads.
- Anzahl der entfernten doppelten und Mate-Reads: Anzahl der als Dublette gekennzeichneten Reads und der Mate-Reads, die bei Verwendung der Option *--remove-duplicates* entfernt werden.
- Anzahl der eindeutigen Reads: Gesamtzahl der Reads abzüglich der als Dublette gekennzeichneten Reads.
- Reads mit sequenziertem Mate: Anzahl der Reads mit einem Mate.
- Reads ohne sequenzierten Mate: Gesamtzahl der Reads abzüglich der Anzahl der Reads mit einem sequenzierten Mate.
- Reads mit einem Qualitätssicherungsfehler: Anzahl der Reads, die Qualitätsprüfungen der Plattform/des Anbieters nicht bestehen (SAM-Markierung 0x200).
- Gemappte Reads: Gesamtzahl der Reads abzüglich der Anzahl der nicht gemappten Reads.
- Anzahl der eindeutigen und gemappten Reads: Anzahl der gemappten Reads abzüglich der als Dublette gekennzeichneten Reads.
- Nicht gemappte Reads: Gesamtzahl der Reads, die nicht gemappt werden konnten.
- Singleton-Reads: Anzahl der Reads, bei denen der Read gemappt, der Paired-Mate jedoch nicht gelesen werden konnte.
- Paired-Reads: Anzahl der Reads, in denen beide Reads im Paar gemappt sind.
- Korrekte Paired-Reads: Beide Reads im Paar sind gemappt und liegen basierend auf der geschätzten Insertionslängenverteilung innerhalb eines akzeptablen Abstands zueinander.
- Nicht korrekte Paired-Reads (diskordant): Anzahl der Paired-Reads abzüglich der Anzahl der korrekten Paired-Reads.
- Auf verschiedene Chromosome gemappte Paired-Reads: Anzahl der Reads mit einem Mate, wobei der Mate auf ein anderes Chromosom gemappt wurde.
- Auf verschiedene Chromosome gemappte Paired-Reads (MAPQ >= 10): Anzahl der Reads mit einem MAPQ >= 10 und einem Mate, wobei der Mate auf ein anderes Chromosom gemappt wurde.
- R1-Reads mit Indel: Prozentsatz der R1-Reads mit mindestens einem Indel.

- **R2-Reads mit Indel**: Prozentsatz der R2-Reads mit mindestens einem Indel.
- **R1-Basen mit Soft-Clipping**: Prozentsatz der Basen in R1-Reads mit Soft Clipping.
- **R2-Basen mit Soft-Clipping**: Prozentsatz der Basen in R2-Reads mit Soft Clipping.
- **Nicht übereinstimmende R1-Basen**: Anzahl der nicht übereinstimmenden Basen in R1, d. h. die Summe aus der SNP-Anzahl und den Indel-Längen. Alles innerhalb von Soft Clippings oder RNA-Introns wird nicht berücksichtigt. Auch bei einer Referenzbase oder Read-Base gleich N wird eine Nichtübereinstimmung nicht gezählt.
- **Nicht übereinstimmende R2-Basen**: Anzahl der nicht übereinstimmenden Basen in R2, d. h. die Summe aus der SNP-Anzahl und den Indel-Längen. Alles innerhalb von Soft Clippings oder RNA-Introns wird nicht berücksichtigt. Auch bei einer Referenzbase oder Read-Base gleich N wird eine Nichtübereinstimmung nicht gezählt.
- **Nicht übereinstimmende R1-Basen (ohne Indels)**: Anzahl der nicht übereinstimmenden Basen in R1. Die Indel-Längen werden ignoriert. Alles innerhalb von Soft Clippings oder RNA-Introns wird nicht berücksichtigt. Auch bei einer Referenzbase oder Read-Base gleich N wird eine Nichtübereinstimmung nicht gezählt.
- **Nicht übereinstimmende R2-Basen (ohne Indels)**: Anzahl der nicht übereinstimmenden Basen in R2. Die Indel-Längen werden ignoriert. Alles innerhalb von Soft Clippings oder RNA-Introns wird nicht berücksichtigt. Auch bei einer Referenzbase oder Read-Base gleich N wird eine Nichtübereinstimmung nicht gezählt.
- **Q30-Basen**: Gesamtzahl der Basen mit einem BQ >= 30.
- **Q30-Basen R1**: Gesamtzahl der Basen in R1 mit einem BQ >= 30.
- **Q30-Basen R2**: Gesamtzahl der Basen in R2 mit einem BQ >= 30.
- **Q30-Basen (ohne Dubletten und geclippte Basen)**: Anzahl der Basen ohne Dublette und nicht geclippten Basen mit einem BQ >= 30.
- **Histogramm der Mapping-Qualität von Reads**
  - Reads mit MAPQ [40:inf)
  - Reads mit MAPQ [30:40)
  - Reads mit MAPQ [20:30)
  - Reads mit MAPQ [10:20)
  - Reads mit MAPQ [0:10)
- **Alignments insgesamt**: Gesamtzahl der Loci-Reads mit einem Alignment-Qualitätswert > 0.
- **Sekundäre Alignments**: Anzahl der sekundären Alignment-Loci.
- **Zusätzliche (chimärische) Alignments**: Ein chimärischer Read ist auf mehrere Loci aufgeteilt (möglicherweise aufgrund struktureller Varianten). Ein Alignment wird als repräsentatives, das andere als zusätzliches Alignment bezeichnet.
- **Geschätzte Read-Länge**: Gesamtzahl der Eingabebasen geteilt durch die Anzahl der Reads.
- **Histogramm**: Weitere Informationen finden Sie unter *Histogram Coverage-Bericht* auf Seite 96.
- **Prozentsatz alignierter Basen innerhalb der Intervallregion**: Anzahl der Basen innerhalb der Intervall- und der Zielregion geteilt durch die Gesamtzahl der alignierten Basen.
- **Geschätzte Probenkontamination**: Wenn das Varianten-Calling mit einem beliebigen Bioinformatik-Tool durchgeführt wird, kann die Genauigkeit durch eine Kreuzkontamination der Proben stark beeinträchtigt sein. Selbst kleine Kontaminationsgrade können viele falsch positive Calls zur Folge haben, insbesondere in Pipelines, in denen Varianten mit niedriger Allelfrequenz nachgewiesen werden sollen.

  Das DRAGEN Cross-Sample Contamination-Modul schätzt anhand eines Mischverteilungsmodells den

Anteil der Reads in einer Probe, die möglicherweise von anderem menschlichen Gewebe stammen. Dieser Probenkontaminationsanteil wird geschätzt als der Parameterwert im Mischverteilungsmodell, der die Wahrscheinlichkeit des Vorliegens der beobachteten Reads an mehreren Pile-up-Positionen maximiert. Das Mischverteilungsmodell berücksichtigt die Populationsallelfrequenzen und die bestimmten Proben-Genotypen.

Um diese Metrik zu aktivieren, müssen Sie in der Befehlszeile den Dateipfad zu einer VCF-Datei angeben, die Markerbereiche (RSIDs) mit Populationsallelfrequenzen enthält. Beispiel:

```
--qc-cross-cont-vcf /opt/edico/config/sample_cross_contamination_
   resource_hg19.vcf
```

Die in DRAGEN enthaltenen VCF-Ressourcendateien können aus der Ensembl-Datenbank rekonstruiert werden. Die im Config-Ordner von DRAGEN enthaltenen VCF-Dateien enthalten ca. 5.000 Markerpositionen mit Populationsallelfrequenzen im Bereich von 0,5. Die Dateien beziehen sich jeweils auf eine bestimmte Referenz (hg19/GRCh37/hg38). DRAGEN bricht ab, wenn eine nicht kompatible Ressourcen- und Referenzdatei verwendet wird (z. B. CRCh37-Ressourcendatei und hg19-Referenz).

Im Folgenden ist eine Beispielausgabe für eine Probe ohne Kontamination dargestellt. Der Wert ist als Anteil angegeben. Ein Wert von 0.011 entspricht demnach 1,1 % geschätzter Kontamination.

```
MAPPING/ALIGNING SUMMARY Estimated sample contamination 0.000
```

Mapping- und Alignment-Metriken im somatischen Modus

Im somatischen Modus werden die Mapping- und Alignment-Metriken für Tumor- und normale Proben separat generiert. Die einzelnen Zeilen beginnen jeweils mit TUMOR oder NORMAL, um die Probe anzugeben. Die Metriken für die Tumorprobe werden vor den Metriken für die normale Probe ausgegeben. Außerdem werden die Metriken pro Read-Gruppe in Tumor- und Normalgruppen aufgeteilt.

Die Metriken für alignierte Reads (durchschnittliche Alignment-Coverage über die Genom-/Zielregion, durchschnittliche Chromosom-X/Y-Coverage über die Genom-/Zielregion, Anteil alignierter Basen innerhalb der Intervallregion usw.) werden für Tumor- und Normalgruppen NICHT separat bestimmt. Stattdessen wird der addierte Wert beider Proben angegeben. Diese Zeilen beginnen nicht mit TUMOR oder NORMAL und werden nicht pro Read-Gruppe angegeben.

Metriken für das Varianten-Calling

Die generierten Metriken für das Varianten-Calling ähneln den von RTG vcfstats berechneten Metriken. In VCF- und gVCF-Dateien mit mehreren Proben werden für jede Probe Metriken erfasst. Abhängig vom jeweiligen Lauf werden die Metriken standardmäßig entweder als VARIANT CALLER oder JOINT CALLER erfasst. Die Metriken werden sowohl für die rohen (PREFILTER) als auch die hart gefilterten (POSTFILTER) VCF-Dateien erfasst.

PON(Panel of Normals, Normalgruppe)- und COSMIC-gefilterte Varianten werden in den Metriken der POSTFILTER-VCF-Dateien als PASS-Varianten behandelt. Dadurch kann die Variantenanzahl in den Metriken der POSTFILTER-VCF-Dateien über der erwarteten Anzahl liegen.

Number of samples: Anzahl der Proben in der Populations-/gemeinsamen VCF-Datei.

Reads Processed: Anzahl der Reads, die für das Varianten-Calling verwendet werden, mit Ausnahme der als Duplikat markierten Reads und Reads, die außerhalb der Zielregion liegen.

Total: die Gesamtzahl der Varianten (SNPs + MNPs + INDELS).

Biallelic: Anzahl der Positionen in einem Genom mit zwei beobachteten Allelen. Die Referenz wird als ein Allel gezählt, sodass ein Varianten-Allel möglich ist.

Multiallelic: Anzahl der Positionen in der VCF-Datei mit mindestens drei beobachteten Allelen. Die Referenz wird als eins gezählt, sodass zwei oder mehr Varianten-Allele möglich sind.

SNPs: Eine Variante wird als SNP gezählt, wenn die Referenz, Allel 1 und Allel 2 über eine Länge von 1 verfügen.

Insertions (Hom): Anzahl der Varianten mit homozygoten Insertionen.

Insertions (Het): Anzahl der Varianten, bei denen es sich bei beiden Allelen um nicht homozygote Insertionen handelt.

Deletions (Het): Anzahl der Varianten mit homozygoten Deletionen.

Indels (Het): Anzahl der Varianten, bei denen die Genotypen vom Typ [insertion+deletion], [insertion+snp] oder [deletion+snp] sind.

DeNovo SNPs: SNPs mit De-novo-Markierung, bei denen DQ > 0.05 ist. Legen Sie die Option *--qc-snp-denovo-quality-threshold* gemäß dem erforderlichen Schwellenwert fest. Der Standardwert ist 0.05.

DeNovo INDELs: Indels mit De-novo-Markierung, bei denen DQ > 0.02 ist. Dieser DQ-Schwellenwert kann mithilfe der Option *--qc-indel-denovo-quality-threshold* auf den erforderlichen Wert festgelegt werden. Der Standardwert ist 0.02.

DeNovo MNPs: entspricht der Option für SNPs. Legen Sie die Option *--qc-snp-denovo-quality-threshold* gemäß dem erforderlichen Schwellenwert fest. Der Standardwert ist 0.05.

(Chr X SNPs)/(Chr Y SNPs) ratio in the genome (or the target region): Anzahl der SNPs im X-Chromosom (oder im Schnittpunkt des X-Chromosoms und der Zielregion) geteilt durch die Anzahl der SNPs im Y-Chromosom (oder im Schnittpunkt des Y-Chromosoms und der Zielregion). Liegt keine Alignierung an entweder das X- oder das Y-Chromosom vor, wird diese Metrik als „NA“ angezeigt.

SNP Transitions: ein Austausch zweier Purine (A<->G) oder zweier Pyrimidine (C<->T).

SNP Transversions: ein Austausch von Purin- und Pyrimidinbasen. Ti/Tv ratio: Verhältnis der Übergänge.

Heterozygous: Anzahl der heterozygoten Varianten.

Homozygous: Anzahl der homozygoten Varianten.

Het/Hom ratio: Verhältnis heterozygot/homozygot.

In dbSNP: Anzahl der erkannten Varianten, die in der dbsnp-Referenzdatei vorhanden sind. Wenn keine dbsnp-Datei über die Option *--bsnp* bereitgestellt wird, werden die Metriken In dbSNP und Novel als „NA“ angezeigt.

Novel: Gesamtzahl der Varianten, abzüglich der Anzahl der Varianten in dbSNP.

Percent Callability: nur im Keimbahn-Modus mit gVCF-Ausgabe verfügbar. Der Prozentsatz an nicht-N-Referenzpositionen mit Genotyp-Call mit dem Wert PASS. Varianten mit mehreren Allelen werden nicht gezählt. Deletionen werden nur bei homozygoten Calls für alle gelöschten Referenzpositionen gezählt. Es werden nur Autosome sowie X-, Y- und M-Chromosome berücksichtigt.

Percent Autosome Callability: Es werden nur Autosome berücksichtigt.

Bericht zur Callfähigkeit

DRAGEN generiert automatisch einen Bericht zur Callfähigkeit in den Varianten-Caller-Metriken, wenn der Keimbahn-Caller für kleine Varianten mit dem Ausgabemodus gVCF ausgeführt wird. Die Callfähigkeit ist definiert als der Anteil an Nicht-N-Referenzpositionen mit dem Wert PASS für den Genotyp-Call. Metriken zur Callfähigkeit werden anhand der folgenden Kriterien berechnet:

- Callfähigkeit wird anhand der gVCF berechnet.
- Decoy-Contigs werden ignoriert.

- Unplatzierte und unlokalisierte Contigs werden ignoriert.
- N-Positionen werden als nicht callfähig behandelt.
- Regionen, für die kein Varianten-Calling erfolgt ist, erhalten eine Callfähigkeit von 0.
- Eine homozygote Deletion gilt für alle Referenzpositionen innerhalb der Deletion als Genotyp-Call mit dem Wert PASS.

Wenn die Option *--vc-target-bed* angegeben wird, erfolgt die Ausgabe als Datei mit der Bezeichnung target_bed_callability.bed, die die allgemeine und autosome Callfähigkeit für die eingegebene Target-BED-Region enthält. Die mit der Option *--vc-target-bed-padding* angegebene Padding-Größe wird verwendet und überlappende Regionen werden zusammengefasst.

Die Callfähigkeit kann auch in den benutzerdefinierten Berichten zur Coverage/Callfähigkeit einer Region ausgegeben werden.

Metriken für die Dauer

Der Abschnitt mit den Metriken für die Dauer enthält für jeden Prozess eine Aufschlüsselung der Laufdauer. Für die Mapper- und Varianten-Caller-Pipeline werden beispielsweise folgende Metriken erstellt:

- Referenzladedauer
- Dauer der Read-Alignierung
- Dauer der Duplikatsortierung und -kennzeichnung
- Dauer der DRAGStr-Kalibrierung
- Dauer der teilweisen Neukonfigurierung
- Dauer des Varianten-Callings
- Gesamtdauer

Berichte zu Coverage/Callfähigkeit für anwendungsspezifische Bereiche

DRAGEN generiert die folgenden Coverage-Berichte:

- Einen Satz mit Standardberichten für das Gesamtgenom oder, bei Verwendung der Option *--vc-target-bed*, für die Zielregion.
- Wahlweise können zusätzliche Berichte für bis zu drei Regionen von Interesse (Coverage-Regionen) erstellt werden.

  Für jede angegebene Region generiert DRAGEN die Standardberichte sowie alle für die Region angeforderten zusätzlichen Berichte.

Verwenden Sie zum Generieren von regionsspezifischen Coverage-Berichten die Option *–qc-coverage-region-i*, wobei „i“ den Wert 1, 2 oder 3 erhält.

- Für jede *–qc-coverage-region-i*-Option muss ein BED-Dateiargument angegeben werden.
- Für die Regionen in den einzelnen BED-Dateien kann mit der Option *--qc-coverage-region-padding-i* ein Padding-Wert angegeben werden. Der Standardwert ist 0.
- Für jede Region wird ein Satz mit Standardberichten erstellt.
- Wahlweise können mit der Option *–qc-coverage-reports-i* zusätzliche Berichte für die einzelnen Regionen angeben werden.

Im folgenden Beispiel werden die für die Generierung von Coverage-Berichten erforderlichen Optionen dargestellt.

```
$ dragen … \
   --qc-coverage-region-1 <BED-Datei 1> \
   --qc-coverage-reports-1 full_res \
   --qc-coverage-region-2 <BED-Datei 2> \
   --qc-coverage-region-3 <BED-Datei 3> \
   --qc-coverage-reports-3 full_res cov_report
```

Zählen von Reads und Basen

Alle unten in Tabelle 7 und Tabelle 8 aufgeführten Standard- und optionalen Coverage-Berichte verwenden die folgenden Standardregeln zur Zählung von Reads und Basen:

- Doppelte Reads werden ignoriert.
- Basen mit Soft und/oder Hard Clipping werden ignoriert.
- Reads mit MAPQ=0 werden ignoriert.
- Überlappende Mates werden doppelt gezählt.

Nicht standardmäßige Einstellungen:

Berichte können mit oder ohne Ausführung von Mapper und Aligner oder Varianten-Caller erstellt werden. Jedoch muss die Option *--enable-sort* auf „true“ festgelegt sein (Standardeinstellung = „true“).

Für jede Region kann eine beliebige Kombination optionaler Berichte angefordert werden. Mehrere Berichtsarten für eine Region müssen durch Leerzeichen getrennt werden.

Die minimale MAPQ und die minimale BQ für eine bestimmte Region können mit qc-coverage-filters überschrieben werden.

Ein Coverage-Filter wird durch eine der *--qc-coverage-filters-i*-Optionen (i = 1, 2 oder 3) in Kombination mit der zugehörigen *--qc-coverage-region-i*-Option aktiviert:

- *--qc-coverage-region-i=<Zielregionen.bed>*
- *--qc-coverage-filters-i <Filterzeichenfolge>*

Beispielsweise wird mit den folgenden Optionen eine Coverage-Ausgabe mit einer Auflösung von 1 bp mit Filterung aktiviert:

```
--qc-coverage-region-1 <Zielregionen.bed>
--qc-coverage-filters-1 'mapq<10,bq<30'
--qc-coverage-reports-1 full_res
```

- Die Argumentsyntax lautet „mapq<Wert,bq<Wert“, d. h., Reads mit einer Mapping-Qualität und/oder Basen mit einer Base-Call-Qualität unter dem angegeben Wert werden nicht gezählt.
- Nur „mapq“ und „bq“ sind gültige Filterargumente. Es können jeweils ein Argument oder beide Argumente angegeben werden.
- Unterstützt wird ausschließlich der Operator <. Die Operatoren <=, >, >= und = werden nicht unterstützt.
- Wenn die Filterung für eine Zielregion aktiviert ist, gibt DRAGEN für diese Region gefilterte Berichtsdateien aus. Für gefilterte Zielregionen werden keine ungefilterten Berichtsdateien ausgegeben.

Callfähigkeit für anwendungsspezifische Regionen

Wenn die Option *--qc-coverage-region-i* zusammen mit *--qc-coverage-reports-i* (i = 1, 2 oder 3) verwendet wird, kann die Callfähigkeit als Berichtsart für die Region hinzugefügt werden. Bei der Ausgabe handelt es sich um eine qc-coverage-region-i_callability.bed-Datei. Die mit *--qc-coverage-region-padding-i*

angegebene Padding-Größe wird verwendet und überlappende Regionen werden zusammengefasst.

Die optionalen Filter für die minimale MAPQ und die minimale BQ beeinflussen ausschließlich die Zählung von Reads und Basen. Sie haben keine Auswirkungen auf die Berichte zur Callfähigkeit.

Verfügbare Berichtsarten

Tabelle 7 Standardbericht

Name des Berichts	DRAGEN-Ausgabedateiname/-typ
Coverage metrics	_coverage_metrics.csv
Fine histogram coverage	_fine_hist.csv
Histogram coverage	_hist.csv
Overall mean coverage	_overall_mean_cov.csv
Per contig mean coverage	_contig_mean_cov.csv
Predicted ploidy	_ploidy.csv

Tabelle 8 Optionale Berichte

Name des Berichts	DRAGEN-Ausgabedateityp
full_res	_full_res.bed
cov_report	_cov_report.bed
callability	_callability.bed

Coverage Metrics-Bericht

Der Coverage Metrics-Bericht wird als _coverage_metrics.csv-Datei ausgegeben und stellt Metriken über eine Region bereit. Bei der Region kann es sich um das Genom, eine Zielregion oder eine Coverage-Region für die Qualitätssicherung handeln. Die erste Spalte der Ausgabedatei enthält die Bezeichnung COVERAGE SUMMARY, die zweite Spalte ist für alle Metriken leer.

Folgende Kriterien gelten bei der Berechnung der Coverage:

- Doppelte Reads und geclippte Basen werden ignoriert.
- Es werden nur Reads mit MAPQ > min MAPQ und Basen mit BQ > min BQ berücksichtigt.

Folgende Metriken werden berechnet:

- Alignierte Basen in der Region: Anzahl der eindeutig auf die Region gemappten Basen und Prozentsatz in Relation zur Anzahl der eindeutig auf das Genom gemappten Basen.
- Durchschnittliche Alignment-Coverage über die Region: Anzahl der eindeutig auf die Region gemappten Basen geteilt durch die Anzahl der Stellen in der Region.
- Einheitlichkeit der Coverage (Prozentsatz > 0,2*Mittelwert) über die Region: Prozentsatz der Stellen mit einer Coverage größer als 20 % der mittleren Coverage in der Region.
- Prozentsatz der Region mit Coverage [ix, inf): Prozentsatz der Stellen in der Region mit einer Coverage von mindestens ix, wobei i gleich 100, 50, 20, 15, 10, 3, 1 und 0 sein kann.
- Prozentsatz der Region mit Coverage [ix, jx): Prozentsatz der Stellen in der Region mit einer Coverage von mindestens ix und weniger als jx, wobei (i, j) gleich (50, 100), (20, 50), (15, 20), (10, 15), (3, 10), (1, 3) und (0, 1) sein kann.

- Durchschnittliche X-Chromosom-Coverage über die Region: Gesamtanzahl der Basen, die mit dem Schnittpunkt des X-Chromosoms mit der Region aligniert sind, geteilt durch die Gesamtanzahl der Loci im Schnittpunkt des X-Chromosoms mit der Region. Ist kein X-Chromosom im Referenzgenom bzw. kein Schnittpunkt des X-Chromosoms mit der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Durchschnittliche Y-Chromosom-Coverage über die Region: Gesamtanzahl der Basen, die mit dem Schnittpunkt des Y-Chromosoms mit der Region aligniert sind, geteilt durch die Gesamtanzahl der Loci im Schnittpunkt des Y-Chromosoms mit der Region. Ist kein Y-Chromosom im Referenzgenom bzw. kein Schnittpunkt des Y-Chromosoms mit der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- XAvgCov/YAvgCov-Verhältnis über Genom/Zielregion: Durchschnittliche Alignment-Coverage für das X-Chromosom in der Region geteilt durch die durchschnittliche Alignment-Coverage für das Y-Chromosom in der Region. Ist kein X- oder Y-Chromosom im Referenzgenom bzw. kein Schnittpunkt des X- oder Y-Chromosoms mit der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Durchschnittliche Mitochondrien-Coverage über die Region: Gesamtanzahl der Basen, die mit dem Schnittpunkt des Mitochondrien-Chromosoms mit der Region aligniert sind, geteilt durch die Gesamtanzahl der Loci im Schnittpunkt des Mitochondrien-Chromosoms mit der Region. Ist kein Mitochondrien-Chromosom im Referenzgenom bzw. kein Schnittpunkt des Mitochondrien-Chromosoms mit der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Durchschnittliche Autosomen-Coverage über die Region: Gesamtanzahl der Basen, die mit den Autosomen-Loci in der Region aligniert sind, geteilt durch die Gesamtanzahl der Loci in den Autosomen-Loci in der Region. Ist kein Autosom im Referenzgenom bzw. kein Schnittpunkt zwischen Autosomen und der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Median der Autosomen-Coverage über die Region: Median der Alignment-Coverage über die Autosomen-Loci in der Region. Ist kein Autosom im Referenzgenom bzw. kein Schnittpunkt zwischen Autosomen und der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Mittelwert-Median-Verhältnis der Autosomen-Coverage über die Region: Mittlere Autosomen-Coverage in der Region geteilt durch den Median der Autosomen-Coverage in der Region. Ist kein Autosom im Referenzgenom bzw. kein Schnittpunkt zwischen Autosomen und der Region vorhanden, wird diese Metrik als „NA“ angezeigt.
- Alignierte Reads in der Region: Anzahl der eindeutig auf die Region gemappten Reads und Prozentsatz in Relation zur Anzahl der eindeutig auf das Genom gemappten Reads. Wenn es sich bei der Region um die Target-BED-Region handelt, ist diese Metrik vergleichbar mit der Erfassungsspezifität basierend auf der Zielregion und kann diese ersetzen.

Fine Histogram Coverage-Bericht

Der Fine Histogram Coverage-Bericht gibt eine _fine_hist.csv-Datei mit zwei Spalten aus: Depth und Overall. Der Wert in der Spalte Depth liegt zwischen 0 und 1000+. Die Spalte Overall gibt die Anzahl der bei der entsprechenden Tiefe abgedeckten Loci an.

Histogram Coverage-Bericht

Der Histogram Coverage-Bericht gibt eine _hist.csv-Datei mit folgendem Inhalt aus:

- Prozentualer Anteil der Basen im Genom/der Zielregion/der Coverage-Region in einem bestimmten Coverage-Bereich.
- Doppelte Reads werden ignoriert, wenn DRAGEN mit *--enable-duplicate-marking true* ausgeführt wird.

Folgende Bereiche werden verwendet:

```
"[100x:inf)", "[1x:3x)", "[0x:1x)"
```

Overall Mean Coverage-Bericht

Der Overall Mean Coverage-Bericht erstellt eine _overall_mean_cov.csv-Datei mit der durchschnittlichen Alignment-Coverage für Coverage-BED/Target-BED/WGS (sofern zutreffend).

Im Folgenden finden Sie ein Beispiel für den Inhalt der _overall_mean_cov.csv-Datei:

```
Average alignment coverage over target_bed,80.69
```

Per Contig Mean Coverage-Bericht

Der Per Contig Mean Coverage-Bericht generiert eine _contig_mean_cov.csv-Datei, die eine Prognose zur Coverage für alle Contigs sowie zur Autosomen-Coverage enthält. Die Datei enthält die folgenden drei Spalten:

Spalte 1	Spalte 2	Spalte 3
Contig-Bezeichnung	Anzahl der zu diesem Contig alignierten Basen, ausgenommen der Basen aus an Dubletten gekennzeichneten Reads, Reads mit MAPQ=0 und geclippten Basen.	Prognose der Coverage, angegeben wie folgt: <Anzahl der zu diesem Contig alignierten Basen (siehe Spalte 2)> geteilt durch <Länge des Contigs oder (wenn ein Target-BED verwendet wird) die Gesamtlänge der Zielregion für dieses Contig>.

Die Contig-Längen und die Längen der Target-BED-Regionen (als Denominatoren verwendet) enthalten Regionen mit N in der FASTA.

Predicted Ploidy-Bericht

Der Predicted Ploidy-Bericht gibt die prognostizierte Ploidie in einer _whg_ploidy.txt.Datei aus.

Im Folgenden finden Sie ein Beispiel für den Inhalt der _ploidy.txt-Datei:

```
Predicted sex chromosome ploidy XX
```

Full Res-Bericht

Der Full Res-Bericht wird als _full_res.bed-Datei ausgegeben. Hierbei handelt es sich um eine tabulatorgetrennte Datei mit den BED-Standardfeldern in den ersten drei Spalten sowie der Tiefe in der vierten Spalte. Jeder Datensatz in der Datei entspricht einem bestimmten Intervall mit konstanter Tiefe. Ändert sich die Tiefe, wird ein neuer Datensatz in die Datei geschrieben. Alignments mit einem Mapping-Qualitätswert von 0, doppelte Reads und geclippte Basen werden nicht in die Tiefe einberechnet.

Nur Basenpositionen in BED-Regionen der benutzerdefinierten Coverage-Region sind in der _full_res.bed-Ausgabedatei enthalten.

Die Struktur der Datei _full_res.bed entspricht derjenigen der Ausgabedatei von *bedtools genomecov -bg*. Die Inhalte sind identisch, wenn die Bedtools-Befehlszeile nach dem Herausfiltern von Alignments mit der Mapping-Qualität 0 und (wenn angegeben) nach Filterung nach Target-BED ausgeführt wird.

Im Folgenden finden Sie ein Beispiel für den Inhalt der Datei _full_res.bed:

```
chr1 121483984 121483985 10
chr1 121483985 121483986 9
chr1 121483986 121483989 8
chr1 121483989 121483991 7
```

```
chr1 121483991 121483992 6
chr1 121483992 121483993 4
chr1 121483993 121483994 2
chr1 121483994 121484039 1
chr1 121484039 121484043 2
chr1 121484043 121484048 3
chr1 121484048 121484050 7
chr1 121484050 121484051 11
chr1 121484051 121484052 17
chr1 121484052 121484053 149
chr1 121484053 121484054 323
chr1 121484054 121484055 2414
```

Coverage-Bericht

Der Bericht cov_report generiert eine _cov_report.bed-Datei. Hierbei handelt es sich um eine tabulatorgetrennte Datei mit den BED-Standardfeldern in den ersten drei Spalten. Die darauffolgenden Spaltenfelder sind über die in der gleichen Datensatzzeile angegebene Intervallregion berechnete Statistiken. Die zusätzlichen Spalten lauten wie folgt:

- total_cvg: Wert für die Gesamt-Coverage.
- mean_cvg: Mittel des Coverage-Werts.
- Q1_cvg: Coverage-Wert des unteren Quartils (25. Perzentile).
- median_cvg: Median des Coverage-Werts.
- Q3_cvg: Coverage-Wert des oberen Quartils (75. Perzentile).
- min_cvg: minimaler Coverage-Wert.
- max_cvg: maximaler Coverage-Wert.
- pct_above_X: Prozentsatz an Basen über die angegebene Intervallregion mit einer Coverage-Tiefe > X.

Standardmäßig wird bei einem Intervall mit einer Gesamt-Coverage von 0 der Datensatz in die Ausgabedatei aufgenommen. Legen Sie in der Konfigurationsdatei die Option *vc-emit-zero-coverage-intervals* auf „false“ fest, wenn Sie Intervalle mit einer Coverage von null herausfiltern möchten.

Im Folgenden finden Sie ein Beispiel für den Inhalt der _cov_report.bed-Datei:

```
chrom  start     end       total_cvg  mean_cvg  Q1_cvg  median_cvg  Q3_cvg  min_cvg  max_cvg  pct_above_5
…
chr5   34190121  34191570  76636      52.89     44.00   54.00       60.00   32       76       100.00
…
chr5   34191751  34192380  39994      63.58     57.00   61.00       69.00   50       85       100.00
…
chr5   34192440  34192642  10074      49.87     47.00   49.00       51.00   44       62       100.00
…
chr9   66456991  66457682  31926      46.20     39.00   45.00       52.00   27       65       100.00
…
chr9   68426500  68426601  4870       48.22     42.00   48.00       54.00   39       58       100.00
…
chr17  41465818  41466180  24470      67.60     4.00    66.00       124.00  2        153      66.30
```

```
…
chr20  29652081   29652203   5738       47.03     40.00   49.00       52.00   34       58       100.00
…
chr21  9826182    9826283    4160       41.19     23.00   52.00       58.00   5        60       99.01
…
```

Anwendungsfälle für Coverage-/Callfähigkeitsberichte und erwartete Ausgabe

In der folgenden Tabelle werden die Ausgaben mit Standardoptionen (*--vc-target-bed*) im Vergleich zu den optionalen Coverage-Regionsoptionen (*--coverage-region*) dargestellt.

--vc-target-bed angegeben? J/N	--qc- coverage-region-i (i = 1, 2 oder 3) angegeben? J/N	Erwartete Ausgabedateien
N	N	wgs_coverage_metrics.csv wgs_fine_hist.csv wgs_hist.csv wgs_overall_mean_cov.csv wgs_contig_mean_cov.csv wgs_ploidy.csv

--vc-target-bed angegeben? J/N	--qc- coverage-region-i (i = 1, 2 oder 3) angegeben? J/N	Erwartete Ausgabedateien
N	J	wgs_coverage_metrics.csv wgs_fine_hist.csv wgs_hist.csv wgs_overall_mean_cov.csv wgs_contig_mean_cov.csv wgs_ploidy.csv Für jede vom Benutzer angegebene Coverage-Region: qc-coverage-region-i_coverage_metrics.csv qc-coverage-region-i_fine_hist.csv qc-coverage-region-i_hist.csv qc-coverage-region-i_overall_mean_cov.csv qc-coverage-region-i_contig_mean_cov.csv qc-coverage-region-i_ploidy.csv qc-coverage-region-i_full_res.bed, wenn Berichtstyp full_res für qc-coverage-region-i angefordert qc-coverage-region-i_cov_report.bed, wenn Berichtstyp cov_report für qc-coverage-region-i angefordert qc-coverage-region-i_callability.bed, wenn GVCF-Modus aktiviert und Berichtstyp callability oder exome-callability angefordert
J	N	wgs_coverage_metrics.csv wgs_fine_hist.csv wgs_hist.csv wgs_overall_mean_cov.csv wgs_contig_mean_cov.csv wgs_ploidy.csv target_bed_coverage_metrics.csv target_bed_fine_hist.csvtarget_bed_hist.csv target_bed_overall_mean_cov.csv target_bed_contig_mean_cov.csv target_bed_ploidy.csv target_bed_callability.bed, wenn GVCF-Modus aktiviert

--vc-target-bed angegeben? J/N	--qc- coverage-region-i (i = 1, 2 oder 3) angegeben? J/N	Erwartete Ausgabedateien
J	J	wgs_coverage_metrics.csv wgs_fine_hist.csv wgs_hist.csv wgs_overall_mean_cov.csv wgs_contig_mean_cov.csv wgs_ploidy.csv target_bed_coverage_metrics.csv target_bed_fine_hist.csv target_bed_hist.csv target_bed_overall_mean_cov.csv target_bed_contig_mean_cov.csv target_bed_ploidy.csv target_bed_callability.bed, wenn GVCF-Modus aktiviert Für jede vom Benutzer angegebene Coverage-Region: qc-coverage-region-i _coverage_metrics.csv qc-coverage-region-i_fine_hist.csv qc-coverage-region-i_hist.csv qc-coverage-region-i_overall_mean_cov.csv qc-coverage-region-i_contig_mean_cov.csv qc-coverage-region-i_ploidy.csv qc-coverage-region-i_full_res.bed, wenn Berichtstyp full_ res für qc-coverage-region-i angefordert qc-coverage-region-i_cov_report.bed, wenn Berichtstyp cov_report für qc-coverage-region-i angefordert qc-coverage-region-i_callability.bed, wenn GVCF-Modus aktiviert und Berichtstyp callability oder exome-callability angefordert

BigWig-Komprimierung von Coverage-Metriken

Die BED-Ausgabedatei für Coverage-Metriken mit einer Auflösung von 1 bp (_full_res.bed) kann sehr groß werden. Die Komprimierung dieser Ausgabedatei in das BigWig-Format kann durch Festlegen der Option *--enable-metrics-compression* auf „true“ aktiviert werden.

Variant Quality Score Recalibration

Das Variant Quality Score Recalibration-Modul (VQSR) erstellt eine Metrik (VQSLOD), indem es Algorithmen für maschinelles Lernen auf eine VCF-Eingabedatei mit Trainingsdaten aus Datenbanken bekannter Varianten anwendet. Diese Metrik verbessert die Unterscheidung zwischen echten und falschen Varianten. Die VQSLOD-Metrik wird in jedem Varianten-Call-Datensatz dem Feld INFO hinzugefügt, wodurch zu einem späteren Zeitpunkt Calls auf Grundlage eines Schwellenwerts gefiltert werden können. Die VQSLOD-Metrik gibt das protokollierte Chancenverhältnis an, mit dem es sich bei dem Call um eine echte Variante bzw. eine falsche Variante handelt.

Algorithmus

Das VQSR-Modul erstellt anhand der Verteilung der Annotationswerte in einer Untergruppe mit Stellen von hoher Varianten-Call-Qualität ein Gaußsches Mischmodell. Diese Varianten-Call-Stellen werden anhand der angegebenen Ressourcendateien oder Trainingssätze (z. B. HapMap3 oder Omni 2.5M SNP) bestimmt. Über das Modell erhält jede Varianten-Call-Stelle des Eingabesatzes auf Grundlage der angegebenen Annotationen eine kovariierende Schätzung der Wahrscheinlichkeit, mit der es sich bei dem Call um eine echte genetische Variante handelt. Der Algorithmus implementiert die Variationsinferenzsversion des Algorithmus zur Erwartungsmaximierung über ein Gaußsches Mischmodell. Die Modellerstellung erfolgt iterativ, bis die Konvergenz erzielt ist. Wenn im Eingabedatensatz oder den Trainingssätzen zu wenige Stellen vorhanden sind, kann möglicherweise keine Konvergenz erzielt werden.

Für jeden erfolgreichen Lauf werden ein positives und ein negatives Modell erstellt. Das negative Modell wird anhand der Stellen mit der niedrigsten Performance mithilfe des Schwellenwerts *--vqsr-lod-cutoff* erstellt. Nach der Erstellung beider Modelle wird die Log-Likelihood berechnet, mit der eine Call-Stelle über jedes Modell generiert wurde. Anschließend wird der Log-Likelihood-Quotient übernommen. Der Log-Likelihood-Quotient wird in der VQSR VCF-Ausgabedatei in der Spalte INFO als „VQSLOD“ annotiert.

Sie können Call-Stellen mit einem bestimmten Schwellenwert weiter filtern, indem Sie Abschnittswerte festlegen, mit denen die Zielsensitivität angegeben wird. Anschließend berechnet das VQSR-Modul den VQSLOD-Mindestscore, der für diese Zielsensitivität erforderlich ist. Mit der Option *--vqsr-filter-level* wird der Schwellenwert für die Filterung von Calls bestimmt.

Nutzung und Einstellungen

Durch Aktivierung der VQSR-Nachverarbeitung wird die DRAGEN-Pipeline für eine Analyse des Gesamtgenoms nur unwesentlich verlängert. Die VQSLOD-Annotationen werden für SNPs und Indels berechnet und in der zusätzlichen VCF-Datei *<Ausgabedateipräfix>.vqsr.vcf* ausgegeben. Sie können VQSR mithilfe der Option *--enable-vqsr true* in der DRAGEN-Befehlszeile aktivieren.

Die VQSR-spezifischen Optionen lauten:

- *--enable-vqsr*

  Aktiviert das VQSR-Nachverarbeitungsmodul. Bei Verwendung mit der Option *--enable-variant-caller* oder *--enable-joint-genotyping* wird die VQSR-Verarbeitung nach dem Varianten-Calling ausgeführt. Bei Verwendung mit der Option *--vqsr-input* verarbeitet die VQSR-Engine die VCF-Dateien im eigenständigen Modus.

- *--vqsr-config*

  Legt den Pfad zur VQSR-Konfigurationsdatei fest. DRAGEN kann optional eine Konfigurationsdatei mit den VQSR-Optionen auslesen. Die in der Konfigurationsdatei festgelegten VQSR-Optionen können mit in der Befehlszeile angegebenen Optionen überschrieben werden. In der Konfigurationsdatei lassen sich Einstellungen speichern, die für mehrere Läufe benötigt werden, z. B. Abschnittswerte oder

Ressourcendateien. Ein Beispiel einer VSQR-Konfigurationsdatei bietet die Datei */opt/edico/config/dragen-VQSR.cfg*.

- *--vqsr-input*

  Legt die zu verarbeitende VCF-Eingabedatei für VQSR fest. Wenn diese Option in der DRAGEN-Befehlszeile festgelegt ist, wird VQSR im eigenständigen Modus ausgeführt. So kann VQSR für vorab erstellte VCF-Dateien ausgeführt werden.

- *--vqsr-annotation*

  Legt die Annotationen, die für das Erstellen von Modellen verwendet werden, als kommagetrennte Zeichenfolge fest, die den Betriebsmodus, gefolgt von einer Liste unter diesem Modus zu verwendender Annotationen, angibt. Beispiel: <Modus>,<Annotation>,<Annotation>…

  Als <Modus> wird entweder *SNP* oder *INDEL* angegeben. Wird nur *SNP* angegeben, können mit VQSR nur SNPs verarbeitet werden. Wird nur *INDEL* angegeben, können nur Indels verarbeitet werden. Sollen sowohl SNPs als auch Indels verarbeitet werden, kann diese Option zweimal in der Befehlszeile verwendet werden, um sowohl *SNP* als auch *INDEL* festzulegen.

  Die Liste mit den Annotationen wird aus der Spalte INFO einer VCF-Datei ausgelesen. Sie können maximal acht Annotationen angeben. Mit diesen Annotationen und deren zugewiesenen Werten wird das Modell erstellt.

  Im Folgenden finden Sie Beispieloptionen für VQSR-Annotationen:

  ```
  --vqsr-annotation SNP,DP,QD,FS,ReadPosRankSum,MQRankSum,MQ
  --vqsr-annotation INDEL,DP,QD,FS,ReadPosRankSum,MQRankSum
  ```

- *--vqsr-resource*

  Legt die Trainingsressourcendateien fest, mit deren Hilfe echte Varianten-Call-Stellen bestimmt werden. Diese Option lässt sich mehrfach festlegen, sodass mehrere Ressourcendateien mit jeweils anderem Wert für die A-priori-Wahrscheinlichkeit eingefügt werden können. DRAGEN unterscheidet nicht zwischen Echtheits- und Trainingsressourcendateien. Alle Ressourcendateien werden sowohl für Echtheitsbestimmungen als auch für Trainingszwecke verwendet.

  Mit dieser Option werden der Betriebsmodus, der Wert für die A-priori-Wahrscheinlichkeit zum Gewichten dieser Ressource und schließlich der Pfad der Ressourcendatei als kommagetrennte Zeichenfolge angegeben. Beispiel: <Modus>,<A-priori>,<Ressourcendatei>.

  Als <Modus> wird entweder *SNP* oder *INDEL* angegeben. Über <Modus> wird die Anwendung der Ressourcendateien festgelegt.

  <A-priori> legt die Gewichtung der Varianten-Call-Stellen mithilfe der angegebenen Ressourcendatei fest. Die Grundlage bildet die A-priori-Wahrscheinlichkeit, dass die Stellen korrekt sind.

  Über <Ressourcendatei> werden Pfad und Name der VCF-Ressourcendatei angegeben. Beispiel:

  ```
  --vqsr-resource "SNP,15.0,<path>/hapmap_3.3.vcf"
  --vqsr-resource "SNP,12.0,<path>/1000G_omni2.5.vcf"
  --vqsr-resource "SNP,10.0,<path>/1000G_phase1.snps.high_confidence.vcf"
  --vqsr-resource "INDEL,12.0,<path>/Mills_and_1000G_gold_
     standard.indels.vcf"
  ```

- *--vqsr-tranche*

  Gibt für die Berechnung der LOD-Schwellenwerte die Sensitivitätslevel für die Echtheit an. Diese Werte werden in Prozent angegeben. Diese Option kann mehrfach mit jeweils unterschiedlichem Sensitivitätslevel angegeben werden. Sofern nicht anders festgelegt, lauten die Standardwerte 100.0, 99.99, 99.90, 99.0 und 90.0. Beispiele:

  ```
  --vqsr-tranche 100.0
  --vqsr-tranche 99.99
  --vqsr-tranche 99.90
  --vqsr-tranche 99.00
  --vqsr-tranche 90.00
  ```

- *--vqsr-filter-level*

  Gibt für die Filterung von Varianten-Calls das Sensitivitätslevel für die Echtheit in Prozent an. Anhand des Sensitivitätslevels für die Echtheit wird der zugehörige VQSLOD-Mindestscore berechnet. Alle annotierten Calls unterhalb dieses Schwellenwerts werden als gefiltert gekennzeichnet. Das Feld FILTER wird über den Filter VQSLODThresholdSNP oder VQSLODThresholdINDEL als fehlgeschlagen gekennzeichnet. Ohne Angabe werden keine Calls gefiltert. Der Filterwert muss für SNPs und Indels separat angegeben werden. Beispiele:

  ```
  --vqsr-filter-level SNP,99.5
  --vqsr-filter-level INDEL,90.0
  ```

- *--vqsr-lod-cutoff*

  Gibt den LOD-Schwellenwert für die Auswahl der für die Erstellung des negativen Modells zu verwendenden Varianten-Call-Stellen an. Der Standardwert ist -5.0.

- *--vqsr-num-gaussians*

  Legt die Anzahl der Gaußschen Normalverteilungen zur Erstellung der positiven und negativen Modelle fest. Diese Option wird als kommagetrennte Zeichenfolge dieser vier ganzzahligen Werte angegeben: <SNP positiv>,<SNP negativ>,<INDEL positiv>,<INDEL negativ>. Ohne Angabe werden die Standardwerte 8,2,4,2 verwendet.

  Die Anzahl der pro Modell zu verwendenden Gaußschen Normalverteilungen muss größer als 0 sein und darf maximal 8 betragen. Es müssen mehr positive als negative Gaußsche Normalverteilungen verwendet werden. Um beispielsweise die Modelle mit 6 Gaußschen Normalverteilungen für „SNP positiv“, 2 für „SNP negativ“, 4 für „Indel positiv“ und 2 für „Indel negativ“ zu erstellen, verwenden Sie folgende Option:

  ```
  --vqsr-num-gaussians 6,2,4,2
  ```

- *--output-directory*

  Legt den Speicherort für die Ausgabedateien fest.

- *--output-file-prefix*

  Legt das Dateipräfix für sämtliche Ausgabedateien fest.

VSQR-Beispielausgabe

Wenn Sie bereits über eine VCF verfügen, die annotiert und gefiltert werden muss, kann das VQSR-Modul als eigenständiges Tool ausgeführt werden. Im Folgenden finden Sie ein Beispiel der Ausgabe:

```
================================================================
DRAGEN Variant Quality Score Recalibration

================================================================
Input file: /home/username/input.vcf
Output file: output/dragen.vqsr.vcf
Tranches: 100, 99.99, 99.9, 99, 90
Annotations:
  SNP: MQ FS QD MQRankSum ReadPosRankSum DP
  INDEL: FS QD MQRankSum ReadPosRankSum DP
Priors and training files:
  Q15.0:/variant_dbases/hapmap_3.3.vcf
  Q12.0:/variant_dbases/Mills_and_1000G_gold_standard.indels.vcf
  Q12.0:/variant_dbases/1000G_omni2.5.vcf
  Q10.0:/variant_dbases/1000G_phase1.snps.high_confidence.vcf
Number of Gaussians     SNP     INDEL
  Positive model:       8       4
  Negative model:       2       2

================================================================
Building SNP Training
Set============================================================

Number of valid records in input file: 5266144
Number of training records detected: 4170912
  WARNING: Annotation 'QD' was missing in 41 records.
  WARNING: Annotation 'ReadPosRankSum' was missing in 702218 records.
  WARNING: Annotation 'MQRankSum' was missing in 702185 records.
Training set statistics:
  DP -  mean: 140.838   std dev: 24.8402
  MQ -  mean: 59.9042   std dev: 0.946757
  QD -  mean: 22.7226   std dev: 6.67918
  FS -  mean: 3.01561   std dev: 4.18792  ReadPosRankSum -      mean: 0.73308
std dev: 0.964922

MQRankSum -   mean: 0.220358  std dev: 0.893418

Number of outliers removed from the full set: 363154 out of 5266144
Number of outliers removed from training set: 12046 out of 4170912

================================================================
Generating SNP Positive Model
================================================================
Number of Gaussians: 8
Number of data points: 4158866
```

```
Initializing model with k-means algorithm
K-means stabilized after 86 iterations
Running expectation-maximization algorithm
..........................................................
Expectation-maximization algorithm converged after 112 iterations

Cluster weight assigned to Gaussian #0 is 0.0109316
Cluster weight assigned to Gaussian #1 is 0.00988
Cluster weight assigned to Gaussian #2 is 0.646337
Cluster weight assigned to Gaussian #3 is 0.160654Cluster weight assigned to
Gaussian #4 is 0.00700244

Cluster weight assigned to Gaussian #5 is 0.005172
Cluster weight assigned to Gaussian #6 is 0.00252354
Cluster weight assigned to Gaussian #7 is 0.157501

================================================================
Building SNP Negative Training Set
================================================================
LOD Cutoff: -5
Number of data points: 359971

================================================================
Generating SNP Negative Model
================================================================
Number of Gaussians: 2
Number of data points: 359971

Initializing model with k-means algorithm
K-means stabilized after 15 iterations
Running expectation-maximization algorithm
.............
Expectation-maximization algorithm converged after 25 iterations

Cluster weight assigned to Gaussian #0 is 0.366887
Cluster weight assigned to Gaussian #1 is 0.633116

================================================================
Calculating SNP LOD Ratios
================================================================
Minimum LOD: -1317.29
Maximum LOD: 21.9196

================================================================
Calculating SNP Truth Sensitivity Tranches
================================================================
Tranche ts = 100.00     minVQSLOD = -1317.2891
Tranche ts = 99.99      minVQSLOD = -2.1371
```

```
Tranche ts = 99.90      minVQSLOD = -1.1866
Tranche ts = 99.00      minVQSLOD = 0.0199
Tranche ts = 90.00      minVQSLOD = 15.9062

================================================================
Building INDEL Training Set
================================================================
Number of valid records in input file: 1156226
Number of training records detected: 440621
  WARNING: Annotation 'QD' was missing in 41 records.
  WARNING: Annotation 'ReadPosRankSum' was missing in 77316 records.
  WARNING: Annotation 'MQRankSum' was missing in 76319 records.
Training set statistics:
  DP -  mean: 137.2     std dev: 41.05
  QD -  mean: 18.83     std dev: 8.258
  FS -  mean: 2.88      std dev: 4.756
  ReadPosRankSum -      mean: 0.2631    std dev: 0.9896
  MQRankSum -   mean: 0.1943    std dev: 0.907


Number of outliers removed from the full set: 8150 out of 1156226
Number of outliers removed from training set: 686 out of 440621

================================================================
Generating INDEL Positive Model
================================================================
Number of Gaussians: 4
Number of data points: 439935

Initializing model with k-means algorithmK-means stabilized after 52 iterations

Running expectation-maximization algorithm
..................
Expectation-maximization algorithm converged after 36 iterations

Cluster weight assigned to Gaussian #0 is 0.02934
Cluster weight assigned to Gaussian #1 is 0.2792
Cluster weight assigned to Gaussian #2 is 0.3979
Cluster weight assigned to Gaussian #3 is 0.2936

================================================================
Building INDEL Negative Training Set
================================================================
LOD Cutoff: -5
Number of data points: 61977

================================================================
Generating INDEL Negative Model
================================================================
```

```
Number of Gaussians: 2
Number of data points: 61977

Initializing model with k-means algorithm
K-means stabilized after 13 iterations
Running expectation-maximization algorithm
..................
Expectation-maximization algorithm converged after 38 iterations

Cluster weight assigned to Gaussian #0 is 0.5351
Cluster weight assigned to Gaussian #1 is 0.4649

================================================================
Calculating INDEL LOD Ratios
================================================================
Minimum LOD: -679.9
Maximum LOD: 6.154

================================================================
Calculating INDEL Truth Sensitivity Tranches
================================================================
Tranche ts = 100.00     minVQSLOD = -679.9446
Tranche ts = 99.99      minVQSLOD = -3.7305
Tranche ts = 99.90      minVQSLOD = -1.4809
Tranche ts = 99.00      minVQSLOD = -0.1606
Tranche ts = 90.00      minVQSLOD = 1.6201

================================================================
Merging SNP and INDEL Records
================================================================
Number of records processed as SNPs: 5266144
Number of records processed as INDELs: 1156226

================================================================
Generating Output VCF
================================================================
Number of total records from input file: 6422370
Number of records annotated with VQSLOD: 6422370
VQSR annotated VCF written to: output/dragen.vqsr.vcf
```

Virtual Long Read Detection

DRAGEN Virtual Long Read Detection (VLRD) ist ein alternativer und genauerer Varianten-Caller mit einem Fokus auf der Verarbeitung homologer/ähnlicher Regionen des Genoms. Ein herkömmlicher Varianten-Caller stützt sich auf den Mapper/Aligner, um zu ermitteln, welche Reads wahrscheinlich von einer bestimmten Position stammen. Er erkennt außerdem die zugrunde liegende Sequenz an dieser Position unabhängig von anderen Regionen, die nicht unmittelbar benachbart sind. Herkömmliche Varianten-Caller leisten gute Arbeit, wenn die Region von Interesse keiner anderen Region des Genoms über die Spanne eines Single-Reads (oder eines Paars von Reads bei Paired-End-Sequenzierung) ähnelt.

Allerdings trifft dieses Kriterium auf einen beträchtlichen Teil des Humangenoms nicht zu. Für viele Regionen des Genoms gibt es an anderer Stelle nahezu identische Kopien und infolgedessen bestehen bei der Ermittlung der Position der wahren Quelle möglicherweise erhebliche Unsicherheiten. Wenn beim Mapping einer Gruppe von Reads eine geringe Zuverlässigkeit vorliegt, ignoriert ein typischer Varianten-Caller die Reads möglicherweise, auch wenn sie nützliche Informationen enthalten. Wenn ein Read falsch gemappt wird (d. h., das primäre Alignment ist nicht die wahre Quelle des Reads), kann dies zu Fehlern bei der Erkennung führen. Short-Read-Sequenzierungstechnologien sind besonders anfällig für diese Probleme. Long-Read-Sequenzierung kann diesen Problemen entgegenwirken, aber sie ist in der Regel mit viel höheren Kosten und/oder höheren Fehlerraten sowie anderen Schwächen verbunden.

DRAGEN VLRD versucht, die Komplexitäten, die sich aus der Redundanz des Genoms ergeben, aus einer Perspektive zu meistern, die sich auf die Short-Read-Daten stützt. Anstatt jede Region isoliert zu betrachten, berücksichtigt VLRD alle Positionen, von denen eine Gruppe von Reads stammen könnte, und versucht, die zugrunde liegenden Sequenzen gemeinsam mithilfe aller verfügbaren Informationen zu erkennen.

Ausführen von DRAGEN-VLRD

Wie der Varianten-Caller von DRAGEN akzeptiert auch VLRD entweder FASTQ- oder sortierte BAM-Dateien als Eingabe und gibt eine VCF-Datei aus. VLRD unterstützt die Verarbeitung eines Satzes mit nur zwei homologen Regionen. DRAGEN kann keine Sätze mit drei oder mehr homologen Regionen verarbeiten. Künftige Versionen bieten Unterstützung für drei oder mehr homologe Regionen.

VLRD ist nicht standardmäßig aktiviert. Legen Sie die Option *--enable-vlrd* auf „true“ fest, um VLRD auszuführen. Das folgende Beispiel enthält einen DRAGEN-Befehl zum Ausführen von VLRD.

```
dragen \
   -r <REF> \
   -1 <FQ1> \
    -2 <FQ2> \
   --RGID <RG> --RGSM <SM> \
   --output-dir <AUSGABE> \
   --output-file-prefix <PRÄFIX> \
   --enable-map-align true \
   --enable-sort=true \
   --enable-duplicate-marking true \
   --enable-vlrd true
   --vc-target-bed similar_regions.bed
```

Aktualisierte Mapping-Alignment-Ausgabe für VLRD

DRAGEN-VLRD kann zusätzlich zur normalen Mapping-Alignment-Ausgabe von DRAGEN eine neu gemappte BAM-/SAM-Datei ausgeben. Legen Sie die Option *--enable-vlrd-map-align-output* auf „true“ fest, um die Ausgabe einer neu gemappten BAM-/SAM-Datei zu aktivieren. Die Standardeinstellung für diese Option ist „false“.

Die zusätzliche Mapping-Alignment-Ausgabe von VLRD enthält ausschließlich Reads, die auf mit VLRD verarbeitete Regionen gemappt wurden.

VLRD aktualisiert die Read-Alignments (Mapping-Position und/oder Mapping-Qualität usw.) anhand der zu allen homologen Regionen verfügbaren Informationen. Die aktualisierte Mapping-Alignment-Ausgabe von VLRD ist besonders für Pile-up-Analysen mit homologen Regionen geeignet.

VLRD-Einstellungen

Die DRAGEN-Hostsoftware verfügt über die folgenden spezifischen VLRD-Optionen.

- *--enable-vlrd*

  Ist diese Option auf „true“ festgelegt, wird VLRD für die DRAGEN-Pipeline aktiviert.

- *--vc-target-bed*

  Gibt die BED-Eingabedatei an. DRAGEN erfordert eine Target-BED-Eingabedatei, in der die homologen Regionen angegeben sind, die von VLRD verarbeitet werden sollen. Diese BED-Datei verfügt über ein spezielles, für VLRD erforderliches Format, um die homologen Regionen korrekt verarbeiten zu können. Die maximale Länge der von VLRD verarbeiteten Region beträgt 900 bp.

  Beispiel:

  ```
  chr1    161497562   161498362   0   0
  chr1    161579204   161580004   0   0
  chr1    21750837     21751637   1   0
  chr1    21809355     21810155   1   1
  ```

  - Die ersten drei Spalten sind mit herkömmlichen BED-Dateien identisch: Spalte 1 ist die Chromosomenbeschreibung, Spalte 2 der Start der Region und Spalte 3 das Ende der Region.
  - Spalte 4 ist die Gruppen-ID der homologen Region. Sie dient zur Gruppierung von Regionen, die zueinander homolog sind.
  - Die Zeilen 1 und 2 weisen in Spalte 4 den gleichen Wert auf und sollten daher als ein Satz von homologen Regionen sowie unabhängig von der nächsten Gruppe in den Zeilen 3 und 4 verarbeitet werden. Ohne eine korrekte Festlegung werden von der Software jedoch möglicherweise Regionen gruppiert, die nicht zueinander homolog sind. Dies führt zu falschen Varianten-Calls.
  - Spalte 5 gibt an, ob eine Region in Bezug auf die anderen homologen Regionen ein umgekehrtes Komplement ist. Ein Wert von 1 bedeutet, dass die Region in Bezug auf die anderen homologen Regionen der gleichen Gruppe ein umgekehrtes Komplement ist.
  - Zeile 4, Spalte 5 ist auf 1 festgelegt. Dies weist darauf hin, dass die Region nur als umgekehrtes Komplement zur Region in Zeile 3 homolog ist.

  Das DRAGEN-Installationspaket umfasst zwei VLRD-BED-Dateien für hg19- und hs37d5-Referenzgenome unter /opt/edico/examples/VLRD. Sie können diese BED-Dateien ohne Änderung zum Ausführen von VLRD oder als Beispiel zum Erstellen einer benutzerdefinierten BED-Datei verwenden.

- *--enable-vlrd-map-align-output*

  Ist diese Option auf „true“ festgelegt, gibt VLRD eine erneut gemappte BAM-/SAM-Datei aus, die nur Reads enthält, die auf von VLRD verarbeiteten Regionen gemappt wurden.

Erzwingen der Genotypisierung

DRAGEN unterstützt jetzt für das Varianten-Calling von Keimbahn-SNVs das Erzwingen der Genotypisierung (ForceGT). Wenn Sie ForceGT verwenden möchten, nutzen Sie die Option *--vc-forcegt-vcf* mit einer Liste kleiner Varianten zum Erzwingen der Genotypisierung. Bei der Eingabeliste mit kleinen Varianten kann es sich um eine .vcf- oder eine .vcf.gz-Datei handeln.

Hinsichtlich ForceGT gelten derzeit folgende Einschränkungen:

- Für das Varianten-Calling von Keimbahn-SNVs wird ForceGT im V3-Modus unterstützt. Die Modi V1, V2 und V2+ werden nicht unterstützt.
- Für das Varianten-Calling von somatischen SNVs wird ForceGT nicht unterstützt.
- ForceGT-Varianten werden bei der gemeinsamen Genotypisierung nicht weitergegeben.

ForceGT-Eingabe

Die DRAGEN-Software unterstützt ausschließlich eine einzelne ForceGT-VCF-Eingabedatei, die folgende Anforderungen erfüllen muss:

- Sie weist dieselben Referenz-Contigs auf wie die für das Varianten-Calling verwendete VCF.
- Sie ist nach Name und Position der Referenz-Contigs sortiert.
- Sie ist normalisiert (nach dem Prinzip der Sparsamkeit und linksbündig ausgerichtet).
- Sie enthält keine komplexen Varianten (Varianten, die mehr als eine Substitution/Insertion/Deletion für den Schritt vom REF-Allel zum ALT-Allel erfordern). Beispielsweise verursachen alle mit der folgenden vergleichbaren Varianten in der ForceGT-VCF ein nicht definiertes Verhalten in der DRAGEN-Software:

  ```
  chrX 153592402 GTTGGGGATGCTGAC CACCCTGAAGGG
  ```

Die folgenden nicht normalisierten Varianten verursachen ein nicht definiertes Verhalten in der DRAGEN-Software:

- Angabe, die das Prinzip der Sparsamkeit nicht befolgt: `chrX 153592402 GC GCG`
- Angabe nach dem Prinzip der Sparsamkeit: `chrX 153592403 C CG`

ForceGT-Vorgang und erwartetes Ergebnis

Wenn das Keimbahn-SNV-Varianten-Calling mit ForceGT erfolgt, wird mithilfe der ForceGT-VCF als Eingabe in der DRAGEN-Befehlszeile eine Einzelproben-gVCF generiert. Die Einzelproben-gVCF-Ausgabedatei enthält alle normalen und ForceGT-Calls wie folgt:

- Wenn ein ForceGT-Call vom Varianten-Caller nicht ermittelt wurde (keine Gemeinsamkeit), wird der Call im Feld INFO mit FGT gekennzeichnet.
- Wenn ein ForceGT-Call auch mit dem Varianten-Caller ermittelt und das Feld FILTER auf PASS gesetzt wurde (Gemeinsamkeit), wird der Call im Feld INFO mit NML:FGT gekennzeichnet. (NML steht für „normal“.)
- Einem normalen Call (und PASS) mit dem Varianten-Caller ohne ForceGT-Call (Normalfall) werden keine zusätzlichen Tags (NML oder FGT) hinzugefügt.

Dieses Schema ermöglicht die Unterscheidung zwischen Calls, die nur aufgrund von FGT vorhanden sind, Calls, die sowohl in der ForceGT-Eingabe und dem normalen Calling vorhanden sind, und normalen Calls.

Alle Varianten in der ForceGT-Eingabe-VCF sind genotypisiert und in der Einzelproben-gVCF-Ausgabedatei enthalten. Der GT für die Varianten wird wie folgt aufgeführt:

Bedingung	Aufgeführter GT
An einer Position ohne Coverage	./.
An einer Position mit Coverage, jedoch ohne Reads, die das ALT-Allel unterstützen	0/0
An einer Position mit Coverage und Reads, die das ALT-Allel unterstützen	0/0 oder 0/1 oder 1/1 oder 1/2

An einer Position, an der sich der Varianten-Call mit der standardmäßigen DRAGEN-Software von dem in der ForceGT-Eingabe-VCF unterscheidet, enthält die Ausgabe-gVCF mehrere Einträge für dieselbe Position:

- Einen Eintrag für den standardmäßigen Varianten-Call von DRAGEN und
- jeweils einen Eintrag für jeden Varianten-Call, der in der ForceGT-Eingabe-VCF für diese Position enthalten ist.

```
chrX 100 G C [Default DRAGEN variant call]
chrX 100 G A [Variant in ForceGT vcf]
```

Wenn mit der ForceGT-Eingabe-VCF eine Target-BED-Datei bereitgestellt wird, enthält die gVCF-Ausgabedatei nur ForceGT-Varianten, die mit den Positionen in der BED-Datei überlappen.

Unique Molecular Identifiers

DRAGEN 3.3 und neuere Versionen beinhalten eine Beta-Version der UMI-Pipeline.

Die DRAGEN UMI-Pipeline unterstützt die TruSight Oncology (TSO) UMI-Reagenzien, die mithilfe von UMIs (Unique Molecular Identifiers, eindeutige molekulare Identifikatoren) für eine geringere PCR- und Sequenzierungsfehlerrate sorgen. Dank geringerer Fehlerraten lassen sich seltene und niedrigfrequente somatische Varianten aus DNA-Proben wie etwa aus Plasma isolierter cfDNA ermitteln. In der UMI-Pipeline aligniert DRAGEN die Reads zunächst und gruppiert sie nach UMI und Alignment. Das System generiert dann für jede Gruppe eine einzelne Konsensussequenz. Diese generierten Reads weisen höhere Qualitätsscores auf, da jeder Base-Call auf der Kombination mehrerer Beobachtungen beruht.

Für die UMI-Pipeline eignen sich Eingabe-Reads aus einem Paired-End-Lauf. Die UMI-Sequenzen müssen im achten durch einen Doppelpunkt abgetrennten Feld des QNAME aufgeführt sein. Die UMIs müssen als Paar vorliegen und durch ein „+“ getrennt sein. Das folgende Beispiel zeigt den an das Ende des Read-Namens angefügten UMI für beide Reads des Paars:

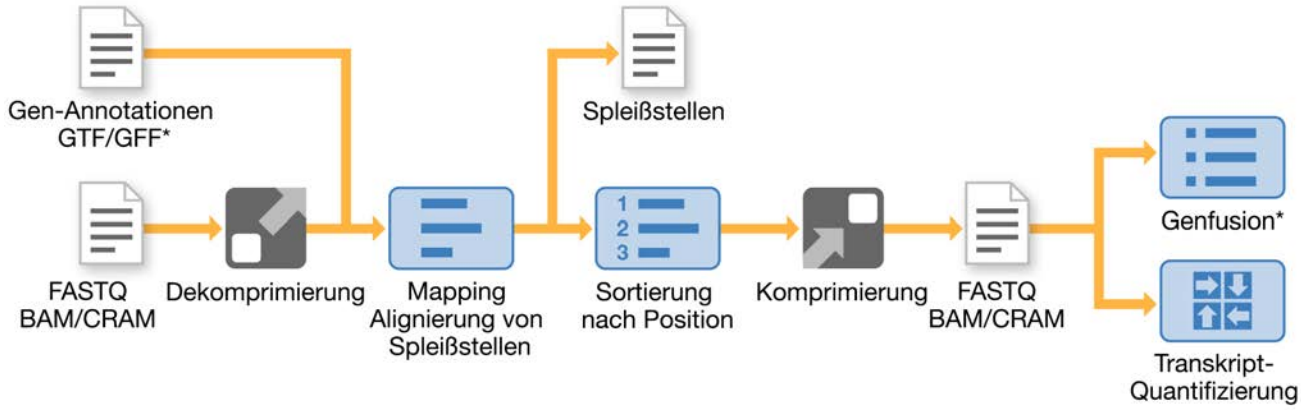
```
NDX550136:7:H2MTNBDXX:1:13302:3141:10799:AAGGATG+TCGGAGA
```

Aktivieren Sie die Option *--enable-umi*, um das Kollabieren von Reads zu ermöglichen. Diese Option kann nicht zusammen mit *–enable-duplicate-marking* verwendet werden, da die UMI-Pipeline einen Konsensus-Read aus einem Satz von Eingabe-Reads generiert, anstatt den besten nicht doppelt vorliegenden Read auszuwählen. Verwenden Sie die Option *–umi-min-supporting-reads*, um die für einen Konsensus-Read erforderliche Anzahl von Eingabe-Reads mit übereinstimmenden UMIs festzulegen. Der folgende Beispielbefehl sorgt für die Ausführung der DRAGEN UMI-Pipeline:

```
/opt/edico/bin/dragen \
   -r <REFERENZ> \
   -1 <FQ1> \
   -2 <FQ2> \
   --output-dir <AUSGABE> \
   --output-file-prefix <PRÄFIX> \
   --enable-map-align true \
   --enable-sort true \
   --enable-umi true \
   --umi-min-supporting-reads 2
```

Kapitel 4 DRAGEN RNA-Pipeline

DRAGEN umfasst einen RNA-Seq-Aligner (spleiß-sensibel) sowie RNA-spezifische Analysekomponenten für die Genexpressionsquantifizierung und die Genfusionserkennung.



Die meisten der unter „Hostsoftware-Optionen“ und „DNA-Mapping“ erläuterten Funktionen und Optionen gelten auch für RNA-Anwendungen. In diesem Abschnitt werden zusätzliche RNA-spezifische Aspekte beschrieben.

Eingabedateien

Gen-Annotationsdatei

Zusätzlich zu den Standardeingabedateien (Reads aus FASTQ- oder BAM-Dateien, Referenzgenom usw.) akzeptiert DRAGEN auch eine Gen-Annotationsdatei als Eingabedatei. Eine Gen-Annotationsdatei hilft beim Alignment von Reads mit Spleißstellen und wird für die Quantifizierung von Expressionen und die Erkennung von Genfusionen verwendet.

Verwenden Sie zum Festlegen einer Gen-Annotationsdatei die Befehlszeilenoption *-a* (*--annotation-file*). Die Eingabedatei muss den GTF-/GFF-Spezifikationen entsprechen (http://uswest.ensembl.org/info/website/upload/gff.html). Die Datei muss Funktionen vom Typ exon und der Datensatz Attribute vom Typ gene_id und transcript_id enthalten. Im folgenden Beispiel ist eine gültige GTF-Datei aufgeführt.

```
chr1    HAVANA  transcript  11869   14409   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000456328.2"; …
chr1    HAVANA  exon        11869   12227   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000456328.2"; …
chr1    HAVANA  exon        12613   12721   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000456328.2"; …
chr1    HAVANA  exon        13221   14409   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000456328.2"; …
chr1    ENSEMBL transcript  11872   14412   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000515242.2"; …
chr1    ENSEMBL exon        11872   12227   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000515242.2"; …
chr1    ENSEMBL exon        12613   12721   .   +   .   gene_id
"ENSG00000223972.4"; transcript_id "ENST00000515242.2"; …
chr1    ENSEMBL exon        13225   14412   .   +   .   gene_id
```

```
"ENSG00000223972.4"; transcript_id "ENST00000515242.2"; …
…
```

Entsprechend kann auch eine GFF-Datei verwendet werden. Jede exon-Funktion muss als Parent über einen Transkript-Bezeichner verfügen, der zur Gruppierung von Exons verwendet wird. Im folgenden Beispiel ist eine gültige GFF-Datei aufgeführt.

```
1   ensembl_havana  processed_transcript    11869   14409       .   +   .
ID=transcript:ENST00000456328;
1   havana          exon                    11869   12227       .   +   .
Parent=transcript:ENST00000456328; …
1   havana          exon                    12613   12721       .   +   .
Parent=transcript:ENST00000456328; …
1   havana          exon                    13221   14409       .   +   .
Parent=transcript:ENST00000456328; …
…
```

Die DRAGEN-Hostsoftware analysiert die Datei nach Exons innerhalb der Transkripte und erzeugt Spleißstellen. In der folgenden Ausgabe wird die Anzahl der erkannten Spleißstellen angezeigt.

```
================================================================
Generating annotated splice junctions
================================================================
Input annotations file: ./gencode.v19.annotation.gtf
Splice junctions database file: output/rna.sjdb.annotations.out.tab

 Number of genes: 27459

 Number of transcripts: 196520
 Number of exons: 1196293
 Number of splice junctions: 343856
```

Die erkannten Spleißstellen werden auch in eine Ausgabedatei (*.sjdb.annotations.out.tab) geschrieben, die der Benutzer anzeigen kann. Bei der Bildung der Spleißstellen aus der Gen-Annotationseingabedatei wird ein einfacher Filter verwendet, durch den alle Spleißstellen verworfen werden, die nicht der erforderlichen Mindestlänge entsprechen. Dadurch kann die Rate falscher Erkennungen für falsch annotierte Stellen reduziert werden. Die minimale Spleißstellen-Annotationslänge wird über die Option *--rna-ann-sj-min-len* geregelt, die einen Standardwert von 6 aufweist.

Ist *--rna-gf-restrict-genes* auf „true“ festgelegt, filtert DRAGEN auch Transkripte mit einer Biotyp-Einstellung, die weder lincRNA noch protein_coding lautet. Legen Sie diese Option auf „false“ fest, wenn Sie diesen Filter deaktivieren möchten. Unter https://www.gencodegenes.org/pages/biotypes.html finden Sie Informationen zu Biotypen in der Gencode-/Ensembl-Annotation.

Two-Pass-Modus

Die DRAGEN-Software kann neben der Verwendung einer GTF-/GFF-Datei für annotierte Spleißstellen auch eine SJ.out.tab-Datei einlesen (siehe *SJ.out.tab* auf Seite 116). Diese Datei ermöglicht das Ausführen von DRAGEN im Two-Pass-Modus, bei dem die im ersten erfolgreichen Lauf entdeckten Spleißstellen (Ausgabe

als SJ.out.tab-Datei) verwendet werden, um die Mapping- und Alignment-Reads während des zweiten Laufs in DRAGEN auszurichten. Dieser Betriebsmodus ist hilfreich, um die Sensitivität für gespleißte Alignments für Fälle zu erhöhen, in denen keine Gen-Annotationsdatei für das Zielgenom verfügbar ist.

RNA-Alignment

Die DRAGEN-RNA-Pipeline verwendet den DRAGEN-RNA-Seq-Spliced-Aligner. Das Mapping von kurzen Seed-Sequenzen aus RNA-Seq-Reads ist vergleichbar mit dem Mapping von DNA-Reads. Zusätzlich werden Spleißstellen (Verbindungsstellen nicht benachbarter Exons in RNA-Transkripten) in der Nähe der gemappten Seeds erkannt und in die vollständigen Read-Alignments aufgenommen.

Alignment-Ausgabe

Die beim Ausführen von DRAGEN im RNA-Modus generierten Ausgabedateien sind mit denen vergleichbar, die im DNA-Modus generiert werden. Im RNA-Modus werden darüber hinaus zusätzliche Informationen zu gespleißten Alignments generiert. Ausführliche Informationen zu den Spleißstellen sind im SAM-Alignment-Datensatz und der SJ.out.tab-Datei enthalten.

BAM

Die BAM-Ausgabedatei erfüllt die SAM-Spezifikation und ist mit nachgeschalteten RNA-Seq-Analysetools kompatibel.

RNA-Seq-BAM-Tags

Die folgenden BAM-Tags werden zusammen mit gespleißten Alignments ausgegeben.

- **XS:A**: Das XS-Tag bezeichnet die Strangausrichtung eines Introns. Siehe *Kompatibilität mit Cufflinks* auf Seite 116.
- **jM:B**: Das jM-Tag führt die Intron-Motive für alle Verknüpfungen in den Alignments auf. Es gelten folgende Definitionen:
  - 0: nicht kanonisch
  - 1: GT/AG
  - 2: CT/AC
  - 3: GC/AG
  - 4: CT/GC
  - 5: AT/AC
  - 6: GT/AT

  Wenn während der Mapping/Alignment-Phase eine Gen-Annotationsdatei verwendet wird und die Spleißstelle als annotierte Verknüpfung erkannt wird, wird zum Motivwert 20 addiert.

NH:i: Ein Standard-SAM-Tag, das die Anzahl der berichteten Alignments anzeigt und die Abfrage im aktuellen Datensatz enthält. Dieses Tag kann für nachgeordnete Tools wie featureCounts verwendet werden.

HI:i: Ein Standard-SAM-Tag, das den Abfragetrefferindex angibt und dessen Wert anzeigt, dass dieses Alignment das *i*-te im SAM gespeicherte ist. Der zugehörige Wert liegt zwischen 1 und NH. Dieses Tag kann für nachgeordnete Tools wie featureCounts verwendet werden.

Kompatibilität mit Cufflinks

Cufflinks erfordert u. U. gespleißte Alignments zur Ausgabe des Strang-Tags XS:A. Dieses Tag ist im SAM-Datensatz enthalten, wenn das Alignment eine Spleißstelle enthält. Das Strang-Tag XS:A kann folgende Werte annehmen:

„.“ (nicht definiert), „+“ (Vorwärtsstrang), „-“ (Rückwärtsstrang) oder „*“ (mehrdeutig).

Wenn das gespleißte Alignment einen nicht definierten Strang enthält oder ein Strangkonflikt vorliegt, kann das Alignment durch Festlegen der Option *--no-ambig-strand* auf 1 unterbunden werden.

Außerdem muss für Cufflinks die MAPQ für einen eindeutig gemappten Read als einzelner Wert vorliegen. Dieser Wert wird mit der Option *--rna-mapq-unique* angegeben. Die Übernahme eines MAPQ-Werts für alle eindeutig gemappten Reads lässt sich durch Festlegen von *--rna-mapq-unique* auf einen Wert ungleich null erzwingen.

SJ.out.tab

Neben den in der SAM-/BAM-Datei ausgegebenen Alignments werden in einer zusätzlichen SJ.out.tab-Datei die Spleißstellen mit hoher Konfidenz als tabulatorgetrennte Datei ausgegeben. Die einzelnen Spalten dieser Datei enthalten Folgendes:

1 Contig-Bezeichnung

2 erste Base der Spleißstelle (1-Base)

3 letzte Base der Spleißstelle (1-Basen-)Strang (0: nicht definiert, 1: +, 2: -)

4 Intron-Motiv: 0: nicht kanonisch, 1: GT/AG, 2: CT/AC, 3: GC/AG, 4: CT/GC, 5: AT/AC, 6: GT/AT

5 0: nicht annotiert, 1: annotiert, nur bei Verwendung einer Gen-Annotationsdatei als Eingabe

6 Anzahl der eindeutig gemappten Reads, die die Spleißstelle umspannen

7 Anzahl der mehrfach gemappten Reads, die die Spleißstelle umspannen

8 maximaler Überhang gespleißter Alignments

Das Feld mit dem maximalen Überhang gespleißter Alignments (Spalte 8) in der SJ.out.tab-Datei ist der Überhang verankerter Alignments. Bei einem als ACGTACGT----------ACGT gespleißten Read ist der Überhang 4. Der maximale Überhang wird für dieselbe Spleißstelle über alle Reads in den Bericht aufgenommen, die diese Stelle umspannen. Der maximale Überhang ist ein Zuverlässigkeitsindikator dafür, dass – basierend auf verankerten Alignments – die Spleißstelle korrekt ist.

Von der DRAGEN-Hostsoftware werden zwei SJ.out.tab-Dateien generiert, eine nicht gefilterte und eine gefilterte Version. Die Datensätze in der nicht gefilterten Datei sind eine Zusammenfassung aller Datensätze zu gespleißten Alignments aus der SAM-/BAM-Ausgabe. Die gefilterte Version ist jedoch aufgrund der Verwendung der folgenden Filter mit deutlich höherer Zuverlässigkeit korrekt.

Der Eintrag einer Spleißstelle in der SJ.out.tab-Datei wird herausgefiltert, wenn eine *beliebige* der folgenden Bedingungen erfüllt ist:

- SJ ist ein nicht kanonisches Motiv und wird nur von < 3 eindeutigen Mappings unterstützt.
- SJ verfügt über eine Länge > 50.000 und wird nur von < 2 eindeutigen Mappings unterstützt.
- SJ verfügt über eine Länge > 100.000 und wird nur von < 3 eindeutigen Mappings unterstützt.
- SJ verfügt über eine Länge > 200.000 und wird nur von < 4 eindeutigen Mappings unterstützt.
- SJ ist ein nicht kanonisches Motiv und der maximale Überhang gespleißter Alignments ist < 30.
- SJ ist ein kanonisches Motiv und der maximale Überhang gespleißter Alignments ist < 12.

Die gefilterte SJ.out.tab-Datei wird zur Verwendung mit einem beliebigen nachgeschalteten Analyse- oder Nachverarbeitungstool empfohlen. Bei Verwendung der ungefilterten SJ.out.tab-Datei können Sie Ihre eigenen Filter anwenden (z. B. mit grundlegenden awk-Befehlen).

Beachten Sie, dass der Filter nicht auf die in der BAM- oder SAM-Datei vorhandenen Alignments angewendet werden kann.

Chimeric.out.junction-Datei

Wenn in der Probe chimärische Alignments vorhanden sind, wird zusätzlich eine Chimeric.out.junction-Datei ausgegeben. In dieser Datei sind Informationen zu Split-Reads enthalten, mit denen eine anschließende Erkennung von Genfusionen durchgeführt werden kann. Jede Zeile enthält einen chimärisch alignierten Read. Die Datei enthält die folgenden Spalten:

1 Chromosom des Spenders.

2 Erste Base des Introns des Spenders (1-Base).

3 Strang des Spenders.

4 Chromosom des Empfängers.

5 Erste Base des Introns des Empfängers (1-Base).

6 Strang des Empfängers.

7 N. z.: wird nicht verwendet und ist nur aus Gründen der Kompatibilität mit anderen Tools vorhanden. Wert ist stets 1.

8 N. z.: wird nicht verwendet und ist nur aus Gründen der Kompatibilität mit anderen Tools vorhanden. Wert ist stets *.

9 N. z.: wird nicht verwendet und ist nur aus Gründen der Kompatibilität mit anderen Tools vorhanden. Wert ist stets *.

10 Read-Name.

11 Erste Base des ersten Segments auf dem +Strang.

12 CIGAR des ersten Segments.

13 Erste Base des zweiten Segments.

14 CIGAR des zweiten Segments.

CIGARs in dieser Datei entsprechen den CIGAR-Standardoperationen laut SAM-Spezifikation, mit einer zusätzlichen Lückenlänge L, die mit der Operation p codiert wird. Bei Paired-End-Reads wird die Sequenz für den zweiten Mate stets für den Gegenstrang durchgeführt, bevor die Strängigkeit bestimmt wird.

Im Folgenden finden Sie einen Beispieleintrag mit zwei chimärisch alignierten Read-Paaren mit einem getrennten Mate, in dem Segmente von chr19 zu chr12 gemappt werden. Außerdem sind die zugehörigen SAM-Datensätze dargestellt, die mit diesen Einträgen verknüpft sind.

```
chr19 580462 + chr12 120876182 + 1 * * R_15448 571532 49M8799N26M8p49M26S
120876183 49H26M
chr19 580462 + chr12 120876182 + 1 * * R_15459 571552 29M8799N46M8p29M46S
120876183 29H46M

R_15448:1   99    chr19   571531      60  49M8799N26M  =       580413
R_15448:2   147   chr19   580413      60  49M26S       =       571531
R_15448:2   2193  chr12   120876182   15  49H26M       chr19   571531
```

```
R_15459:1   99    chr19   571551      60  29M8799N46M  =       580433
R_15459:2   147   chr19   580433      4   29M46S       =       571551
R_15459:2   2193  chr12   120876182   15  29H46M       chr19   571551
```

RNA-Alignment-Optionen

In der Aligner-Phase des RNA-Spliced-Aligners werden die Optionen für das Alignment-Scoring nach Smith-Waterman sowie Bewertungsoptionen für das Spleißen verwendet.

Optionen für das Alignment-Scoring nach Smith-Waterman

Ausführliche Informationen über den in DRAGEN verwendeten Alignment-Algorithmus finden Sie unter *Einstellungen für das Alignment-Scoring nach Smith-Waterman* auf Seite 19. Die folgenden Scoring-Optionen sind spezifisch für die Verarbeitung kanonischer und nicht kanonischer Motive in Introns.

- *--Aligner.intron-motif12-pen*

  Die Option *--Aligner.intron-motif12-pen* regelt den Abzug für die kanonischen Motive 1/2 (GT/AG, CT/AC). Die Hostsoftware berechnet den Standardwert wie folgt: 1*(match-score + mismatch-pen).

- *--Aligner.intron-motif34-pen*

  Die Option *--Aligner.intron-motif34-pen* regelt den Abzug für die kanonischen Motive 3/4 (GC/AG, CT/GC). Die Hostsoftware berechnet den Standardwert wie folgt: 3*(match-score + mismatch-pen).

- *--Aligner.intron-motif56-pen*

  Die Option *--Aligner.intron-motif56-pen* regelt den Abzug für die kanonischen Motive 5/6 (AT/AC, GT/AT). Die Hostsoftware berechnet den Standardwert wie folgt: 4*(match-score + mismatch-pen).

- *--Aligner.intron-motif0-pen*

  Die Option *--Aligner.intron-motif0-pen* regelt den Abzug für nicht kanonische Motive. Die Hostsoftware berechnet den Standardwert wie folgt: 6*(match-score + mismatch-pen).

Spleiß-Score-Optionen

- *--Mapper.min-intron-bases*

  Im Rahmen des RNA-Seq-Mappings kann eine Referenz-Alignment-Lücke als Deletion oder Intron interpretiert werden. Ist keine annotierte Spleißstelle vorhanden, wird mit der Option min-intron-bases ein Schwellenwert für die Lückenlänge festgelegt. Referenzlücken mit einer Länge ab diesem Schwellenwert werden als Introns interpretiert und bewertet. Referenzlücken mit einer geringeren Länge werden als Deletionen interpretiert und bewertet. Alignments können jedoch mit annotierten Spleißstellen zurückgegeben werden, deren Länge diesen Schwellenwert unterschreitet.

- *--Mapper.max-intron-bases*

  Mit der Option max-intron-bases wird das größtmögliche protokollierte Intron gesteuert. So lassen sich falsche Spleißstellen im Bericht vermeiden. Legen Sie für diese Option einen Wert fest, der für die zu mappende Spezies geeignet ist.

- *--Mapper.ann-sj-max-indel*

  Im Rahmen der RNA-Sequenzierung kann durch Seed-Mapping eine Referenzlücke an der Position eines annotierten Introns ermittelt werden, die Länge weicht jedoch geringfügig ab. Falls diese Abweichung nicht den mit der Option festgelegten Wert überschreitet, prüft der Mapper, ob das Intron exakt gemäß der Annotation vorhanden ist und die Längenabweichung durch ein Indel an einem der beiden Enden der

Spleißstelle verursacht wird. Indels, die den mit der Option festgelegten Wert überschreiten, und nahe gelegene Spleißstellen werden wahrscheinlich nicht erkannt. Das Festlegen hoher Werte verlängert möglicherweise die Dauer des Mappings und kann die Zahl falscher Erkennungen steigern.

MAPQ-Scoring

Standardmäßig erfolgt die MAPQ-Berechnung für die RNA-Sequenzierung wie bei der DNA-Sequenzierung. Wichtigster Faktor bei der MAPQ-Berechnung ist die Differenz zwischen dem besten und dem zweitbesten Alignment-Score. Daher lässt sich der MAPQ-Wert durch Anpassung der Parameter für das Alignment-Scoring beeinflussen. Diese Anpassungen werden unter *Einstellungen für das Alignment-Scoring nach Smith-Waterman* auf Seite 19 beschrieben.

Die Option *--mapq-strict-sjs* ist RNA-spezifisch und kommt zur Anwendung, wenn mindestens ein Exonsegment zuverlässig aligniert ist, jedoch Unsicherheiten bezüglich möglicher Spleißstellen bestehen. Ist diese Option auf 0 festgelegt, wird ein höherer MAPQ-Wert zurückgegeben, wodurch das Alignment als zumindest teilweise korrekt klassifiziert wird. Ist diese Option auf 1 festgelegt, wird ein niedrigerer MAPQ-Wert zurückgegeben, wodurch die Spleißstelle als mehrdeutig klassifiziert wird.

Bestimmte nachgeordnete Tools wie Cufflinks erwarten für alle eindeutig gemappten Reads einen eindeutigen MAPQ-Wert. Dieser Wert wird mit der Option *--rna-mapq-unique* angegeben. Das Festlegen dieser Option auf einen Wert ungleich null überschreibt alle MAPQ-Prognosen auf Grundlage des Alignment-Scores. Dadurch wird die MAPQ für alle eindeutig gemappten Reads auf den Wert von *--rna-mapq-unique* festgelegt. Sämtliche mehrfach gemappten Reads haben einen MAPQ-Wert von int(-10*log10(1 - 1/NH), wobei der NH-Wert die Anzahl der Treffer (primäre und sekundäre Alignments) für diesen Read angibt.

Erkennung von Genfusionen

Das Modul DRAGEN Gene Fusion verwendet den DRAGEN-RNA-Spliced-Aligner für die Erkennung von Genfusionsereignissen. Mithilfe einer Split-Read-Analyse der ergänzenden (chimärischen) Alignments werden mögliche Unterbrechungspunkte festgestellt. Die putativen Fusionsereignisse durchlaufen dann verschiedene Filterungsphasen, um mögliche falsch positive Ergebnisse zu minimieren. Zusätzlich zum Endergebnis werden alle potenziellen Kandidaten (ungefiltert) ausgegeben, wodurch die Sensitivität maximiert werden kann.

Ausführen von DRAGEN Gene Fusion

Das Modul DRAGEN Gene Fusion kann zusammen mit einem regulären RNA-Seq-Mapping-Alignment-Auftrag ausgeführt werden. Dieses zusätzliche Modul führt zu einer minimal längeren Laufzeit für die Verarbeitung und bietet zusätzliche Informationen zu Ihren RNA-Seq-Versuchen.

Legen Sie zum Aktivieren des Moduls DRAGEN Gene Fusion in Ihren aktuellen RNA-Seq-Befehlszeilenskripts die Option *--enable-rna-gene-fusion* auf „true“ fest. Das Modul DRAGEN Gene Fusion erfordert die Verwendung einer Gen-Annotationsdatei im Format GTF oder GFF.

Im Folgenden ist ein Beispiel für eine Befehlszeile zur Ausführung eines vollständigen RNA-Seq-Versuchs aufgeführt.

```
/opt/edico/bin/dragen \
   -r <HASHTABELLE> \
   -1 <FASTQ1> \
   -2 <FASTQ2> \
   -a <GTF-DATEI> \
   --output-dir <AUSGABEVERZEICHNIS> \
   --output-file-prefix <PRÄFIX> \
```

```
--RGID <READ-GRUPPEN-ID> \
--RGSM <PROBENNAME> \
--enable-rna true \
--enable-rna-gene-fusion true
```

Am Ende eines Laufs wird eine Zusammenfassung der ermittelten Genfusionsereignisse ähnlich wie in folgendem Beispiel ausgegeben.

```
================================================================
Loading gene annotations file
================================================================
  Input annotations file: ref_annot.gtf
  Number of genes: 27459
  Number of transcripts: 196520
  Number of exons: 1196293

================================================================
Launching DRAGEN Gene Fusion Detection
================================================================
annotation-file:            ref_annot.gtf
rna-gf-blast-pairs:         blast_pairs.outfmt6
rna-gf-exon-snap:           50
rna-gf-min-anchor:          25
rna-gf-min-neighbor-dist:   15
rna-gf-max-partners:        3
rna-gf-min-score-ratio:     0.15
rna-gf-min-support:         2
rna-gf-min-support-be:      10
rna-gf-restrict-genes       true

================================================================
Completed DRAGEN Gene Fusion Detection
================================================================
Chimeric alignments: 107923
Total fusion candidates: 38 (2116 before filters)

Time loading annotations:                              00:00:08.543
Time running gene fusion:                              00:00:18.470
Total runtime:                                         00:00:27.760
**********************************************************
DRAGEN finished normally
```

Eigenständiges Ausführen von Gene Fusion

Das Modul DRAGEN Gene Fusion kann mit der *.Chimeric.out.junction-Datei als Eingabedatei und einer Genannotationsdatei im Format GTF/GFF als eigenständiges Tool ausgeführt werden. Das eigenständige Ausführen des Gene Fusion-Moduls eignet sich insbesondere zum Testen unterschiedlicher Konfigurationsoptionen zur Erkennung von Genfusionen, ohne die RNA-Seq-Daten mehrfach mappen und alignieren zu müssen.

Geben Sie zur Verwendung des Gene Fusion-Moduls von DRAGEN als eigenständiges Tool mit der Option *--rna-gf-input-file* die bereits generierte *.Chimeric.out.junction-Datei an.

Es folgt eine Beispielbefehlszeile für die Ausführung des Gen Fusion-Moduls als eigenständiges Tool.

```
/opt/edico/bin/dragen \
   -a <GTF-DATEI> \
   --rna-gf-input-file <EINGABE-CHIMERIC> \
   --output-dir <AUSGABEVERZEICHNIS \
   --output-file-prefix <PRÄFIX> \
   --enable-rna true \
   --enable-rna-gene-fusion true
```

Die Ergebnisse des eigenständigen Modus unterscheiden sich von denen bei der Ausführung im Rahmen von Reads.

Genfusionskandidaten

Die erfassten Genfusionsereignisse werden in den fusion_candidate-Ausgabedateien aufgeführt. Im Ausgabeverzeichnis befinden sich zwei Dateien, *.fusion_candidates.preliminary und *.fusion_candidates.final. Bei der vorläufigen Datei (preliminary) handelt es sich um eine vorgefilterte Liste mit erfassten Kandidatenereignissen. Die endgültige Datei (final) enthält die Kandidatenereignisse mit ausreichend hoher Konfidenz nach Durchlaufen sämtlicher Filter. (Der Inhalt der Datei wird im Folgenden beschrieben.) Die Datei enthält die folgenden fünf Spalten:

- Fusionsgen
- Einen Score
- Die beiden Unterbrechungspunkte
- Die zugrundeliegenden Reads

Die Unterbrechungspunkte beziehen sich auf die Split-Read-Unterbrechungspunkte aus den Alignments. Das + bzw. - im Unterbrechungspunkt gibt an, von welchem Strang, bezogen auf die Referenz-FASTA, dieses Gen transkribiert wurde. Die Read-Namen sind durch Semikola getrennt. Die Datei ist nach Score sortiert. Dieser gibt die Anzahl der Reads mit diesem Fusionsereignis an.

```
#FusionGene         Score  LeftBreakpoint      RightBreakpoint      ReadNames
BSG--AL021546.6     108    chr19:580461:+      chr12:120876182:+    R_1;R_2;R_3;
…
FIS1--PMEL          70     chr7:100884111:-    chr12:56351868:-     …
CBX3--G3BP2         57     chr7:26242642:+     chr4:76572341:-      …
ELOVL5--SF3B14      46     chr6:53139888:-     chr2:24291329:-
ATXN10--GORASP2     44     chr22:46134719:+    chr2:171804860:+
FADS3--MTOR         39     chr11:61646784:-    chr1:11184690:-
AKR1B1--HMGB1       33     chr7:134135538:-    chr13:31036849:-
```

Optionen und Filter für Genfusionen

Durch die Implementierung verschiedener Filter wird die Anzahl der falsch positiven Genfusionskandidaten verringert. Folgende Schwellenwerte und Optionen sind konfigurierbar. Nach Anwendung der Filter werden alle Kandidaten, die weiterhin qualifiziert sind, in der Datei **.fusion_candidates.final* ausgegeben. In der Ausgabedatei **.fusion_candidates.preliminary* befinden sich alle vorgefilterten Fusionskandidaten.

- *--rna-gf-blast-pairs*

  Eine Datei mit Genpaaren, die eine hohe Ähnlichkeit aufweisen. Diese Liste mit Genpaaren wird als Homologiefilter verwendet, um die Anzahl falsch positiver Treffer zu verringern. In der Fusion Filter Wiki finden Sie Anweisungen zum Erstellen dieser Datei.

- *--rna-gf-restrict-genes*

  Bei der Analyse der Gen-Annotationsdatei (GTF/GFF) für das DRAGEN Gene Fusion-Modul können mithilfe dieser Option die Einträge von Interesse auf ausschließlich proteincodierende Regionen beschränkt werden. Durch die Beschränkung der GTF-Datei auf proteincodierende Gene reduziert sich die Quote falsch positiver Treffer in den derzeit beobachteten Fusionsereignissen. Die Standardeinstellung ist „true“.

Genexpressionsquantifizierung

Die DRAGEN-RNA-Pipeline umfasst ein Genexpressionsquantifizierungsmodul, das die Expression jedes Transkripts und Gens in einem RNA-Sequenzierungsdatensatz schätzt. Zunächst wird das Genom-Mapping eines jeden Reads (Read-Paar) intern in die entsprechenden Transkript-Mappings umgewandelt. Dann werden mithilfe eines Expectation-Maximization(EM)-Algorithmus die Transkript-Expressionswerte bestimmt, die mit allen beobachteten Reads am besten übereinstimmen. Der EM-Algorithmus kann auch die GC-Verzerrung modellieren und in den aufgeführten Quantifizierungsergebnissen korrigieren.

Ausführen der Quantifizierung

Legen Sie zum Aktivieren des Quantifizierungsmoduls in Ihren aktuellen RNA-Seq-Befehlszeilenskripts die Option *--enable-rna-quantification* auf „true“ fest. Für die Quantifizierung ist darüber hinaus eine Datei für die Gen-Annotation (GTF/GFF) erforderlich, die die Genomposition aller zu quantifizierenden Transkripts bereitstellt. Die entsprechende Festlegung erfolgt über die Option „-a“ (oder *--annotation-file*).

Ausgaben der Quantifizierung

Die Transkript-Quantifizierungsergebnisse werden in der Datei <Ausgabepräfix>.quant.sf aufgeführt. In dieser Textdatei werden Ergebnisse für jedes Transkript aufgeführt. Beispiel:

```
Name                Length  EffectiveLength     TPM             NumReads
ENST00000364415.1   116     12.3238             5.2328          1
ENST00000564138.1   2775    2105.58             1.28293         41.8885
```

- Name führt die transcriptID des Transkripts auf.
- Length ist die Länge des (gespleißten) Transkripts in Basenpaaren.
- EffectiveLength ist die für die RNA-Sequenzierung verfügbare Länge, wobei die Insertgröße und Randeffekte berücksichtigt werden.
- TPM steht für Transkripte pro Million und stellt die für Transkriptlänge und Sequenzierungstiefe normalisierte Expression des Transkripts dar.
- NumReads steht für die geschätzte Anzahl der Reads aus dem Transkript (nicht normalisiert).

Diese Datei kann mithilfe von Tools wie tximport und DESeq2 als Eingabe für die differentielle Genexpression verwendet werden.

Entsprechend enthält die Datei <Ausgabepräfix>.quant.genes.sf die Quantifizierungsergebnisse auf Genebene. Die Ergebnisse werden berechnet durch die Summierung aller Transkripte mit der gleichen geneID in der Annotation (GTF). Length und EffectiveLength sind die (nach Expression) gewichteten Mittelwerte der einzelnen Transkripte im Gen.

Quantifizierungsoptionen

- *--enable-rna-quantification*

  Ist diese Option auf „true“ festgelegt, wird die RNA-Quantifizierung aktiviert. Es ist erforderlich, dass *--enable-rna* ebenfalls auf „true“ festgelegt wird.

- *--rna-quantification-library-type*

  Gibt die Art der RNA-Seq-Bibliothek an.
  - IU: nicht strangspezifische Paired-End-Bibliothek.
  - ISR: strangspezifische Paired-End-Bibliothek, in der read2 mit dem Transkript-Strang übereinstimmt (z. B. TruSeq RNA).
  - ISF: strangspezifische Paired-End-Bibliothek, in der read1 mit dem Transkript-Strang übereinstimmt.
  - U: nicht strangspezifische Single-End-Bibliothek.
  - SR: strangspezifische Single-End-Bibliothek, in der Reads in umgekehrter Ausrichtung zum Transkript-Strang stehen (z. B. TruSeq RNA).
  - SF: strangspezifische Single-End-Bibliothek, in der Reads mit dem Transkript-Strang übereinstimmen.
  - A (automatische Erkennung, Standardwert): DRAGEN prüft für diesen Wert die ersten Reads/Paare im Datensatz, um automatisch den richtigen Bibliothekstyp zu ermitteln.

- *--rna-quantification-gc-bias*

  Die Korrektur der GC-Verzerrung schätzt den Effekt von „transcript %GC“ auf die Sequenzierungs-Coverage und berücksichtigt diesen bei der Schätzung der Expression. Bei Festlegung dieser Option auf „false“ wird die Korrektur der GC-Verzerrung deaktiviert.

- *--rna-quantification-fld-max*, *--rna-quantification-fld-mean*, *--rna-quantification-fld-sd*

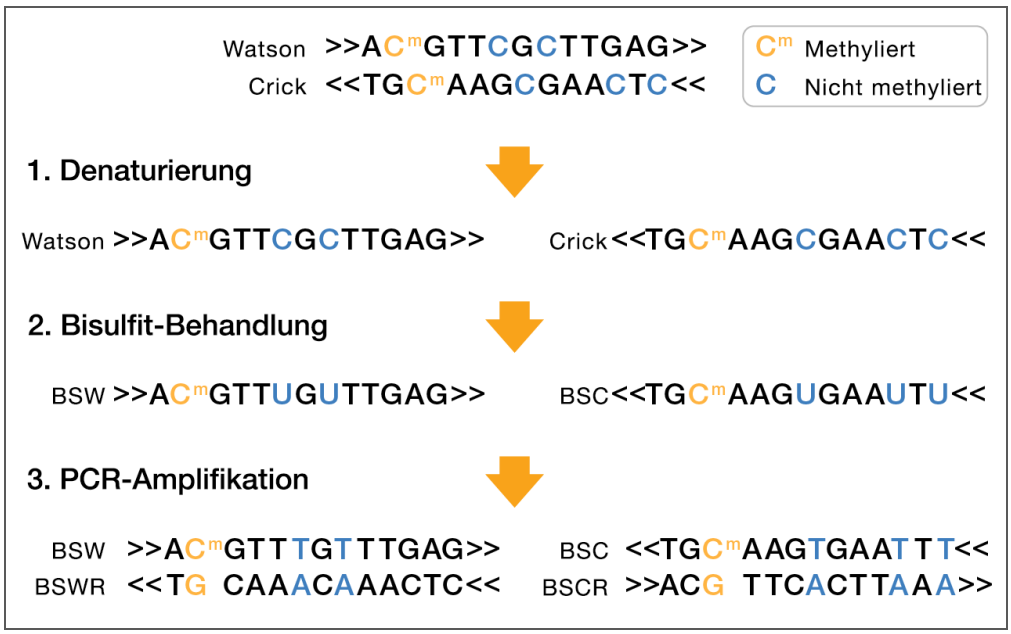
  Diese Optionen werden verwendet, um die Insertgrößenverteilung der RNA-Seq-Bibliothek für Single-End-Läufe anzugeben. Dies ist relevant für die Korrektur der GC-Verzerrung. Die Standardwerte sind 250 +- 25, max=1000. Eine Änderung dieser Werte entsprechend der jeweiligen Bibliothek kann die Genauigkeit erhöhen.

Kapitel 5 DRAGEN-Methylierungspipeline

Die epigenetische Methylierung von Cytosin-Basen in DNA kann erhebliche Auswirkungen auf die Genexpression haben. Für die Erkennung von Mustern epigenetischer Methylierung bei Einzelbasenauflösung ist Bisulfit-Sequenzierung ein äußerst zuverlässiges, allgemein anerkanntes Verfahren. Bei dieser Methode wird DNA mit Natriumbisulfit behandelt, um die unmethylierten Cytosin-Basen in Uracil umzuwandeln. Methylierte Cytosine hingegen werden nicht verändert. Anschließend wird das Uracil mit PCR-Amplifikation vollständig in Thymin umgewandelt.

Eine Bisulfit-Sequenzierungsbibliothek kann entweder nicht direktional oder direktional sein. Im nicht direktionalen Verfahren ergibt jedes doppelstrangige DNA-Fragment vier eindeutige Stränge für die Sequenzierung, nach Amplifikation, wie in folgendem Diagramm dargestellt:

Abbildung 11 Nicht direktionale Bisulfit-Sequenzierung



- Bisulfit-Watson (BSW), umgekehrtes Komplement von BSW (BSWR),
- Bisulfit-Crick (BSC), umgekehrtes Komplement von BSC (BSCR)

Für direktionale Bibliotheken werden die vier Strangtypen generiert. Adapter werden jedoch so an die DNA-Fragmente angebunden, dass nur die BSW- und BSC-Stränge sequenziert werden (Lister-Protokoll). Die BSWR- und BSCR-Stränge werden seltener für die Sequenzierung ausgewählt (direktionales Komplement-Protokoll).

BSW- und BSC-Stränge:

- A, G, T: unverändert
- Methyliertes C bleibt C
- Unmethyliertes C wird zu T

BSWR- und BSCR-Stränge:

- Zu den ursprünglichen Watson/Crick-A-, -G-, -T-Basen komplementäre Basen bleiben unverändert.
- Zum ursprünglichen methylierten Watson/Crick-C komplementäres G bleibt G.
- Zum ursprünglichen unmethylierten Watson/Crick-C komplementäres G wird zu A.

Sequenzierungs-Reads werden per Standard-DNA-Sequenzierung generiert. Die Behandlung mit Bisulfit wirkt sich auf Reads mit einer höheren Zahl nicht umgewandelter C-Basen oder zu nicht umgewandelten C-Basen komplementärer G-Basen weniger stark aus. Für diese Reads besteht außerdem eine höhere Mapping-Wahrscheinlichkeit als für Basen mit einer höheren Anzahl veränderter Basen. Laut Standardprotokoll wird diese Mapping-Verzerrung durch mehrere Alignments pro Read minimiert. Hierbei werden spezifische Kombinationen von Reads und Referenzgenombasen vor jedem Alignierungslauf *in-silico* konvertiert. Jeder Alignierungslauf verfügt über eine Reihe von Einschränkungen und Basenkonvertierungen, die einem der Stränge vom Typ Bisulfit+PCR entsprechen, die im Protokoll erwartet werden. Über einen Vergleich der Alignierungsergebnisse aus mehreren Läufen können Sie für jeden Read oder jedes Read-Paar die beste Alignierung und den Strangtyp mit der höchsten Wahrscheinlichkeit bestimmen. Diese Informationen sind für das nachfolgende Methylierungs-Calling erforderlich.

Methylierungs-Calling mit DRAGEN

Bei unterschiedlichen Methylierungsprotokollen ist die Generierung von zwei oder vier Alignments pro Eingabe-Read erforderlich. Im Anschluss erfolgt eine Analyse zur Auswahl eines besten Alignments und zur Bestimmung, welche Cytosine methyliert werden. DRAGEN kann diesen Prozess automatisieren. Es wird eine einzelne BAM-Ausgabedatei mit Bismark-kompatiblen Tags (XR, XG und XM) erstellt, die in nachgeordneten Pipelines wie Bismark-Methylierungsextraktionsskripts verwendet werden können.

Ist die Option *--methylation-protocol* auf einen gültigen Wert ungleich „none“ festgelegt, generiert DRAGEN automatisch den erforderlichen Satz an Alignments, jeweils mit entsprechenden Konvertierungen auf den Reads, Konvertierungen auf der Referenz und Einschränkungen, ob Reads vorwärts oder in Richtung des Gegenstrangs (umgekehrtes Komplement, RC) mit der Referenz aligniert werden. Ist die Option *--enable-methylation-calling* auf „true“ festgelegt, analysiert DRAGEN die verschiedenen Alignments, um eine einzelne BAM-Datei mit Methylierungs-Tags zu erstellen. Ist die Option *--enable-methylation-calling* auf „false“ festgelegt, gibt DRAGEN eine separate BAM-Datei pro Alignment-Lauf aus.

In der nachfolgenden Tabelle werden diese Alignment-Läufe beschrieben:

Protokoll	BAM	Referenz	Read 1	Read 2	Ausrichtungsbeschränkung
Direktional					
	1	C->T	C->T	G->A	nur vorwärts
	2	G->A	C->T	G->A	nur RC
Nicht direktional oder direktional-komplementär					
	1	C->T	C->T	G->A	nur vorwärts
	2	G->A	C->T	G->A	nur RC
	3	C->T	G->A	C->T	nur RC
	4	G->A	G->A	C->T	nur vorwärts

In direktionalen Protokollen ist die Bibliothek so vorbereitet, dass nur die BSW- und BSC-Stränge sequenziert werden. Daher werden Alignment-Läufe mit den zwei Kombinationen aus Basenkonvertierungen und Ausrichtungsbeschränkungen durchgeführt, die diesen Strängen entsprechen (direktionale Läufe 1 und 2 oben). In nicht direktionalen Protokollen liegen Reads aus jedem der vier Stränge gleichermaßen wahrscheinlich vor, sodass Alignment-Läufe mit zwei weiteren Kombinationen aus Basenkonvertierungen und Ausrichtungsbeschränkungen durchgeführt werden müssen (nicht direktionale Läufe 3 und 4 oben).

Das **direktional-komplementäre** Protokoll bezweckt das Gleiche wie das direktionale Protokoll, die Sequenzierungs-Reads stammen hier jedoch aus den RC-Strängen der BSW- und BSC-Stränge. Beim direktional-komplementären Protokoll werden nur sehr wenige gute Alignments aus den Läufen 1 und 2 erwartet, sodass DRAGEN für diese Läufe automatisch einen schnelleren Analysemodus festlegt.

Jedes Protokoll muss mit einer Referenz ausgeführt werden, die mit aktivierter Option *--ht-methylated* generiert wurde. Informationen hierzu finden Sie unter *Pipelinespezifische Hashtabellen* auf Seite 139.

Im Folgenden finden Sie ein Beispiel für eine DRAGEN-Befehlszeile für das direktionale Protokoll:

```
dragen --enable-methylation-calling true \
   --methylation-protocol directional \
   --ref-dir /staging/ref/mm10/methylation --RGID RG1 --RGCN CN1 \
   --RGLB LIB1 --RGPL illumina --RGPU 1 --RGSM Samp1 \
   --intermediate-results-dir /staging/tmp \
   -1 /staging/reads/samp1_1.fastq.gz \
   -2 /staging/reads/samp1_2.fastq.gz \
   --output-directory /staging/outdir \
   --output-file-prefix samp1_directional_prot
```

BAM-Tags in Zusammenhang mit der Methylierung

Wenn *--enable-methylation-calling* auf „true“ festgelegt ist, analysiert DRAGEN automatisch die für das konfigurierte *--methylation-protocol* erstellten Alignments und generiert eine einzelne BAM-Ausgabedatei, die die Tags in Zusammenhang mit der Methylierung für alle gemappten Reads enthält. Wie in Bismark werden Reads ohne eindeutiges bestes Alignment aus der Ausgabe-BAM ausgeschlossen. Folgende Tags werden hinzugefügt:

Tag	Kurzbeschreibung	Beschreibung
XR:Z	Read-Konvertierung	Die Basenkonvertierung, die für das beste Alignment im Read durchgeführt wurde: CT oder GA.
XG:Z	Referenzkonvertierung	Die Basenkonvertierung, die für das beste Alignment in der Referenz durchgeführt wurde: CT oder GA.
XM:Z	Methylierungs-Call	Eine Zeichenfolge für die Methylierung mit einem Byte pro Base.

Das Tag XM:Z (Methylierungs-Call) enthält ein Byte für jede Base in der Read-Sequenz. Jede Position ohne Cytosin wird mit einem Punkt („.“) gekennzeichnet, Positionen mit Cytosin durch einen Buchstaben. Der Buchstabe gibt den Kontext an (CpG, CHG, CHH oder unbekannt). Die Groß-/Kleinschreibung gibt die Methylierung an. Positionen mit Großschreibung sind methyliert. Positionen mit Kleinschreibung sind nicht methyliert. Folgende Buchstaben werden für Cytosin-Positionen verwendet:

Buchstabe	Methyliert?	Kontext
.	kein Cytosin	kein Cytosin
z	Nein	CpG
Z	Ja	CpG
X	Nein	CHG
X	Ja	CHG
h	Nein	CHH

Buchstabe	Methyliert?	Kontext
H	Ja	CHH
u	Nein	Unbekannt
U	Ja	Unbekannt

Berichte zur Cytosin-Methylierung und M-Verzerrung

Legen Sie die Option *--methylation-generate-cytosine-report* auf „true“ fest, wenn Sie einen Bericht zur genomweiten Cytosin-Methylierung generieren möchten. Position und Strang von jedem C im Genom stehen in den ersten drei Feldern des Berichts. Ein Datensatz mit einem „-“ im Strangfeld steht für ein G in der FASTA-Referenzdatei. Die Anzahl von methylierten und unmethylierten Cs, die die Position abdecken, stehen in den Feldern vier bzw. fünf. Der C-Kontext in der Referenz (CG, CHG oder CHH) steht im sechsten Feld und der Trinukleotid-Sequenzkontext im letzten Feld (z. B. CCC, CGT, CGA usw.). Im Folgenden ist ein Beispiel für einen Cytosin-Datensatz aufgeführt:

```
chr2    24442367    +    18    0    CG    CGC
```

Legen Sie die Option *--methylation-generate-mbias-report* auf „true“ fest, wenn Sie einen Bericht zur M-Verzerrung generieren möchten. Dieser Bericht umfasst drei Tabellen für Single-End-Daten mit einer Tabelle für jeden C-Kontext und sechs Tabellen für Paired-End-Daten. Jede Tabelle stellt eine Serie von Datensätzen mit einem Datensatz pro Read-Basen-Position dar. Beispielsweise enthält der erste Datensatz für die CHG-Tabelle die Zählung der methylierten Cs (Feld 2) und der unmethylierten Cs (Feld 3) auf der ersten Read-Basen-Position mit der Einschränkung auf die Reads, bei denen die erste Base mit einer CHG-Position im Genom aligniert ist. Jeder Tabellendatensatz enthält außerdem den Prozentsatz der methylierten C-Basen (Feld 4) und die Summe der methylierten und unmethylierten C-Zählungen (Feld 5).

Im Folgenden ist ein Beispiel für einen Datensatz zur M-Verzerrung für die Read-Basen-Position 10 aufgeführt:

```
10    7335    2356    75.69    9691
```

Wenn sich Paired-End-Reads in einem Datensatz überlappen, werden in den Berichten zu Cytosin und zur M-Verzerrung alle Cs im zweiten Read, der mit dem ersten Read überlappt, übersprungen. Darüber hinaus werden 1-Basen-Koordinaten für Positionen in beiden Berichten verwendet.

Legen Sie die Option *--methylation-match-bismark* auf „true“ fest, um die von Bismark Version 0.19.0 generierten Berichte zu bismark_methylation_extractor-Cytosin und zur M-Verzerrung aufeinander abzustimmen. Die Reihenfolge von Datensätzen in Bismark- und DRAGEN-Cytosin-Berichten kann abweichen. DRAGEN-Berichte werden nach der genomischen Position sortiert.

Verwenden von Bismark für das Methylierungs-Calling

Das empfohlene Vorgehen beim Methylierungs-Calling besteht darin, DRAGEN die mehreren erforderlichen Alignments automatisch durchführen und die XM-, XR- und XG-Tags wie bereits beschrieben hinzufügen zu lassen. Sie können *--enable-methylation-calling* jedoch auf „false“ festlegen, damit DRAGEN eine separate BAM-Datei für alle Einschränkungen und Konvertierungen erstellt, die sich aus dem Methylierungsprotokoll ergeben. Sie können diese BAM-Datei für das Methylierungs-Calling mithilfe eines Drittanbietertools verwenden. Beispielsweise lässt sich Bismark so modifizieren, dass Reads auf diesen BAM-Dateien verarbeitet werden, statt Bismark Bowtie oder Bowtie2 intern ausführen zu lassen. Wenden Sie sich an den technischen Support von Illumina, wenn Sie Hilfe bei diesem Verfahren benötigen.

Bei der Ausführung in diesem Modus erstellt ein einzelner DRAGEN-Lauf mehrere BAM-Dateien im mit *--output-directory* angegebenen Ausgabeverzeichnis. Diese enthalten die Alignments in Reihenfolge der Eingabe-Reads. Bei diesen Läufen kann keine Sortierung oder Dublettenkennzeichnung verwendet werden. Alignments enthalten MD-Tags. Zusätzlich wird für die Kompatibilität mit Bismark „/1“ oder „/2“ an die Namen von Paired-End-Reads angefügt. Diese BAM-Dateien verwenden die folgenden Namenskonventionen:

- Single-End-Reads: *Ausgabeverzeichnis/Ausgabedateipräfix.*{CT,GA}read{CT,GA}reference.bam
- Paired-End-Reads: *Ausgabeverzeichnis/Ausgabedateipräfix.*{CT,GA}read1{CT,GA}read2 {CT,GA}reference.bam,

Ausgabeverzeichnis und *Ausgabedateipräfix* werden mit den entsprechenden Optionen (output-directory und output-file-prefix) angegeben. CT und GA entsprechen den in der obigen Tabelle aufgeführten Basenkonvertierungen.

Bismark verfügt nicht über einen Richtungs-Komplement-Modus. Sie können solche Proben jedoch im nicht richtungsgebundenen Modus von Bismark verarbeiten, wobei die Läufe 1 und 2 nur sehr wenige korrekte Alignments erwarten lassen. Aus diesem Grund reduziert DRAGEN bei der Ausführung eines nicht richtungsgebundenen Protokolls automatisch die bei diesen Läufen für Alignments eingesetzten Ressourcen.

Kapitel 6 Vorbereiten eines Referenzgenoms

Vor der Verwendung eines Referenzgenoms mit DRAGEN muss dieses vom FASTA-Format in ein spezielles Binärformat für die DRAGEN-Hardware konvertiert werden. Die in diesem Vorbereitungsschritt verwendeten Optionen ermöglichen eine Abstimmung von Leistung und Mapping-Qualität.

Das DRAGEN-System wird mit den Referenzgenomen hg19 und GRCh37 ausgeliefert. Beide Referenzgenome sind anhand von empfohlenen Einstellungen für allgemeine Anwendungen vorinstalliert. Wenn Sie mit der Leistung und der Mapping-Qualität zufrieden sind, können Sie wahrscheinlich einfach mit diesen im Lieferumfang enthaltenen Referenzgenomen arbeiten. Je nach Read-Längen oder spezifischen Aspekten der Anwendung lassen sich die Mapping-Qualität und/oder die Leistung durch Verändern der Referenzvorbereitungsoptionen verbessern.

Hashtabellenhintergrund

Der DRAGEN-Mapper extrahiert viele überlappende Seeds (Teilsequenzen oder K-mere) aus jedem Read und sucht nach diesen Seeds in einer Hashtabelle im Arbeitsspeicher der PCIe-Karte, um Positionen im Referenzgenom mit einer Seed-Übereinstimmung zu identifizieren. Hashtabellen sind ideal für äußerst schnelle Suchabfragen nach exakten Übereinstimmungen geeignet. Die DRAGEN-Hashtabelle muss mithilfe der Option *dragen --build-hash-table* aus einem gewählten Referenzgenom erstellt werden, wobei viele überlappende Seeds aus dem Referenzgenom extrahiert und in Datensätze der Hashtabelle eingetragen werden. Die Hashtabelle wird als Binärdatei gespeichert.

Referenz-Seed-Intervall

Die Größe der DRAGEN-Hashtabelle ist proportional zur Anzahl der Seeds, die mit Daten aus dem Referenzgenom ausgefüllt werden. In der Standardeinstellung erhält ein Seed für jede Position im Referenzgenom Daten. Das sind bei einem Humangenom ca. 3 Milliarden Seeds. Diese Standardeinstellung erfordert mindestens 32 GB Arbeitsspeicher auf dem DRAGEN-PCIe-Board.

Für größere nicht humane Genome oder zur Minimierung der Hashtabelle kann mithilfe der Option *--ht-ref-seed-interval* ein durchschnittliches Referenzintervall angegeben werden, sodass nicht alle Referenz-Seeds ausgefüllt werden.Das Standardintervall für das vollständige Ausfüllen lautet *--ht-ref-seed-interval 1*. Das Intervall für das Ausfüllen von 50 % der Seeds wird mit *--ht-ref-seed-interval 2* angegeben. Beim Ausfüllintervall muss es sich nicht um eine Ganzzahl handeln. Beispielsweise gibt *--ht-ref-seed-interval 1.2* eine Ausfüllung von 83,3 % an, wobei mit hauptsächlich 1- und einigen 2-Basen-Intervallen ein Basenintervalldurchschnitt von 1,2 erreicht wird.

Hashtabellenbelegung

Einer Hashtabelle wird in der Regel eine bestimmte Größe zugeordnet und sie enthält stets einige leere Datensätze, sodass die Belegung unter 100 % liegt. Eine ausreichende Menge an Leerstellen ist auch für einen schnellen Zugriff auf die DRAGEN-Hashtabelle wichtig. Als guter Richtwert gilt eine Belegung von ca. 90 %. Leerstellen sind wichtig, da die Datensätze pseudozufällig in der Hashtabelle platziert werden, wodurch an einigen Stellen eine abnorm hohe Anzahl an Datensätzen vorliegt. Diese überfüllten Regionen können recht groß werden, wenn der Anteil der Leerstellen gegen null geht. Dies führt dazu, dass sich Abfragen vom DRAGEN-Mapper für einige Seeds zunehmend verlangsamen.

Hashtabelle/Seed-Länge

Die Hashtabelle wird mit Referenz-Seeds einer einzelnen gängigen Länge ausgefüllt. Diese primäre Seed-Länge wird mit der Option *--ht-seed-len* geregelt. Der Standardwert ist 21.

Bei einer Tabellengröße von 8 GB bis 31,5 GB werden primäre Seeds von bis zu 27 Basen unterstützt. Im Allgemeinen verbessern längere Seeds die Laufzeitleistung und kürzere Seeds die Mapping-Qualität (Erfolgsrate und Genauigkeit). Ein längerer Seed ist im Referenzgenom mit höherer Wahrscheinlichkeit eindeutig, wodurch ein schnelles Mapping ohne die Prüfung vieler alternativer Positionen möglich ist. Ein längerer Seed überlappt jedoch auch mit höherer Wahrscheinlichkeit mit einer Abweichung von der Referenz (Varianten- oder Sequenzierungsfehler), wodurch ein erfolgreiches Mapping durch eine exakte Übereinstimmung dieses Seeds verhindert wird (selbst wenn ein anderer Seed aus dem Read möglicherweise weiterhin gemappt werden kann). In jedem Read sind weniger lange Seed-Positionen verfügbar.

Längere Seeds sind besser für längere Reads geeignet, da mehr Seed-Positionen verfügbar sind, um Abweichungen zu vermeiden.

Tabelle 9 Empfehlungen für die Seed-Länge

Wert für *-ht-seed-len*	Read-Länge
21	100 bp bis 150 bp
17 bis 19	kürzere Reads (36 bp)
27	mehr als 250 bp

Hashtabelle/Seed-Extensionen

Aufgrund repetitiver Sequenzen stimmen einige Seeds beliebiger Länge mit vielen Positionen im Referenzgenom überein. Mithilfe der einzigartigen Methode der Seed-Extension kann DRAGEN solche Seeds mit hoher Häufigkeit mappen. Wenn die Software feststellt, dass ein primärer Seed an zahlreichen Referenzpositionen auftritt, wird dieser Seed um einige Basen an beiden Enden auf eine größere, für die Referenz eindeutigere Länge erweitert.

Ein primärer Seed mit 21 Basen kann beispielsweise an jedem Ende um 7 Basen erweitert werden, sodass daraus ein erweiterter Seed mit 35 Basen resultiert. Ein primärer Seed mit 21 Basen kann mit 100 Positionen in der Referenz übereinstimmen. Extensionen mit 35 Basen dieser 100 Seed-Positionen können jedoch in 40 Gruppen mit 1 bis 3 identischen Seeds mit 35 Basen unterteilt werden. Iterative Seed-Extensionen werden ebenfalls unterstützt. Sie werden automatisch erstellt, wenn ein großer Satz identischer primärer Seeds verschiedene Teilsätze enthält, die am besten mithilfe verschiedener Extensionslängen aufgelöst werden.

Die maximale erweiterte Seed-Länge entspricht in der Standardeinstellung der Länge des primären Seeds plus 128. Mithilfe der Option *--ht-max-ext-seed-len* kann dieser Wert angepasst werden. Für kurze Reads wird beispielsweise empfohlen, die maximale Länge für Seed-Extensionen auf einen Wert unterhalb der Read-Länge festzulegen, da es für Extensionen mit einer größeren Länge als der des Gesamt-Reads keine Übereinstimmungen geben kann.

Mithilfe der folgenden Optionen können Sie auch festlegen, wie streng Seeds erweitert werden (für fortgeschrittene Benutzer):

- *--ht-cost-coeff-seed-len*
- *--ht-cost-coeff-seed-freq*
- *--ht-cost-penalty*
- *--ht-cost-penalty-incr*

Extensionslänge und Trefferhäufigkeit beeinflussen sich gegenseitig. Schnelleres Mapping kann durch längere Seed-Extensionen erzielt werden, da so die Seed-Trefferhäufigkeit verringert wird. Präziseres Mapping kann durch das Vermeiden von Seed-Extensionen oder durch kurze Extensionen unter Inkaufnahme höherer Trefferhäufigkeiten erzielt werden. Die Qualität des Mappings kann durch kürzere Extensionen verbessert werden, indem sich Seeds besser in SNPs einpassen lassen und indem mehr potenzielle Mapping-Positionen

ermittelt werden, an denen Alignierungen erzielt werden können. Mit relativ kurzen Seed-Extensionen und hohen Trefferhäufigkeiten begünstigen die Standardeinstellungen für Extensionen und Seed-Frequenz deutlich ein präzises Mapping.

Für die Seed-Frequenz sind folgende Standardwerte festgelegt:

Option	Standardwert
--ht-cost-coeff-seed-len	1
--ht-cost-coeff-seed-freq	0.5
--ht-cost-penalty	0
--ht-cost-penalty-incr	0.7
--ht-max-seed-freq	16
--ht-target-seed-freq	4

Seed-Frequenz – Limit und Target

Ein primärer oder erweiterter Seed kann mit mehreren Stellen im Referenzgenom übereinstimmen. Alle derartigen Übereinstimmungen werden in die Hashtabelle eingefügt und abgerufen, wenn der DRAGEN-Mapper nach einem übereinstimmenden Seed aus einer Read-Extraktion sucht. Die verschiedenen Referenzpositionen werden dann bei der Erstellung einer alignierten Mapper-Ausgabe berücksichtigt und verglichen. Die Option *dragen* erzwingt jedoch ein Limit bezüglich der Anzahl an Übereinstimmungen bzw. der Häufigkeit von Seeds. Die Steuerung erfolgt über die Option *--ht-max-seed-freq*. Das Häufigkeitslimit ist standardmäßig auf 16 festgelegt. Wenn die Software auf einen Seed mit einer höheren Häufigkeit trifft, wird dieser auf einen ausreichend langen sekundären Seed erweitert, sodass die Häufigkeit eines beliebigen erweiterten Seed-Musters in dieses Limit fällt. Wenn jedoch bei einer maximalen Seed-Extension das Limit immer noch überschritten wird, wird der Seed zurückgewiesen und nicht in die Hashtabelle aufgenommen. Stattdessen füllt *dragen* einen einzelnen High Frequency-Datensatz aus.

Das Limit für die Seed-Frequenz beeinträchtigt die DRAGEN-Mappingqualität in der Regel nicht wesentlich. Dies liegt an den folgenden beiden Gründen. Erstens werden Seeds nur zurückgewiesen, wenn die Extension fehlschlägt. Nur primäre Seeds mit äußerst hoher Häufigkeit – in der Regel mit Tausenden von Übereinstimmungen – werden zurückgewiesen. Solche Seeds sind für das Mapping nicht sehr hilfreich. Zweitens müssen auch andere Seed-Positionen in einem gegebenen Read geprüft werden. Wenn eine andere Seed-Position eindeutig genug ist, dass mindestens eine Übereinstimmung zurückgegeben wird, kann der Read noch richtig gemappt werden. Wenn jedoch alle Seed-Positionen aufgrund hoher Häufigkeit zurückgewiesen werden, bedeutet dies oft, dass der gesamte Read gleichermaßen gut in vielen Referenzpositionen übereinstimmt. Ein Mapping des Reads wäre demnach eine willkürliche Wahl mit sehr niedriger MAPQ oder einer MAPQ von null.

Daher funktioniert das Häufigkeitsstandardlimit von 16 für *--ht-max-seed-freq* gut. Es kann jedoch verringert oder bis auf einen Maximalwert von 256 erhöht werden. Bei einem höheren Häufigkeitslimit wird die Anzahl der gemappten Reads geringfügig erhöht (insbesondere bei kurzen Reads), in der Regel weisen die zusätzlich gemappten Reads jedoch eine sehr niedrige MAPQ oder eine MAPQ von null auf. Das DRAGEN-Mapping wird verlangsamt, da eine entsprechend große Anzahl an möglichen Mappings in Betracht gezogen werden muss.

Neben dem Häufigkeitslimit kann mit der Option *--ht-target-seed-freq* eine *target*-Seed-Häufigkeit festgelegt werden. Diese Target-Häufigkeit wird verwendet, wenn die Extensionen für primäre Seeds mit hoher Häufigkeit generiert werden. Die Extensionslängen werden mit einer Präferenz für die Häufigkeiten erweiterter Seeds in der Nähe des Targets ausgewählt. Der Standardwert 4 für *--ht-target-seed-freq* bedeutet, dass die Software eher kürzere Seed-Extensionen generiert als für das Mapping von eindeutigen Seeds erforderlich.

Handhabung von Decoy-Contigs

Das Verhalten von DRAGEN hinsichtlich der Handhabung von Decoy-Contigs in der Referenz wurde nach Version 2.6 geändert.

Seit Version DRAGEN 3.x erkennt der Hashtabellen-Builder automatisch die Abwesenheit der Decoy-Contigs in der Referenz und fügt sie vor dem Erstellen der Hashtabelle der FASTA-Datei hinzu. Die Decoys-Datei befindet sich unter /opt/edico/liftover/hs_decoys.fa. Wenn die Decoy-Contigs nicht in der Referenz vorhanden sind, werden die auf die Decoy-Contigs gemappten Reads in der ausgegebenen BAM künstlich als nicht gemappt markiert (da das Decoy-Contig in der ursprünglichen Referenz nicht vorhanden ist). Daraus ergibt sich eine künstlich verringerte Mapping-Rate. Allerdings wird die Genauigkeit der Varianten-Callings verbessert, da falsch positive Calls durch Decoy-Reads entfernt werden.

Illumina empfiehlt, diese Funktion als Standardeinstellung zu verwenden. Sie können allerdings die Option *--htsuppress-decoys* auf „true“ festlegen, wenn Sie diese Decoys nicht zur Hashtabelle hinzufügen möchten.

In der folgenden Tabelle wird das unterschiedliche Verhalten älterer DRAGEN-Versionen (bis 2.6) im Vergleich zu den Versionen DRAGEN 3.x hinsichtlich der Handhabung von Decoy-Contigs im Hashtabellen-Builder erläutert:

Verhalten von DRAGEN	DRAGEN 2.6 und ältere Versionen	DRAGEN 3.x
Decoy-Contigs (z. B. GRCh37) nicht in Referenz enthalten	Decoy-Reads werden aufgrund der fehlenden Contigs in der Referenz fälschlicherweise anderen Regionen des Genoms zugewiesen. • Künstlich erhöhte Mapping-Rate. • Falsch positive Calls in Regionen mit Rauschen, zu denen die Decoy-Contigs falsch zugewiesen werden.	DRAGEN erkennt automatisch das Fehlen des Decoy-Contigs in der Referenz und fügt dieses der FASTA-Datei hinzu. • Künstlich niedrigere Mapping-Rate, da die auf die Decoy-Contigs gemappten Decoy-Reads in der ausgegebenen BAM künstlich als nicht gemappt markiert werden (da das Decoy-Contig nicht in der ursprünglichen Referenz vorhanden ist). • Falsch positive Calls werden dank des automatischen Hinzufügens der Decoy-Contigs vermieden und das Varianten-Calling wird so unterstützt.
Decoy-Contigs (z. B. hs37d5) in Referenz enthalten	Mapping der Decoy-Reads zu den Decoy-Contigs. • Hohe Mapping-Rate • Keine falsch positiven Calls durch Decoy-Reads dank des korrekten Mappings der Decoy-Reads	Mapping der Decoy-Reads zum Decoy-Contig. • Hohe Mapping-Rate • Keine falsch positiven Calls durch Decoy-Reads dank des korrekten Mappings der Decoy-Reads

ALT-sensible Hashtabellen

Erstellen Sie zur Aktivierung des ALT-sensiblen Mappings in DRAGEN GRch38 (und andere Referenzen mit ALT-Contigs) mit einer Liftover-Datei. Verwenden Sie hierzu die Option *--ht-alt-liftover*. Der Hashtabellen-Builder stuft sämtliche Referenzsequenzen anhand der Liftover-Datei als primär oder alternativ ein und speichert die Primärdaten vor den Alternativdaten in reference.bin. Die SAM-Liftover-Dateien für hg38DH und hg19 befinden sich im Ordner /opt/edico/liftover. Die Option *--ht-alt-liftover* gibt den Pfad zur Liftover-Datei für die Erstellung einer ALT-sensiblen Hashtabelle an.

Sie können die Erforderlichkeit einer Liftover-Datei überschreiben, indem Sie die Option *--ht-alt-aware-validate* beim Erstellen von Hashtabellen und beim Ausführen von DRAGEN auf „false“ festlegen.

Benutzerdefinierte Liftover-Dateien

Benutzerdefinierte Liftover-Dateien können anstelle der mit DRAGEN bereitgestellten verwendet werden. Liftover-Dateien müssen das SAM-Format aufweisen. Eine SAM-Kopfzeile ist jedoch nicht erforderlich. Die Felder SEQ und QUAL können ausgelassen werden („*“). Jeder Alignment-Datensatz muss als QNAME eine alternative Haplotyp-Referenzsequenzbezeichnung aufweisen, die RNAME und POS des Liftover-Alignments in einer Zielreferenzsequenz (in der Regel die primäre Assembly) angibt.

Alignments in Richtung des Gegenstrangs (umgekehrtes Komplement) sind in FLAG mit dem Bit 0x10 gekennzeichnet. Als nicht gemappt (0x4) oder sekundär (0x100) gekennzeichnete Datensätze werden ignoriert. Der CIGAR kann Hard oder Soft Clipping enthalten, wodurch das ALT-Contig nicht vollständig aligniert wird.

Eine einzelne Referenzsequenz kann nicht gleichzeitig als ALT-Contig (in QNAME enthalten) und als Liftover-Ziel (in RNAME enthalten) verwendet werden. Derselben Position der primären Assembly können mehrere ALT-Contigs aligniert werden. Außerdem können mehrere Alignments für ein einziges ALT-Contig bereitgestellt werden (wahlweise lassen sich Zusätze mit 0x800 als ergänzend kennzeichnen), beispielsweise um das Alignment eines Teils vorwärts und eines anderen Teils in Richtung des Gegenstrangs durchzuführen. Jedoch erhält jede Base des ALT-Contigs nur ein Liftover-Image gemäß dem ersten Alignment-Datensatz mit einer M CIGAR-Operation, die diese Base abdeckt.

SAM-Datensätze ohne QNAME im Referenzgenom werden ignoriert, sodass dieselbe Liftover-Datei für unterschiedliche Referenzuntergruppen verwendet werden kann. Es tritt jedoch ein Fehler auf, wenn bei einem Alignment der entsprechende QNAME vorhanden ist und der RNAME fehlt.

Befehlszeilenoptionen

Mithilfe der Option *--build-hash-table* können Sie eine FASTA-Referenzdatei in die Hashtabelle für das DRAGEN-Mapping übertragen. Als Eingabe sind eine FASTA-Datei (mehrere Referenzsequenzen werden verkettet) und ein bereits vorhandenes Ausgabeverzeichnis erforderlich. Folgende Dateien werden generiert:

reference.bin	Die in 4 Bits pro Base codierten Referenzsequenzen. Vier-Bit-Codes werden verwendet, damit die Größe in Bytes ungefähr der Hälfte der Referenzgenomgröße entspricht. Zwischen Referenzsequenzen werden N gekürzt und das Padding erfolgt automatisch. Beispielsweise verfügt hg19 über 3.137.161.264 Basen in 93 Sequenzen. Die Codierung ist 1.526.285.312 Byte = 1,46 GB, wobei 1 GB gleichbedeutend mit 1 GiB oder 2^{30} Byte ist.
hash_table.cmp	Komprimierte Hashtabelle. Die Hashtabelle wird dekomprimiert und vom DRAGEN-Mapper für die Suche nach primären Seeds mit einer über die Option *--ht-seed-len* definierten Länge sowie nach erweiterten Seeds von variabler Länge verwendet.
hash_table.cfg	Liste von Parametern und Attributen für die generierte Hashtabelle in einem Textformat. Diese Datei enthält wichtige Informationen über das Referenzgenom und die Hashtabelle.
hash_table.cfg.bin	Binärversion von **hash_table.cfg** zur Konfiguration der DRAGEN-Hardware.
hash_table_stats.txt	Textdatei mit umfangreichen internen Statistiken zur erstellten Hashtabelle, einschließlich der Prozentwerte für die Hashtabellenbelegung. Diese Tabelle wird zu Informationszwecken zur Verfügung gestellt. Sie wird nicht von anderen Tools verwendet.

Die Befehlszeilensyntax wird wie folgt erstellt:

```
dragen --build-hash-table true [Optionen] --ht-reference <Referenz.fasta>
   --output-directory <Ausgabeverzeichnis>
```

Die folgenden Abschnitte enthalten Informationen über die Optionen zur Erstellung einer Hashtabelle.

Eingabe-/Ausgabeoptionen

Die Optionen *--ht-reference* und *--output-directory* sind für die Erstellung einer Hashtabelle erforderlich. Die Option *--ht-reference* gibt den Pfad zur Referenz-FASTA-Datei an, während *--output-directory* ein bereits vorhandenes Verzeichnis angibt, in dem die Ausgabedateien mit den Hashtabellen gespeichert werden. Illumina empfiehlt, unterschiedliche Hashtabellen-Builds in unterschiedlichen Ordnern zu speichern. Als Best Practice sollten die Ordnernamen alle Einstellungen für Nichtstandardparameter enthalten, mit denen die enthaltene Hashtabelle erstellt wurde.

Primäre Seed-Länge

Die Option *--ht-seed-len* gibt die ursprüngliche Länge in Nukleotiden von Seeds aus dem Referenzgenom an, die in die Hashtabelle eingetragen wird. Der Mapper extrahiert während der Laufzeit Seeds identischer Länge aus jedem Read und sucht nach exakten Übereinstimmungen in der Hashtabelle (sofern Seed-Editing deaktiviert ist).

Die maximale primäre Seed-Länge ist abhängig von der Hashtabellengröße. Der Grenzwert ist k=27 für Tabellengrößen von 16 GB bis 64 GB (übliche Größen für das gesamte Humangenom) oder k=26 für Größen von 4 GB bis 16 GB.

Die minimale primäre Seed-Länge ist abhängig von der Größe und Komplexität des Referenzgenoms. Sie muss lang genug sein, um die meisten Referenzpositionen eindeutig zu bestimmen. Bei Referenzen für das gesamte Humangenom ist die Hashtabellenerstellung in der Regel bei k < 16 fehlerhaft. Die Untergrenze kann für kürzere Genome niedriger bzw. für weniger komplexe (repetitivere) Genome höher sein. Der Schwellenwert für die Eindeutigkeit von *--ht-seed-len 16* für das Humangenom mit 3,1 Gbp ist intuitiv verständlich, da log4(3.1 G) ≈ 16. Es sind mindestens 16 Auswahlmöglichkeiten aus 4 Nukleotiden erforderlich, um 3,1-G-Referenzpositionen zu unterscheiden.

Hinweise hinsichtlich der Genauigkeit

Für ein erfolgreiches Read-Mapping muss mindestens ein primärer Seed genau übereinstimmen (bzw. mit einem einzelnen SNP übereinstimmen, wenn bearbeitete Seeds verwendet werden). Kürzere Seeds können mit höherer Wahrscheinlichkeit erfolgreich auf die Referenz gemappt werden, da diese seltener mit Varianten oder Sequenzierungsfehlern überlappen und mehr von ihnen in einen Read passen. Hinsichtlich der Mapping-Genauigkeit sind kürzere Seeds in der Regel besser.

Jedoch können besonders kurze Seeds die Mapping-Genauigkeit mitunter beeinträchtigen. Besonders kurze Seeds lassen sich häufig auf mehrere Referenzpositionen mappen und führen dazu, dass der Mapper mehr falsche Mapping-Positionen berücksichtigt. Aufgrund der unzureichenden Abbildung von Mutationen und Fehlern mit dem Smith-Waterman-Alignment-Scoring und anderen Heuristiken gelangen diese Fehlzuordnungen möglicherweise in die Berichte. Mit Laufzeitqualitätsfiltern wie *--Aligner.aln_min_score* lassen sich die Probleme hinsichtlich der Genauigkeit bei besonders kurzen Seeds in den Griff bekommen.

Hinweise hinsichtlich der Geschwindigkeit

Kürzere Seeds verlangsamen in der Regel das Mapping, da sie auf mehr Referenzpositionen gemappt werden. Zur Ermittlung des besten Ergebnisses sind daher zusätzliche Arbeitsschritte wie beispielsweise Smith-Waterman-Alignments erforderlich. Dieser Effekt ist am stärksten ausgeprägt, wenn sich die Länge des primären Seeds an den Schwellenwert für die Eindeutigkeit des Referenzgenoms (z. B. K=16 für das Humangesamtgenom) annähert.

Überlegungen zur Anwendung

- **Read-Länge**: In der Regel sind kürzere Seeds für kürzere Reads und längere Seeds für längere Reads geeignet. Innerhalb eines kurzen Reads können einige Positionen mit Nichtübereinstimmungen (Varianten oder Sequenzierungsfehler) den Read in kurze, mit der Referenz übereinstimmende Segmente teilen. In diesem Fall können nur kurze Seeds zwischen den Abweichungen vorhanden sein und der Referenz exakt entsprechen. Bei einem Read mit 36 bp kann beispielsweise ein einziger SNP in der Mitte Übereinstimmungen von Seeds mit einer Länge von über 18 bp mit der Referenz verhindern. Im Gegensatz dazu sind in einem Read von 250 bp Länge 15 SNPs erforderlich, um Seeds mit einer Länge von 27 bp mit einer Wahrscheinlichkeit von über 0,01 % zu verhindern.
- **Paired-Ends**: Paired-End-Reads können bei längeren Reads eine hohe Mapping-Genauigkeit ermöglichen. DRAGEN verbessert die Mapping-Genauigkeit mithilfe von Paired-End-Informationen, z. B. mit Rescue-Scans, die den erwarteten Referenzbereich durchsuchen, wenn nur ein Mate über Seeds mit Mappings auf eine vorgegebene Referenzregion verfügt. Daher verdoppeln Paired-End-Reads die Wahrscheinlichkeit, dass für einen Seed mit exakter Übereinstimmung die korrekte Alignierung ermittelt wird.
- **Varianten- oder Fehlerrate**: Wenn häufiger Unterschiede zwischen Reads und der Referenz auftreten, müssen sich kürzere Seeds möglicherweise zwischen die Positionen mit den Unterschieden in einem bestimmten Read einpassen lassen und mit der Referenz exakt übereinstimmen.
- **Anforderung hinsichtlich des Mapping-Prozentsatzes**: Kurze Seeds können hilfreich sein, wenn für die Anwendung ein hoher Prozentsatz an Reads auf bestimmte Positionen gemappt werden muss (auch bei geringer MAPQ). Einige Reads, die mit der Referenz nicht übereinstimmen, lassen sich mit höherer Wahrscheinlichkeit mappen, wenn mithilfe kurzer Seeds teilweise Übereinstimmungen mit der Referenz ermittelt werden.

Maximale Seed-Länge

Mithilfe der Option *--ht-max-ext-seed-len* wird die Länge der erweiterten Seeds beschränkt, die in die Hashtabelle eingetragen werden. Primäre Seeds (Länge durch *--ht-seed-len* festgelegt), die mit vielen Referenzpositionen übereinstimmen, können erweitert werden, um mehr eindeutige Übereinstimmungen zu erzielen. Dies kann erforderlich sein, um Seeds innerhalb der maximalen Trefferhäufigkeit (*--ht-max-seed-freq*) zu mappen.

Bei einer primären Seed-Länge k kann die maximale Seed-Länge zwischen k und k+128 festgelegt werden. Der Standardwert ist die Obergrenze von k+128.

Begrenzung der Seed-Extension

Die Option *--ht-max-ext-seed-len* eignet sich für kurze Reads, z. B. für Reads unter 50 bp. In solchen Fällen ist es hilfreich, die Seed-Extension auf die Read-Länge abzüglich einer kleinen Differenz wie z. B. 1–4 bp zu begrenzen. Beispiel: Bei einem Read mit 36 bp ist für die Option *--ht-max-ext-seed-len* ein Wert von 35 geeignet. Dadurch wird sichergestellt, dass der Hashtabellen-Builder keine Seed-Extension plant, die länger als der Read ist und die bei Seeds, die mit einer kürzeren Extension in den Read passen würden, zu einem Laufzeitfehler bei Seed-Extension und -Mapping führt.

Die Seed-Extension kann für längere Reads auf ähnliche Weise begrenzt werden, z. B. durch Festlegen von *--ht-max-ext-seed-len* auf 99 für Reads mit 100 bp. Dies ist jedoch wenig hilfreich, da Seeds stets konservativ erweitert werden. Selbst bei einem Standardgrenzwert von k+128 werden einzelne Seeds nur auf die Länge erweitert, die erforderlich ist, um die maximale Trefferhäufigkeit (*--ht-max-seed-freq*) zu unterbieten. Allenfalls ist die Erweiterung einige Basen länger, um die Target-Trefferhäufigkeit (*--ht-target-seed-freq*) zu erreichen oder zu viele inkrementelle Erweiterungsschritte zu vermeiden.

Maximale Trefferhäufigkeit

Die Option *--ht-max-seed-freq* legt eine feste Obergrenze für die Anzahl der Seed-Treffer fest (Referenzgenompositionen), die für einen beliebigen primären oder erweiterten Seed ausgefüllt werden können. Wenn ein gegebener primärer Seed mehr als den durch diese Obergrenze festgelegten Referenzpositionen zugeordnet werden kann, muss er so lange erweitert werden, bis die erweiterten Seeds in kleinere Gruppen identischer Seeds aufgeteilt werden können, die innerhalb dieses Limits bleiben. Wenn eine Gruppe identischer Referenz-Seeds selbst bei der maximal erweiterten Seed-Länge (*--ht-max-ext-seed-len*) über dieser Obergrenze liegt, werden die Referenzpositionen nicht in die Hashtabelle eingetragen. Stattdessen füllt *dragen* einen einzelnen High Frequency-Datensatz aus.

Die maximale Trefferhäufigkeit kann auf einen Wert zwischen 1 und 256 konfiguriert werden. Ist dieser Wert jedoch zu niedrig, kann die Hashtabellenerstellung fehlschlagen, da zu viele Seed-Extensionen erforderlich sind. Das sinnvolle Minimum für eine Gesamthumangenomreferenz ist 8, sofern alle anderen Optionen auf den Standardwert festgelegt sind.

Hinweise hinsichtlich der Genauigkeit

Im Allgemeinen erhöht eine größere maximale Trefferhäufigkeit die Wahrscheinlichkeit für ein erfolgreiches Mapping. Dies hat zwei Ursachen. Erstens werden bei einem höheren Limit weniger Referenzpositionen zurückgewiesen, die nicht gemappt werden können. Zweitens ermöglicht ein höheres Limit kürzere Seed-Extensionen, was die Wahrscheinlichkeit für eine exakte Seed-Übereinstimmung ohne überlappende Varianten oder Sequenzierungsfehler erhöht.

Jedoch kann das Zulassen einer hohen Trefferanzahl (wie bei besonders kurzen Seeds auch) die Mapping-Genauigkeit beeinträchtigen. Die meisten Seed-Treffer in einer großen Gruppe befinden sich nicht genau am Mapping-Ort. Gelegentlich kann es aufgrund mangelhafter Scoring-Modelle vorkommen, dass einer dieser Fehltreffer in den Bericht aufgenommen wird. Außerdem ist die Gesamtzahl der Referenzpositionen begrenzt, die der Mapper verarbeitet. Das Zulassen extrem hoher Trefferzahlen führt möglicherweise dazu, dass die beste Übereinstimmung aufgrund einer Verdrängung nicht verarbeitet wird.

Hinweise hinsichtlich der Geschwindigkeit

Größere maximale Trefferhäufigkeiten verlangsamen das Read-Mapping, da Seed-Mappings mehr Referenzpositionen finden. Zur Ermittlung des besten Ergebnisses sind daher zusätzliche Arbeitsschritte wie beispielsweise Smith-Waterman-Alignments erforderlich.

Optionen für ALT-sensible Liftover-Dateien

Weitere Informationen zum Erstellen einer benutzerdefinierten Liftover-Datei finden Sie unter *ALT-sensible Hashtabellen* auf Seite 132.

- *--ht-alt-liftover*

  Die Option *--ht-alt-liftover* gibt den Pfad zur Liftover-Datei für die Erstellung einer ALT-sensiblen Hashtabelle an. Diese Option ist für die Erstellung anhand einer Referenz mit ALT-Contigs erforderlich. SAM-Liftover-Dateien für hg38DH und hg19 befinden sich unter /opt/edico/liftover.

- *--ht-alt-aware-validate*

  Für das Generieren einer Hashtabelle aus einer Referenz mit ALT-Contigs ist eine Liftover-Datei erforderlich. Wenn Sie die Option *--ht-alt-aware-validate* auf „false“ festlegen, wird diese Anforderung außer Kraft gesetzt.

- *--ht-decoys*

  Die DRAGEN-Software erkennt automatisch die Verwendung von hg19- und hg38-Referenzen. Wenn

diese nicht in der FASTA-Tabelle gefunden werden, fügt die Software der Hashtabelle Decoys hinzu. Mithilfe der Option *--ht-decoys* geben Sie den Pfad der Decoys-Datei an. Der Standardpfad lautet /opt/edico/liftover/hs_decoys.fa.

- *--ht-suppress-decoys*

  Mit der Option *--ht-suppress-decoys* setzen Sie die Verwendung der Decoys-Datei beim Erstellen der Hashtabelle außer Kraft.

Optionen der DRAGEN-Software

- *--ht-num-threads*

  Die Option *--ht-num-threads* bestimmt die maximale Anzahl der Worker-CPU-Threads, die eingesetzt werden, um die Generierung von Hashtabellen zu beschleunigen. Der Standardwert für diese Option ist 8. Maximal sind 32 Threads zulässig.

  Wenn Ihr Server die Ausführung von mehr Threads unterstützt, wird empfohlen, das Maximum zu verwenden. Beispielsweise verfügen die DRAGEN-Server über 24 Kerne mit aktiviertem Hyper-Threading, daher sollte ein Wert von 32 verwendet werden. Bei Verwendung eines höheren Wertes muss *--ht-max-table-chunks* ebenfalls angepasst werden. Die Server verfügen über 128 GB Arbeitsspeicher.

- *--ht-max-table-chunks*

  Die Option *--ht-max-table-chunks* steuert während der Generierung von Hashtabellen die Arbeitsspeicherauslastung, indem die Anzahl der Hashtabellen-Abschnitte mit ca. 1 GB, die sich gleichzeitig im Arbeitsspeicher befinden dürfen, begrenzt wird. Jeder weitere Abschnitt verbraucht während der Generierung etwa das Doppelte seiner Größe (ca. 2 GB) an Systemarbeitsspeicher.

  Die Hashtabelle wird in unabhängige Zweierpotenz-Abschnitte einer festen Abschnittsgröße X geteilt, die sich abhängig von der Größe der Hashtabelle im Bereich 0,5 GB < X ≤ 1 GB bewegt. Beispielweise enthält eine 24 GB große Hashtabelle 32 unabhängige Abschnitte von 0,75 GB Größe, die durch parallele Threads mit genügend Arbeitsspeicher generiert werden können, während eine 16 GB große Hashtabelle 16 unabhängige Abschnitte von 1 GB Größe enthält.

  Die Standardeinstellung ist *--ht-max-table-chunks* gleich *--ht-num-threads*, jedoch mit einem Mindeststandardwert für *--ht-max-table-chunks* von 8. Eine Übereinstimmung dieser beiden Optionen ist sinnvoll, da für die Generierung eines Abschnitts der Hashtabelle ein ebenso großer Abschnitt Arbeitsspeicher sowie ein Thread zur Verarbeitung erforderlich sind. Dennoch bietet ein Anheben von *--ht-max-table-chunks* auf einen höheren Wert als *--ht-num-threads* oder von *--ht-num-threads* auf einen höheren Wert als *--ht-max-table-chunks* Vorteile bezüglich der Generierungsgeschwindigkeit.

Größenoptionen

- *--ht-mem-limit*: Arbeitsspeicherbegrenzung

  Die Option *--ht-mem-limit* legt die Größe der generierten Hashtabelle fest, indem der auf dem DRAGEN-Board verfügbare Arbeitsspeicher für die Hashtabelle und das codierte Referenzgenom angegeben werden. Die Option *--ht-mem-limit* wird auf den Standardwert von 32 GB festgelegt, wenn das Referenzgenom WHG-Größe erreicht, bzw. auf eine mehr als ausreichende Größe bei kleineren Referenzen. Normalerweise gibt es keinen Grund, diese Standardeinstellungen zu überschreiben.

- *--ht-size*: Größe der Hashtabelle

  Diese Option gibt die Größe der zu generierenden Hashtabelle an und wird anstelle der Berechnung einer passenden Tabellengröße anhand der Größe des Referenzgenoms und des verfügbaren Arbeitsspeichers (Option *--ht-mem-limit*) verwendet. Es wird empfohlen, die Standardtabellengröße zu verwenden. Die nächstbeste Wahl ist die Verwendung von *--ht-mem-limit*.

Optionen für das Ausfüllen von Seeds

- *--ht-ref-seed-interval*: Seed-Intervall

  Die Option*--ht-ref-seed-interval* legt die Schrittgröße zwischen Seed-Positionen im Referenzgenom fest, die in die Hashtabelle eingefügt werden. Ein Intervall von 1 (Standardwert) bedeutet, dass jede Seed-Position ausgefüllt wird, 2 bedeutet, dass 50 % der Positionen ausgefüllt werden usw. Nachkommastellen werden unterstützt, beispielsweise gibt der Wert 2.5 an, dass 40 % ausgefüllt werden.

  Mit 32 GB Arbeitsspeicher auf DRAGEN-Boards lassen sich die Seeds eines Humanreferenzgenoms problemlos zu 100 % ausfüllen. Ändern Sie diese Option, wenn ein wesentlich größeres Referenzgenom verwendet wird.

- *--ht-soft-seed-freq-cap* und *--ht-max-dec-factor*: Variable Häufigkeitsgrenze und maximaler Minderungsfaktor für das Seed Thinning

  Beim Seed Thinning handelt es sich um ein experimentelles Verfahren zur Verbesserung der Mapping-Leistung in Regionen mit hoher Häufigkeit. Wenn die Häufigkeit primärer Seeds über der mit der Option *--ht-soft-seed-freq-cap* festgelegten Grenze liegt, werden nur so viele Seed-Positionen ausgefüllt, wie ohne Überschreitung des Grenzwerts möglich. Die Option *--ht-max-dec-factor* gibt den maximalen Faktor an, mit dem die Seeds ausgedünnt werden können. Beispielsweise bleiben bei *--ht-max-dec-factor 3* mindestens 1/3 der ursprünglichen Seeds erhalten. *--ht-max-dec-factor 1* deaktiviert die Ausdünnung vollständig.

  Die Seeds werden nach spezifischen Mustern ausgedünnt, die lange unausgefüllte Abschnitte verhindern. Mit dem Seed Thinning soll sich eine gemappte Seed-Coverage in Referenzregionen mit hoher Frequenz erzielen lassen, wo andernfalls die maximale Trefferhäufigkeit überschritten würde. Außerdem kann mit dem Seed Thinning die Seed-Extension begrenzt werden, was ein erfolgreiches Mapping unterstützt. Bislang vorliegende Tests zeigen keine Überlegenheit von Seed Thinning im Vergleich zu anderen Verfahren zur Optimierung der Genauigkeit.

- *--ht-rand-hit-hifreq* und *--ht-rand-hit-extend*: Zufallsprobentreffer mit HIFREQ- und EXTEND-Datensatz

  Immer wenn ein HIFREQ- oder EXTEND-Datensatz in die Hashtabelle eingefügt wird, steht dieser stellvertretend für einen großen Satz an Referenztreffern in einem bestimmten Seed. Wahlweise kann der Hashtabellen-Builder auch zufällig einen Vertreter dieses Satzes auswählen und diesen HIT-Datensatz zusätzlich zum HIFREQ- oder EXTEND-Datensatz einfügen.

  Zufallsprobentreffer bieten alternative Alignments, die besonders nützlich zur MAPQ-Bestimmung für die gemeldeten Alignments sind. Diese werden ausschließlich innerhalb des vorliegenden Kontexts zur Meldung von Alignment-Positionen verwendet, da dies andernfalls eine verzerrte Coverage der Loci zur Folge hätte, die während der Generierung der Hashtabelle ausgewählt wurden.

  Legen Sie *--ht-rand-hit-hifreq* auf 1 fest, um einen Probentreffer aufzunehmen. Die Option *--ht-rand-hit-extend* gibt eine minimale Trefferanzahl (vor Extension) für einen Probentreffer an. Null deaktiviert die Option. Es wird empfohlen, diese Optionen *nicht* zu ändern.

Steuerung von Seed-Extensionen

Die dynamische Seed-Extension von DRAGEN wird bei Bedarf bei bestimmten k-meren angewendet, die auf zu viele Referenzpositionen gemappt werden können. Seeds werden inkrementell in Schritten von 2–14 Basen (stets geradzahlig) von einer primären Seed-Länge zu einer vollständig erweiterten Länge erweitert. Die Basen werden bei jedem Extensionsschritt symmetrisch angehängt. Dadurch wird ggf. auch das nächste Extensionsinkrement festgelegt.

Jedem primären Seed mit hoher Häufigkeit ist eine potenziell komplexe Seed-Extensionsstruktur zugeordnet. Jede vollständige Struktur wird während der Hashtabellen-Erstellung generiert und während des Seed-Mappings wird ein Pfad des Stamms durch iterative Extensionsschritte verfolgt. Der Hashtabellen-Builder durchsucht mithilfe eines dynamischen Programmieralgorithmus den Bereich aller möglichen Seed-Extensionsstrukturen nach einer optimalen Struktur. Dabei kommt eine Kostenfunktion zum Einsatz, die für eine ausgewogene Mapping-Geschwindigkeit und -Genauigkeit sorgt. Diese Kostenfunktion wird durch folgende Optionen definiert:

- *--ht-target-seed-freq*: Target-Trefferhäufigkeit

  Die Option *--ht-target-seed-freq* definiert die ideale Trefferanzahl pro Seed, auf die die Seed-Extension abzielen soll. Höhere Werte führen zu weniger und kürzeren endgültigen Seed-Extensionen, da kürzere Seeds in der Regel mit mehreren Referenzpositionen übereinstimmen.

- *--ht-cost-coeff-seed-len*: Kostenkoeffizient für Seed-Länge

  Die Option *--ht-cost-coeff-seed-len* weist die Kostenkomponente für jede Base zu, um die ein Seed erweitert wird. Zusätzliche Basen werden als Kosten berücksichtigt, da bei längeren Seeds das Risiko von überlappenden Varianten oder Sequenzierungsfehlern sowie dem Verlust der richtigen Mappings besteht. Höhere Werte führen zu kürzeren endgültigen Seed-Extensionen.

- *--ht-cost-coeff-seed-freq*: Kostenkoeffizient für Trefferhäufigkeit

  Die Option *--ht-cost-coeff-seed-freq* weist die Kostenkomponente für die Differenz zwischen der Target-Trefferhäufigkeit und der für einen einzelnen Seed ausgefüllten Trefferanzahl zu. Höhere Werte führen in erster Linie dazu, dass Seeds mit hoher Häufigkeit weiter erweitert werden, um deren Häufigkeiten dem Target anzupassen.

- *--ht-cost-penalty*: Kostenauswirkung für die Seed-Extension

  Die Option *--ht-cost-penalty* weist eine Kostenpauschale für die Extension über die primäre Seed-Länge hinweg zu. Ein höherer Wert führt dazu, dass insgesamt weniger Seeds erweitert werden. Aktuelle Tests zeigen, dass null (0) ein geeigneter Wert für diesen Parameter ist.

- *--ht-cost-penalty-incr*: Kosteninkrement für Extensionsschritt

  Die Option *--ht-cost-penalty-incr* weist wiederkehrende Kosten für jeden Schritt der inkrementellen Seed-Extension von der primären bis zur endgültigen erweiterten Seed-Länge zu. Mehr Schritte werden als höhere Kosten betrachtet, da eine Extension in vielen kleinen Schritten mehr Raum in der Hashtabelle für vorläufige EXTEND-Datensätze erfordert und zum Ausführen der Extensionen erheblich mehr Laufzeit erforderlich ist. Ein höherer Wert führt zu Seed-Extensionsstrukturen mit weniger Knoten, wobei die erweiterten Blatt-Seed-Längen in weniger und größeren Schritten von der primären Stamm-Seed-Länge erreicht werden können.

Pipelinespezifische Hashtabellen

Beim Generieren einer Hashtabelle konfiguriert DRAGEN standardmäßig die Optionen für die DNA-Seq-Verarbeitung. Zum Ausführen von RNA-Seq-Daten müssen Sie mit der Option *--ht-build-rna-hashtable true* eine RNA-Seq-Hashtabelle erstellen. Verweisen Sie bei einem RNA-Seq-Alignment-Lauf auf das ursprüngliche *--output-directory*-Verzeichnis und nicht auf das automatisch generierte Unterverzeichnis.

Bei der Erstellung der Hashtabelle für die CNV-Pipeline muss *--enable-cnv* auf „true“ festgelegt werden. Es wird eine zusätzliche k-mer-Hashmap erstellt, die im CNV-Algorithmus verwendet wird. Illumina empfiehlt, stets die Option *--enable-cnv* zu verwenden, falls Sie das CNV-Calling mit derselben Hashtabelle durchführen möchten, die auch für Mapping und Alignment verwendet wird.

Für DRAGEN-Methylierungsläufe ist die Generierung eines besonderen Hashtabellenpaars erforderlich, in dem die Referenzbasen von C->T in einer Tabelle und von G->A in der anderen Tabelle konvertiert werden. Wenn Sie die Hashtabellenerstellung mit der Option *--ht-methylated* ausführen, werden diese Konvertierungen automatisch vorgenommen. Die konvertierten Hashtabellen werden im mit *--output-directory* angegebenen Zielverzeichnis in zwei Unterverzeichnissen erstellt. Die Unterverzeichnisse werden den automatischen Basenkonvertierungen entsprechend mit „CT_converted“ und „GA_converted“ bezeichnet. Verweisen Sie bei der Verwendung dieser Hashtabellen für methylierte Alignment-Läufe auf das ursprüngliche *--output-directory*-Verzeichnis und nicht auf eines der automatisch generierten Unterverzeichnisse.

Diese Basenkonvertierungen entfernen einen erheblichen Teil der Informationen aus den Hashtabellen. Daher kann es erforderlich sein, die Hashtabellenparameter anders als bei einer herkömmlichen Hashtabellenerstellung festzulegen. Folgende Optionen werden für die Erstellung von Hashtabellen für Säugetierspezies empfohlen:

```
dragen --build-hash-table=true --output-directory $REFDIR \
   --ht-reference $FASTA --ht-max-seed-freq 16 \
   --ht-seed-len 27 --ht-num-threads 40 --ht-methylated=true
```

Kapitel 7 Tools und Dienstprogramme

Konvertieren von Illumina-BCL-Daten

Beim BCL-Format handelt es sich um das native Ausgabeformat von Illumina-Sequenziersystemen. Es besteht aus einem Verzeichnis mit zahlreichen Daten- und Metadatendateien. Die Daten werden anhand des Fließzellen-Layouts des Sequenzierers organisiert, wodurch die Konvertierung dieser Daten in probenspezifische FASTQ-Dateien eine komplexe und mitunter zeitaufwendige Aufgabe darstellt.

DRAGEN umfasst eine schnelle Implementierung der Konvertierungssoftware, die die Hardwarebeschleunigung der DRAGEN-Plattform nutzt. Sie können diese Konvertierung mit den Optionen *--bcl-input-directory <BCL-ROOT>*, *--output-directory <VERZEICHNIS>* und *--bcl-conversion-only true* ausführen.

Die Implementierung der DRAGEN-BCL-Konvertierung gibt FASTQ-Dateien aus, die der bcl2fastq2 v2.20-Ausgabe von Illumina entsprechen. DRAGEN unterstützt die meisten Funktionen von bcl2fastq2, einschließlich der Demultiplexierung von Proben nach Barcode mit optionaler Fehlzuordnungstoleranz, Adaptersequenzmaskierung oder Kürzung mit anpassbarer Matching-Stringenz sowie Tagging und Kürzung von UMI-Sequenzen. Die Probenblatteinstellungen FindAdapterWithIndels, CreateFastqForIndexReads und ReverseComplement werden von DRAGEN nicht unterstützt. DRAGEN unterstützt nicht die Befehlszeilenoption no-lane-splitting.

Befehlszeilenoptionen

Der folgende Beispielbefehl enthält die erforderlichen Optionen für die BCL-Konvertierung in DRAGEN:

```
dragen --bcl-conversion-only --bcl-input-directory <…> --output-directory
   <…>
```

Die folgenden zusätzlichen Optionen können in der Befehlszeile angegeben werden:

- *--sample-sheet*: Gibt den Pfad zur Datei SampleSheet.csv an. *--sample-sheet* ist optional, wenn sich die Datei SampleSheet.csv im Verzeichnis *--bcl-input-directory* befindet.
- *--strict-mode*: Wird diese Option auf „true“ festgelegt, führt *dragen* einen Abbruch durch, wenn Dateien fehlen. Die Standardeinstellung ist „false“.
- *--first-tile-only*: Wird diese Option auf „true“ festgelegt, konvertiert *dragen* nur die erste Eingabedatei (zum Testen und Debuggen). Die Standardeinstellung ist „false“.
- *--bcl-only-lane <Nr.>*: Konvertiert in diesem Konvertierungslauf nur die angegebene Lane.
- *-f*: Konvertierung ins Ausgabeverzeichnis auch, wenn dieses bereits eine Konvertierung enthält (erzwingen).
- *--bcl-use-hw false*: Während der BCL-Konvertierung keine DRAGEN-FPGA-Beschleunigung verwenden.

Das BCL-Eingabestammverzeichnis und das Ausgabeverzeichnis müssen angegeben werden. Der angegebene Eingabepfad ist nicht das BaseCalls-Verzeichnis, sondern liegt drei Ebenen höher und muss u. a. folgende Dateien und Verzeichnisse enthalten:

```
Config\
Data\
Logs\
runParameters.xml
RunInfo.xml
```

Das Ausgabeverzeichnis für die FASTQ-Dateien wird mit der Option *--output-dir* angegeben.

Probenblattoptionen

Diese Optionen bestehen zusätzlich zu den Befehlszeilenoptionen, die das Verhalten der BCL-Konvertierung steuern. Für das Probenblatt können im Abschnitt [Settings] der Konfigurationsdatei Einstellungen zur Verarbeitung der Proben festgelegt werden. Die Probenblatteinstellungen für die BCL-Konvertierung lauten wie folgt:

Option	Standardwert	Wert	Beschreibung
AdapterBehavior	trim	trim, mask	Legt fest, ob der Adapter gekürzt oder maskiert werden soll.
AdapterRead1	Keine	Read 1-Adaptersequenz mit A, C, G oder T	Die zu kürzende oder maskierende Sequenz vom Ende von Read 1.
AdapterRead2	Keine	Read 2-Adaptersequenz mit A, C, G oder T	Die zu kürzende oder maskierende Sequenz vom Ende von Read 2.
AdapterStringency	0.9	Gleitkommazahl zwischen 0.5 und 1.0	Die Stringenz für eine Übereinstimmung von Read und Adapter mithilfe des Sliding-Window-Algorithmus.
BarcodeMismatchesIndex1	1	0, 1 oder 2	Die Anzahl der zulässigen Nichtübereinstimmungen zwischen dem ersten Index-Read und der Indexsequenz.
BarcodeMismatchesIndex2	1	0, 1 oder 2	Die Anzahl der zulässigen Nichtübereinstimmungen zwischen dem zweiten Index-Read und der Indexsequenz.
MinimumTrimmedReadLength	Der kleinere Wert von 35 und der kürzesten, nicht indizierten Read-Länge.	0 bis zur kürzesten, nicht indizierten Read-Länge	Unter diesen Wert gekürzte Reads werden maskiert.
MaskShortReads	Mindestens 22 sowie MinimumTrimmedReadLength.	0 bis MinimumTrimmedReadLength	Unter diesen Wert gekürzte Reads werden komplett ausgeblendet.
OverrideCycles	Keine	Y: Gibt einen Sequenzierungs-Read an I: Gibt einen Index-Read an U: Gibt die vom Read zu kürzende UMI-Länge an	Zeichenfolge zum Festlegen von UMI-Zyklen und Ausblenden von Zyklen eines Reads.

Die OverrideCycles-Maskierungselemente sind durch Semikola getrennt. Beispiel:

```
OverrideCycles,N1Y150;I8;I7N1;Y141U10
```

DRAGEN bietet ab sofort eine flexible UMI-Verarbeitung während der BCL-Konvertierung zur Unterstützung weiterer Drittanbieter-Assays, einschließlich UMI-Sequenzen in Index-Reads sowie mehrere UMI-Regionen pro Read. UMI-Sequenzen werden aus FASTQ-Read-Sequenzen gekürzt und wie gewöhnlich im Sequenzbezeichner für jeden Read platziert.

Ausgabe von BCL-Metriken

Bei der BCL-Konvertierung von DRAGEN werden Metriken im Ausgabeunterordner „Reports/“ ausgegeben. Die Angaben umfassen Demultiplexing-Metriken, Index-Hopping-Metriken (nur für eindeutige Doppel-Indizes) sowie die wichtigsten unbekannten Barcodes für jede Lane.

Überwachen des Systemzustands

Beim Einschalten des DRAGEN-Systems wird ein Daemon (*dragen_mond*) gestartet, der die Karte im Hinblick auf Hardwareprobleme überwacht. Dieser Daemon wird auch bei der Installation oder Aktualisierung des DRAGEN-Systems gestartet. Der Hauptzweck des Daemons ist die Überwachung der Temperatur des DRAGEN Bio-IT-Prozessors. Wenn die Temperatur einen festgelegten Schwellenwert überschreitet, führt dies zum Abbruch von DRAGEN.

Führen Sie den folgenden Befehl als Root aus, wenn Sie die Überwachung manuell starten, stoppen oder erneut starten möchten:

```
sudo service dragen_mond [stop|start|restart]
```

In der Standardeinstellung überprüft die Überwachung das System jede Minute auf Hardwareprobleme und protokolliert stündlich die Temperatur.

Die Datei /etc/sysconfig/dragen_mond gibt die Befehlszeilenoptionen zum Starten von *dragen_mond* an, wenn der Dienstbefehl ausgeführt wird. Wenn Sie die Standardoptionen ändern möchten, bearbeiten Sie in dieser Datei DRAGEN_MOND_OPTS. Der folgende Befehl bewirkt beispielsweise, dass die Abfragezeit auf 30 Sekunden und die Protokollierungszeit auf alle 2 Stunden geändert wird:

```
DRAGEN_MOND_OPTS="-d -p 30 -l 7200"
```

Die Option *-d* ist erforderlich, um die Überwachung als Daemon auszuführen.

Im Folgenden sind die Befehlszeilenoptionen für *dragen_mond* aufgeführt:

Option	Beschreibung
-m --swmaxtemp <n>	Maximale Temperatur für Softwarealarm (in °C). Der Standardwert ist 85.
-i --swmintemp <n>	Minimale Temperatur für Softwarealarm (in °C). Der Standardwert ist 75.
-H --hwmaxtemp <n>	Maximale Temperatur für Hardwarealarm (in °C). Der Standardwert ist 100.
-p --polltime <n>	Abfrageintervall für das Chipstatusregister (in Sekunden). Der Standardwert ist 60.
-l --logtime <n>	Protokollierung der FPGA-Temperatur alle n Sekunden. Der Standardwert ist 3600. Es muss sich um ein Vielfaches der Abfragezeit handeln.
-d --daemon	Lösen und Ausführung als Daemon.
-h --help	Ausgabe der Hilfe und Beendigung.
-V --version	Ausgabe der Version und Beendigung.

Mit dem Befehl *dragen_info -t* können Sie die aktuelle Temperatur des DRAGEN Bio-IT-Prozessors anzeigen. Dieser Befehl wird nur ausgeführt, wenn auch *dragen_mond* ausgeführt wird.

```
% dragen_info -t
FPGA Temperature: 42C  (Max Temp: 49C, Min Temp: 39C)
```

Protokollierung

Alle Hardwareereignisse werden unter /var/log/messages und /var/log/dragen_mond.log protokolliert. Im Folgenden finden Sie ein Beispiel für einen unter /var/log/messages protokollierten Temperaturalarm:

```
 Jul 16 12:02:34 komodo dragen_mond[26956]: WARNING: FPGA software over temperature alarm has been
triggered -- temp threshold: 85 (Chip status: 0x80000001)
 Jul 16 12:02:34 komodo dragen_mond[26956]: Current FPGA temp: 86, Max temp: 88, Min temp: 48
 Jul 16 12:02:34 komodo dragen_mond[26956]: All dragen processes will be stopped until alarm clears
 Jul 16 12:02:34 komodo dragen_mond[26956]: Terminating dragen in process 1510 with SIGUSR2 signal
```

In der Standardeinstellung wird die Temperatur stündlich in /var/log/dragen_mond.log protokolliert:

```
 Aug 01 09:16:50 Setting FPGA hardware max temperature threshold to 100
 Aug 01 09:16:50 Setting FPGA software max temperature threshold to 85
 Aug 01 09:16:50 Setting FPGA software min temperature threshold to 75
 Aug 01 09:16:50 FPGA temperatures will be logged every 3600 seconds
 Aug 01 09:16:50 Current FPGA temperature is 52 (Max temp = 52, Min temp = 52)
 Aug 01 10:16:50 Current FPGA temperature is 53 (Max temp = 56, Min temp = 49)
 Aug 01 11:16:50 Current FPGA temperature is 54 (Max temp = 56, Min temp = 49)
```

Bei Ausführung von DRAGEN nach Feststellen eines Temperaturalarms wird im Terminalfenster für den DRAGEN-Vorgang Folgendes angezeigt:

```
**********************************************************
**  Received external signal -- aborting dragen.        **
**  An issue has been detected with the dragen card.    **
**  Check /var/log/messages for details.                **
**                                                      **
**  It may take up to a minute to complete shutdown.    **
**********************************************************
```

Beenden Sie die DRAGEN-Software sofort, wenn diese Meldung angezeigt wird. Wirken Sie mit folgenden Schritten der Überhitzung der Karte entgegen:

- Stellen Sie eine ausreichende Belüftung der Karte sicher. Verwenden Sie ggf. einen besser belüfteten Kartensteckplatz, fügen Sie einen weiteren Lüfter hinzu oder erhöhen Sie die Leistung des Lüfters.
- Sorgen Sie für einen freien Bereich um die Karte. Wenn Sie über entsprechende PCIe-Steckplätze verfügen, wählen Sie einen Steckplatz, an dem die Karte möglichst frei positioniert ist.

Wenn Sie den Temperaturalarm nicht beheben können, wenden Sie sich an den technischen Support von Illumina.

Hardware-Alarme

In der folgenden Tabelle sind die Hardwareereignisse aufgeführt, die bei Auslösen eines Alarms von der Überwachung protokolliert werden:

ID	Beschreibung	Aktion der Überwachung
0	Überhitzung – Software	Verwendung wird beendet, bis der DRAGEN Bio-IT-Prozessor auf die in der Software zulässige Temperatur abgekühlt ist.
1	Überhitzung – Hardware	Kritisch. DRAGEN-Software wird beendet, das System muss neu gestartet werden.
2	Überhitzung – Platinen-SPD	Wird als nicht kritisch protokolliert.
3	Überhitzung – SODIMM	Wird als nicht kritisch protokolliert.
4	Stromversorgung 0	Kritisch. DRAGEN-Software wird beendet, das System muss neu gestartet werden.
5	Stromversorgung 1	Kritisch. DRAGEN-Software wird beendet, das System muss neu gestartet werden.
6	Stromversorgung – DRAGEN Bio-IT-Prozessor	Wird als nicht kritisch protokolliert.
7	Lüfter 0	Wird als nicht kritisch protokolliert.
8	Lüfter 1	Wird als nicht kritisch protokolliert.
9	SE5338	Kritisch. DRAGEN-Software wird beendet, das System muss neu gestartet werden.
10–30	Nicht definiert (reserviert)	Kritisch. DRAGEN-Software wird beendet, das System muss neu gestartet werden.

Wenn ein kritischer Alarm ausgelöst wird, kann die DRAGEN-Hostsoftware nicht mehr ausgeführt werden und das System muss neu gestartet werden. Wenn die Software eine Überhitzung feststellt und einen Alarm auslöst, werden sämtliche aktiven DRAGEN-Prozesse unterbrochen. Die Überwachung unterbricht neu initiierte DRAGEN-Prozesse, bis die Temperatur den in der Software festgelegten zulässigen Wert erreicht und die Hardware den Alarm für den Chipstatus beendet. Wenn der durch die Software ausgelöste Überhitzungsalarm beendet wird, können DRAGEN-Aufträge fortgesetzt werden.

Wenden Sie sich an den technischen Support von Illumina, wenn einer dieser Alarme in Ihrem System ausgelöst wird. Halten Sie dazu die Protokolldateien bereit.

Hardwarebeschleunigte Komprimierung und Dekomprimierung

Der Komprimierung mit gzip ist in der Bioinformatik üblich. FASTQ-Dateien werden häufig mit gzip komprimiert. Beim BAM-Format handelt es sich um eine spezielle Variante des gzip-Formats. Aus diesem Grund bietet der DRAGEN Bio-IT-Prozessor Hardwareunterstützung, dank der sich Daten mit gzip schneller komprimieren und dekomprimieren lassen. DRAGEN erkennt mit gzip komprimierte Eingabedateien und dekomprimiert diese automatisch. Ebenso werden BAM-Dateien bei der Ausgabe automatisch komprimiert.

DRAGEN bietet eigene Befehlszeilenoptionen zur Komprimierung und Dekomprimierung beliebiger Dateien. Diese entsprechen den Linux-Befehlen „gzip“ und „gunzip“, lauten jedoch *dzip* und *dunzip* (kurz für „dragen zip“ und „dragen unzip“). Beide Befehle akzeptieren eine einzelne Datei als Eingabe und erstellen eine einzelne Ausgabedatei, bei der die .gz-Dateierweiterung hinzugefügt bzw. entfernt wird. Beispiel:

```
dzip file1     # generiert die Ausgabedatei file1.gz
dunzip file2.gz # generiert die Ausgabedatei file2
```

Derzeit bestehen bei *dzip* und *dunzip* im Vergleich zu gzip/gunzip die folgenden Einschränkungen und Unterschiede:

- Mit den Befehlen kann jeweils nur eine Datei verarbeitet werden. Zusätzliche Dateinamen (auch mit dem Platzhalterzeichen * generierte) werden ignoriert.
- Die Befehle können nicht gleichzeitig mit der DRAGEN-Hostsoftware ausgeführt werden.
- Befehlszeilenoptionen, die bei gzip und gunzip verwendet werden können (z. B. --recursive, --fast, --best, --stdout), werden nicht unterstützt.

Nutzungsberichte

Während der Installation wird ein Daemon (dragen_licd) erstellt (oder angehalten und neu gestartet). Dieser Hintergrundprozess wird am Ende jedes Tages automatisch aktiviert und lädt die Nutzungsdaten zur DRAGEN-Hostsoftware auf einen Illumina-Server hoch. Die Daten umfassen das Datum, die Dauer, die Größe (Anzahl der Basen), den Status der einzelnen Läufe und die Version der verwendeten Software.

Die Kommunikation mit dem Illumina-Server wird durch Verschlüsselung geschützt. Bei einem Kommunikationsfehler versucht der Daemon bis zum nächsten Morgen, den Vorgang erneut durchzuführen. Wenn der Upload weiterhin fehlschlägt, erfolgt in der nächsten Nacht ein weiterer Versuch, bis der Upload erfolgreich durchgeführt wurde. Auf diese Weise stehen die Systemressourcen während der Arbeitszeit uneingeschränkt zur Verfügung.

Mit dem Befehl *dragen_lic* lässt sich die derzeitige Lizenznutzung überprüfen.

Kapitel 8 Fehlerbehebung

Gehen Sie folgendermaßen vor, wenn das DRAGEN-System nicht reagiert:

1 Befolgen Sie die Anweisungen unter *Ermitteln, ob sich das System aufgehängt hat*, um festzustellen, ob sich das DRAGEN-System aufgehängt hat.

2 Erfassen Sie nach dem Aufhängen bzw. Abstürzen des Systems Diagnosedaten, wie unter *Senden von Diagnosedaten an den Illumina-Support* beschrieben.

3 Setzen Sie gegebenenfalls das System nach dem Erfassen aller Daten zurück, wie unter *Zurücksetzen eines aufgehängten oder abgestürzten Systems* beschrieben.

Ermitteln, ob sich das System aufgehängt hat

Das DRAGEN-System wird durch einen Watchdog auf Ausfälle überwacht. Falls ein Lauf mehr Zeit als erforderlich benötigt, wird der Ausfall möglicherweise nicht durch den Watchdog festgestellt. Versuchen Sie diese Schritte:

- Suchen Sie mithilfe des Befehls *top* den aktiven DRAGEN-Prozess. Bei ordnungsgemäßer Durchführung des Laufs sollte er über 100 % der CPU-Leistung in Anspruch nehmen. Nimmt der Prozess maximal 100 % in Anspruch, hat sich Ihr System möglicherweise aufgehängt.
- Führen Sie im Verzeichnis der BAM-/SAM-Ausgabedatei den Befehl *du -s* aus. Während eines ordnungsgemäßen Laufs sollte dieses Verzeichnis entweder mit vorläufigen Ausgabedaten (bei aktivierter Sortierung) oder mit BAM-/SAM-Daten gefüllt werden.

Senden von Diagnosedaten an den Illumina-Support

Illumina schätzt Ihr Feedback zum DRAGEN-System, einschließlich Berichten zu Fehlfunktionen des Systems. Führen Sie den Befehl *sosreport* aus, wenn das System abstürzt, sich aufhängt oder ein Watchdog-Fehler auftritt, um Diagnose- und Konfigurationsdaten zu erfassen. Gehen Sie dabei wie folgt vor:

```
sudo sosreport --batch --tmp-dir /staging/tmp
```

Das Ausführen dieses Befehls nimmt mehrere Minuten in Anspruch. Der Speicherort der Diagnosedaten unter /staging/tmp wird angegeben. Fügen Sie diesen Bericht bitte als Anlage an, wenn Sie ein Ticket für den technischen Support von Illumina erstellen.

Zurücksetzen eines aufgehängten oder abgestürzten Systems

Wenn das DRAGEN-System abstürzt oder sich aufhängt, muss das Hilfsprogramm *dragen_reset* ausgeführt werden, um die Hardware und Software neu zu initialisieren. Dieses Hilfsprogramm wird immer dann automatisch von der Hostsoftware ausgeführt, wenn sie einen unerwarteten Zustand erkennt. In diesem Fall zeigt die Hostsoftware folgende Meldung an:

```
Running dragen_reset to reset DRAGEN Bio-IT processor and software
```

Wenn sich die Software aufhängt, erfassen Sie bitte wie in Unterabschnitt *Senden von Diagnosedaten an den Illumina-Support* auf Seite 147 beschrieben Diagnosedaten und führen Sie dann *dragen_reset* wie folgt manuell aus:

```
/opt/edico/bin/dragen_reset
```

Bei jeder Ausführung von *dragen_reset* muss das Referenzgenom neu in das DRAGEN-Board geladen werden. Die Hostsoftware lädt die Referenz bei der nächsten Ausführung automatisch neu.

Anhang A Befehlszeilenoptionen

Hostsoftware-Optionen

Die folgenden Informationen sind im Standardabschnitt der Konfigurationsdatei enthalten. Der Standardabschnitt verfügt nicht über eine spezifische Bezeichnung (wie z. B. [Aligner]). Der Standardabschnitt befindet sich in der Konfigurationsdatei ganz oben. Beachten Sie, dass einige Pflichtfelder in der Befehlszeile angegeben werden müssen und nicht in Konfigurationsdateien enthalten sind.

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
alt-aware	Aktiviert die Sonderverarbeitung für alternative Contigs, wenn in der Hashtabelle Alt-Liftover verwendet wurde. Standardmäßig aktiviert, wenn die Referenz mit Liftover erstellt wurde.	--alt-aware	true/false
annotation-file	Transkript-Annotationsdatei (RNA).	--annotation-file, -a	
append-read-index-to-name	Standardmäßig erhalten in DRAGEN die beiden zusammengehörigen Enden eines Paares dieselbe Bezeichnung. Ist diese Option auf „true“ festgelegt, fügt DRAGEN an die beiden Enden /1 und /2 an.		true/false
bam-input	Alignierte BAM-Datei für die Eingabe in den DRAGEN-Varianten-Caller.	-b, --bam-input	
bcl-conversion-only	Konvertierung von Illumina BCL in das FASTQ-Format.	--bcl-conversion-only	
bcl-input-directory	Verzeichnis der BCL-Eingabedatei für die BCL-Konvertierung.	--bcl-input-directory	
sample-sheet	Für die BCL-Eingabe, Pfad zur Datei SampleSheet.csv. Der Standardspeicherort ist das BCL-Stammverzeichnis.	--sample-sheet	
bcl-use-hw	Legen Sie diese Option auf „false“ fest, wenn die Verwendung der DRAGEN-FPGA-Beschleunigung während der BCL-Konvertierung verhindert werden soll. Die Standardeinstellung ist „true“.	--bcl-use-hw	true/false
build-hash-table	Zur Generierung einer Referenz-/Hashtabelle.	--build-hash-table	true/false
cram-input	CRAM-Datei für die Eingabe in den DRAGEN-Varianten-Caller.	--cram-input	
dbsnp	Pfad zur VCF-Datei (oder .vcf.gz) für die Variantenannotationsdatenbank.	--dbsnp	
enable-auto-multifile	Zum Importieren nachfolgender Segmente der *_001.{dbam,fastq}-Dateien.	--enable-auto-multifile	true/false
enable-bam-indexing	Zur Aktivierung der Generierung einer BAI-Indexdatei.	--enable-bam-indexing	true/false
enable-cnv	Zur Aktivierung der Kopienzahlvariante (CNV).	--enable-cnv	true/false
enable-duplicate-marking	Zur Aktivierung der Kennzeichnung doppelter Ausgabe-Alignment-Datensätze.	--enable-duplicate-marking	true/false

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
enable-map-align-output	Aktiviert die Speicherung der Ausgabe aus der Mapping-/Alignment-Phase. Die Standardeinstellung ist „true“, wenn nur Mapping/Alignment ausgeführt wird. Die Standardeinstellung ist „false“, wenn der Varianten-Caller ausgeführt wird.	--enable-map-align-output	true/false
enable-methylation-calling	Gibt an, ob Methylierungs-Tags automatisch hinzugefügt werden und eine einzelne BAM für die Methylierungsprotokolle ausgegeben wird.		true/false
enable-rna	Aktivierung der Verarbeitung von RNS-Seq-Daten.	--enable-rna	true/false
enable-rna-quantification	Aktiviert/deaktiviert die RNA-Quantifizierung. Wird diese Option auf „true“ festgelegt, muss auch enable-rna auf „true“ festgelegt werden.	--enable-rna-quantification	true/false
enable-sampling	Automatische Erkennung von Paired-End-Parametern durch Verarbeitung einer Probe mit dem Mapper/Aligner.		true/false
enable-sort	Aktivierung der Sortierung nach Mapping/Alignment.		true/false
enable-variant-caller	Aktiviert den Varianten-Caller.	--enable-variant-caller	true/false
enable-vcf-compression	Aktivierung der Komprimierung von VCF-Ausgabedateien. Die Standardeinstellung ist „true“.		true/false
fastq-file1	FASTQ-Datei für die Eingabe in die DRAGEN-Pipeline (kann mit gzip komprimiert werden).	-1, --fastq-file1	
fastq-file2	Zweite FASTQ-Datei mit Paired-End-Reads für die Eingabe.	-2, --fastq-file2	
fastq-list	CSV-Datei mit einer Liste der zu verarbeitenden FASTQ-Dateien.	--fastq-list	
fastq-list-all-samples	Aktivierung/Deaktivierung der gemeinsamen Verarbeitung aller Proben unabhängig vom RGSM-Wert.	--fastq-list-all-samples	true/false
fastq-n-quality	Base-Call-Ausgabequalität für N-Basen. Wird für alle Ausgabe-Ns automatisch zu fastq-n-quality hinzugefügt.	--fastq-n-quality	0 bis 255
fastq-offset	FASTQ-Qualitätsversatzwert.	--fastq-offset	33 oder 64
filter-flags-from-output	Filterung der Ausgabe-Alignments mit allen Bits, die in Werten im Kennzeichnungsfeld vorhanden sind. Es sind Hexadezimal- und Dezimalwerte zulässig.	--filter-flags-from-output	
first-tile-only	Zur ausschließlichen Konvertierung der ersten Platte jeder Lane während der BCL-Konvertierung.	--first-tile-only	
force	Erzwingen der Überschreibung der vorhandenen Ausgabedatei.	-f	
force-load-reference	Erzwingen des Ladens der Referenz- und Hashtabellen vor dem Starten der DRAGEN-Pipeline.	-l	
generate-md-tags	Legt fest, ob mit Alignment-Ausgabedatensätzen MD-Tags generiert werden. Die Standardeinstellung ist „false“.	--generate-md-tags	true/false
generate-sa-tags	Legt fest, ob für Datensätze mit chimärischen/ergänzenden Alignments SA:Z-Tags generiert werden.	--generate-sa-tags	true/false

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
generate-zs-tags	Legt fest, ob für Alignment-Ausgabedatensätze ZS-Tags generiert werden. Die Standardeinstellung ist „false“.	--generate-sz-tags	true/false
ht-alt-liftover	Liftover-Datei im SAM-Format für alternative Contigs in der Referenz.	--ht-alt-liftover	
ht-alt-aware-validate	Deaktivierung der Erforderlichkeit einer Liftover-Datei beim Generieren einer Hashtabelle aus einer Referenz mit alternativen Contigs.	--ht-alt-aware-validate	true/false
ht-build-rna-hashtable	Aktivierung der Generierung einer RNA-Hashtabelle. Die Standardeinstellung ist „false“.	--ht-build-rna-hashtable	true/false
ht-cost-coeff-seed-freq	Kostenkoeffizient der erweiterten Seed-Frequenz.	--ht-cost-coeff-seed-freq	
ht-cost-coeff-seed-len	Kostenkoeffizient der erweiterten Seed-Länge.	--ht-cost-coeff-seed-len	
ht-cost-penalty-incr	Kostenauswirkung der inkrementellen Erweiterung eines Seeds um einen weiteren Schritt.	--ht-cost-penalty-incr	
ht-cost-penalty	Kostenauswirkung der Erweiterung eines Seeds um eine beliebige Anzahl von Basen.	--ht-cost-penalty	
ht-decoys	Gibt den Pfad zu einer Decoys-Datei an.	--ht-decoys	
ht-max-dec-factor	Maximaler Minderungsfaktor für das Seed Thinning.	--ht-max-dec-factor	
ht-max-ext-incr	Maximale Anzahl von Basen, um die ein Seed in einem Schritt erweitert werden kann.	--ht-max-ext-incr	
ht-max-ext-seed-len	Maximale erweiterte Seed-Länge.	-- ht-max-ext-seed-len	
ht-max-seed-freq	Maximale zulässige Häufigkeit für eine Seed-Übereinstimmung nach Erweiterungsversuchen.	--ht-max-seed-freq	1–256
ht-max-table-chunks	Maximale Anzahl von Thread-Tabellenabschnitten mit ca. 1 GB, die sich gleichzeitig im Arbeitsspeicher befinden dürfen.	--ht-max-table-chunks	
ht-mem-limit	Arbeitsspeicherbegrenzung (Hashtabelle und Referenz), Einheiten B|KB|MB|GB.	--ht-mem-limit	
ht-methylated	Automatische Generierung von C->T- und G->A-konvertierten Referenz-Hashtabellen.	--ht-methylated	true/false
ht-num-threads	Maximale Anzahl der Worker-CPU-Threads für die Generierung einer Hashtabelle.	--ht-num-threads	
ht-rand-hit-extend	Aufnahme eines Zufallstreffers für jeden EXTEND-Datensatz dieses Häufigkeitsdatensatzes.	--ht-rand-hit-extend	
ht-rand-hit-hifreq	Aufnahme eines Zufallstreffers für jeden HIFREQ-Datensatz.	--ht-rand-hit-hifreq	
ht-ref-seed-interval	Anzahl der Positionen pro Referenz-Seed.	--ht-ref-seed-interval	
ht-reference	Referenzdatei im .fasta-Format für die Generierung einer Hashtabelle.	--ht-reference	
ht-seed-len	Ursprüngliche Seed-Länge zur Speicherung in der Hashtabelle.	--ht-seed-len	

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
ht-size	Größe der Hashtabelle, Einheiten B|KB|MB|GB.	--ht-size	
ht-soft-seed-freq-cap	Weiche Seed-Frequenzgrenze für das Ausdünnen.	--ht-soft-seed-freq-cap	
ht-suppress-decoys	Unterdrückung der Verwendung einer Decoys-Datei beim Generieren einer Hashtabelle.	--ht-suppress-decoys	
ht-target-seed-freq	Target-Seed-Frequenz für die Seed-Extension.	--ht-target-seed-freq	
input-qname-suffix-delimiter	Legt das Trennzeichen für append-read-index-to-name und zur Erkennung der Bezeichnungen zusammengehöriger Paare bei der BAM-Eingabe fest.		/ oder . oder :
interleaved	Überlappende Paired-End-Reads in einer einzelnen FASTQ.	-i	
intermediate-results-dir	Verzeichnis für die Speicherung von Zwischenergebnissen (z. B. Sortierungspartitionen).		
lic-no-print	Unterdrückung der Lizenzstatusmeldung am Ende eines Laufs.	--lic-no-print	true/false
mapq-strict-js	RNA-spezifisch. Ist diese Option auf 0 festgelegt, wird ein höherer MAPQ-Wert zurückgegeben, wodurch das Alignment als zumindest teilweise korrekt klassifiziert wird. Ist diese Option auf 1 festgelegt, wird ein niedrigerer MAPQ-Wert zurückgegeben, wodurch die Spleißstelle als mehrdeutig klassifiziert wird.	--mapq-strict-js	0/1
methylation-generate-cytosine-report	Generierung eines genomweiten Cytosin-Methylierungsberichts.	--methylation-generate-cytosine-report	true/false
methylation-generate-mbias-report	Generierung eines Methylierungsabweichungsberichts für einzelne Sequenziererzyklen.		true/false
methylation-match-bismark	Bei Festlegung auf „true“ genaue Übereinstimmung mit Bismark-Tags einschließlich Fehlern.	--methylation-match-bismark	true/false
methylation-protocol	Bibliotheksprotokoll für die Methylierungsanalyse.	--methylation-protocol	none / directional / nondirectional / directional-complement
num-threads	Die Anzahl der zu verwendenden Prozessor-Threads.	-n, --num-threads	
output-directory	Ausgabeverzeichnis.	--output-directory	
output-file-prefix	Präfix für den Namen aller in der Pipeline generierten Ausgabedateien.	--output-file-prefix	
output-format	Das Format der Ausgabedatei der Mapping-Alignment-Phase. Gültige Werte sind „bam“ (Standardwert), „sam“ oder „dbam“ (ein proprietäres Binärformat).	--output-format	BAM/SAM/DBAM
pair-by-name	Gibt an, ob die Reihenfolge der BAM-Eingabedatensätze so festgelegt werden soll, dass zusammengehörige Paired-End-Paare gemeinsam verarbeitet werden.		

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
pair-suffix-delimiter	Änderung der Trennzeichen für Suffixe.	--pair-suffix-delimiter	/ . :
preserve-bqsr-tags	Gibt an, ob die BI- und BD-Kennzeichnungen der BAM-Eingabedatei beibehalten werden sollen. Beachten Sie, dass dies u. U. Probleme bei Hard Clipping verursacht.		true/false
preserve-map-align-order	Erstellung einer Ausgabedatei mit der ursprünglichen Read-Reihenfolge der Eingabedatei.		true/false
qc-coverage-region-1	Erste BED-Datei für die Coverage-Berichterstellung.	--qc-coverage-region-1	
qc-coverage-region-2	Zweite BED-Datei für die Coverage-Berichterstellung.	--qc-coverage-region-2	
qc-coverage-region-3	Dritte BED-Datei für die Coverage-Berichterstellung.	--qc-coverage-region-3	
qc-coverage-reports-1	Arten der für qc-coverage-region-1 angeforderten Berichte.	--qc-coverage-reports-1	full_res / cov_report
qc-coverage-reports-2	Arten der für qc-coverage-region-2 angeforderten Berichte.	--qc-coverage-reports-2	full_res / cov_report
qc-coverage-reports-3	Arten der für qc-coverage-region-3 angeforderten Berichte.	--qc-coverage-reports-3	full_res / cov_report
ref-dir	Verzeichnis mit der Referenz-Hashtabelle. Diese Referenz wird, wenn nicht bereits erfolgt, automatisch in die DRAGEN-Karte geladen.	-r, --ref-dir	
ref-sequence-filter	Zur Ausgabe von Reads nur für diese Referenzsequenz.	--ref-sequence-filter	
remove-duplicates	Ist diese Option auf „true“ festgelegt, werden doppelte Alignment-Datensätze entfernt, statt diese nur zu kennzeichnen.		true/false
RGCN	Bezeichnung des Sequenzierungszentrums der Read-Gruppe.	--RGCN	
RGCN-tumor	Bezeichnung des Sequenzierungszentrums der Read-Gruppe für die Tumor-Eingabe.	--RGCN-tumor	
RGDS	Beschreibung der Read-Gruppe.	--RGDS	
RGDS-tumor	Beschreibung der Read-Gruppe für die Tumor-Eingabe.	--RGDS-tumor	
RGDT	Laufdatum der Read-Gruppe.	--RGDT	
RGDT-tumor	Laufdatum der Read-Gruppe für die Tumor-Eingabe.	--RGDT-tumor	
RGID	ID der Read-Gruppe.	--RGID	
RGID-tumor	ID der Read-Gruppe für die Tumor-Eingabe.	--RGID-tumor	
RGLB	Bibliothek der Read-Gruppe.	--RGLB	
RGLB-tumor	Bibliothek der Read-Gruppe für die Tumor-Eingabe.	--RGLB-tumor	
RGPI	Prognostizierte Insert-Größe der Read-Gruppe.	--RGPI	
RGPI-tumor	Prognostizierte Insert-Größe der Read-Gruppe für die Tumor-Eingabe.	--RGPI-tumor	
RGPL	Sequenzierungstechnologie der Read-Gruppe.	--RGPL	

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
RGPL-tumor	Sequenzierungstechnologie der Read-Gruppe für die Tumor-Eingabe.	--RGPL-tumor	
RGPU	Plattformeinheit der Read-Gruppe.	--RGPU	
RGPU-tumor	Plattformeinheit der Read-Gruppe für die Tumor-Eingabe.	--RGPU-tumor	
RGSM	Bezeichnung der Probe der Read-Gruppe.	--RGSM	
RGSM-tumor	Bezeichnung der Probe der Read-Gruppe für die Tumor-Eingabe.	--RGSM-tumor	
rna-ann-sj-min-len	Während der Generierung von Spleißstellen aus einer Annotationsdatei (GTF/GFF/SJ.out.tab) werden Spleißstellen mit einer Länge unter diesem Wert ignoriert.		
rna-gf-input-file	Eine bereits erstellte .Chimeric.out.junction-Datei. Wenn diese Datei bereitgestellt wird, erfolgt die Ausführung des DRAGEN Gene Fusion-Moduls im eigenständigen Modus.	--rna-gf-input-file	
rna-mapq-unique	Aktivieren Sie für die Kompatibilität mit Cufflinks diesen Parameter mit einem Wert ungleich null. Bei eindeutigen Mappern ist diesem Wert eine MAPQ zugeordnet. Multimapper weisen eine MAPQ von int(-10*log10(1 – 1 /NH)) auf.		0, 1 bis 255
rna-quantification-fld-max	Gibt die Insert-Größenverteilung der RNA-Seq-Bibliothek für Single-End-Läufe an. Der Standardwert ist 250 +- 25.	--rna-quantification-fld-max	Der Maximalwert beträgt 1000.
rna-quantification-fld-mean		--rna-quantification-fld-mean	
rna-quantification-fld-sd		--rna-quantification-fld-sd	
rna-quantification-library-type	Gibt die Art der RNA-Seq-Bibliothek an. Der Standardwert ist A (automatische Erkennung).	--rna-quantification-library-type	IU, ISR, ISF, U, SR, SF oder A.
sample-size	Anzahl der zu berücksichtigenden Reads, wenn enable-sampling auf „true“ festgelegt ist.		
sample-sex	Geschlecht der Probe.	--sample-sex	
strict-mode	Abbruch, wenn Dateien fehlen.	--strict-mode	
strip-input-qname-suffixes	Gibt an, ob Read-Indexsuffixe (z. B. /1 und /2) von Eingabe-QNAMEs entfernt werden.		true/false
tumor-bam-input	Alignierte BAM-Datei für den DRAGEN-Varianten-Caller im somatischen Modus.	--tumor-bam-input	
tumor-cram-input	Alignierte CRAM-Datei für den DRAGEN-Varianten-Caller im somatischen Modus.	--tumor-cram-input	
tumor-fastq-list	Eine CSV-Datei mit einer Liste von FASTQ-Dateien für den Mapper, den Aligner und den Caller für somatische Varianten.	--tumor-fastq-list	
tumor-fastq-list-sample-id	Die Proben-ID für die mit tumor-fastq-list angegebene Liste von FASTQ-Dateien.	--tumor-fastq-list-sample-id	

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
tumor-fastq1	FASTQ-Datei für die DRAGEN-Pipeline, die den Varianten-Caller im somatischen Modus verwendet (kann mit gzip komprimiert werden).	--tumor-fastq1	
tumor-fastq2	Zweite FASTQ-Datei, die mit tumor-fastq1-Reads gepaarte Reads für die DRAGEN-Pipeline enthält, die den Varianten-Caller im somatischen Modus verwendet (kann mit gzip komprimiert werden).	--tumor-fastq2	
umi-enable	Aktivierung der UMI-basierten Read-Verarbeitung.	--umi-enable	true/false
umi-min-supporting-reads	Anzahl der Eingabe-Reads mit zugehörigem UMI und für die Generierung des Konsens-Reads erforderlicher Position.	--umi-min-supporting-reads	
verbose	Aktivierung der Verbose-Ausgabe aus DRAGEN.	-v	
version	Ausgabe der Version und Beendigung.	-V	

Mapper-Optionen

Die folgenden Optionen sind im Abschnitt [Mapper] der Konfigurationsdatei enthalten. Ausführliche Informationen zu diesen Optionen finden Sie unter *DNA-Mapping* auf Seite 16.

Name	Beschreibung	Befehlszeilenentsprechung	Bereich
ann-sj-max-indel	In der Nähe einer annotierten Spleißstelle zu erwartende maximale Indel-Länge.	--Mapper.ann-sj-max-indel	0 bis 63
edit-chain-limit	edit-mode 1 oder 2: Maximale Seed-Kettenlänge in einem Read zur Qualifizierung für die Seed-Bearbeitung.	--Mapper.edit-chain-limit	edit-chain-limit >= 0
edit-mode	0 = keine Bearbeitungen, 1 = Kettenlängentest, 2 = gepaarter Kettenlängentest, 3 = alle Standard-Seeds bearbeiten.	--Mapper.edit-mode	0 bis 3
edit-read-len	edit-mode 1 oder 2: Read-Länge, bei der mit edit-seed-num bearbeitete Seeds getestet werden.	--Mapper.edit-read-len	edit-read-len > 0
edit-seed-num	edit-mode 1 oder 2: Angeforderte Anzahl von Seeds pro Read, die bearbeitet werden sollen.	--Mapper.edit-seed-num	edit-seed-num >= 0
enable-map-align	Aktiviert die Verwendung von BAM-Eingabedateien für Mapper/Aligner.	--enable-map-align	true/false
map-orientations	0=Normal, 1=Kein umg. Komp., 2=Kein vorwärts (für Paired-Ends ist „Normal“ erforderlich).	--Mapper.map-orientations	0 bis 2
max-intron-bases	Maximale gemeldete Intron-Länge.	--Mapper.max-intron-bases	
min-intron-bases	Minimale Referenz-Deletionslänge, die als Intron gemeldet wird.	--Mapper.min-intron-bases	
seed-density	Angeforderte Seed-Dichte von in der Hashtabelle abgefragten Reads.	--Mapper.seed-density	0 > seed-density > 1

Aligner-Optionen

Die folgenden Optionen sind im Abschnitt [Aligner] der Konfigurationsdatei enthalten. Weitere Informationen finden Sie unter *DNA-Alignierung* auf Seite 19

Name	Beschreibung	Befehlszeilenentsprechung	Wert
aln-min-score	Minimaler Alignment-Score (mit Vorzeichen) für den Bericht; Basisniveau für MAPQ. Bei Verwendung von lokalen Alignments (global = 0) wird aln-min-score von der Hostsoftware wie folgt berechnet: 22 * match-score. Bei Verwendung von globalen Alignments (global = 1) ist aln-min-score auf -1000000 festgelegt. Die Berechnung der Hostsoftware kann überschrieben werden, indem aln-min-score in der Konfigurationsdatei festlegt wird.	--Aligner.aln-min-score	-2,147,483,648 bis 2,147,483,647
dedup-min-qual	Mindestbasenqualität zur Berechnung der Read-Qualitätsmetrik für die Deduplikation.	--Aligner.dedup-min-qual	0–63
en-alt-hap-aln	Lässt die Ausgabe chimärischer Alignments als Ergänzung zu.	--Aligner.en-alt-hap-aln	0–1
en-chimeric-aln	Lässt die Ausgabe chimärischer Alignments als Ergänzung zu.	--Aligner.en-chimeric-aln	0–1
gap-ext-pen	Score-Abzug für Lückenerweiterung.	--Aligner.gap-ext-pen	0–15
gap-open-pen	Score-Abzug für die Öffnung einer Lücke (Insertion oder Deletion).	gap-open-pen	0–127
global	Alignment ist global (Needleman-Wunsch), nicht lokal (Smith-Waterman).	--Aligner.global	0–1
hard-clips	Kennzeichnungen für Hard Clipping: [0] primär, [1] ergänzend, [2] sekundär.	--Aligner.hard-clips	3 Bits
map-orientations	Beschränkt die Ausrichtung auf folgende Alignment-Optionen: nur vorwärts, nur in Richtung des Gegenstrangs (umgekehrtes Komplement) oder beliebig.	--Aligner.map-orientations	0 (beliebig) 1 (nur vorwärts) 2 (nur rückwärts)
mapq-max	Obergrenze für gemeldeten MAPQ-Wert.	--Aligner.mapq-max	0 bis 255
match-n-score	Score-Inkrement (mit Vorzeichen) für die Übereinstimmung mit einem Referenz-Nukleotid-IUB-Code „N“.	--Aligner.match-n-score	-16–15
match-score	Score-Inkrement für die Übereinstimmung mit einem Referenz-Nukleotid.	--Aligner.match-score	Wenn global = 0, dann match-score > 0; wenn global = 1, dann match-score >= 0
max-rescues	Maximale Anzahl an Rescue-Alignments pro Read-Paar. Der Standardwert ist 10.	--max-rescues	0–1023
min-score-coeff	Anpassung an aln-min-score pro Read-Base.	--Aligner. min-score-coeff	-64–63.999

Name	Beschreibung	Befehlszeilenentsprechung	Wert
mismatch-pen	Score-Abzug für Nichtübereinstimmung.	--Aligner.mismatch-pen	0–63
no-unclip-score	Wenn no-unclip-score auf 1 festgelegt ist, werden alle Zusätze ohne Clipping (unclip-score), die für ein Alignment verwendet werden, vor der weiteren Verarbeitung aus dem Alignment-Score entfernt.	--Aligner.no-unclip-score	0–1
no-unpaired	Wenn nur korrekt gepaarte Alignments für gepaarte Reads gemeldet werden sollen.	--Aligner. no-unpaired	0–1
pe-max-penalty	Maximaler Paar-Score-Abzug für nicht gepaarte oder entfernt liegende Enden.	--Aligner.pe-max-penalty	0–255
pe-orientation	Erwartete Paired-End-Ausrichtung: 0=FR, 1=RF, 2=FF.	--Aligner.pe-orientation	0, 1, 2
rescue-sigmas	Abweichungen von der mittleren Read-Länge, die für den Rescue-Scan-Radius verwendet werden. Der Standardwert ist 2.5.	--Aligner.rescue-sigmas	
sec-aligns	Maximale sekundäre (suboptimale) Alignments, die pro Read gemeldet werden.	--Aligner.sec-aligns	0–30
sec-aligns-hard	Auf „force unmapped“ festlegen, wenn nicht alle sekundären Alignments ausgegeben werden können.	--Aligner.sec-aligns-hard	0–1
sec-phred-delta	Nur sekundäre Alignments, die sich wahrscheinlich innerhalb dieses Phreds des primären Alignments befinden, werden gemeldet.	--Aligner.sec-phred-delta	0–255
sec-score-delta	Es sind nur sekundäre Alignments zulässig, deren Paar-Score den Wert für das primäre Alignment um höchstens diesen Wert unterschreitet.	--Aligner. sec-score-delta	
supp-aligns	Maximale ergänzende (chimärische) Alignments, die pro Read gemeldet werden.	--Aligner.supp-aligns	0–30
supp-as-sec	Wenn ergänzende Alignments mit sekundärer Kennzeichnung gemeldet werden sollen.	--Aligner.supp-as-sec	0–1
supp-min-score-adj	Die erforderliche Erhöhung des minimalen Alignment-Scores für ergänzende Alignments. Dieser Score wird von der Hostsoftware für DNA wie folgt berechnet: 8 * match-score. Für RNA ist der Standardwert 0.	--Aligner. supp-min-score-adj	
unclip-score	Score-Zusatz für das Erreichen der Read-Enden.	--Aligner.unclip-score	0–127
unpaired-pen	Abzug für nicht gepaarte Alignments in der Phred-Skala.	--Aligner.unpaired-pen	0–255

Wenn Sie die automatische Erkennung der Insert-Längenstatistik über die Option *--enable-sampling* deaktivieren, müssen zur Festlegung der Statistik alle folgenden Optionen überschrieben werden. Weitere Informationen finden Sie unter *Bestimmung der mittleren Insert-Größe* auf Seite 22. Diese Optionen sind im Abschnitt [Aligner] der Konfigurationsdatei enthalten.

Option	Beschreibung	Befehlszeilenentsprechung	Wert
pe-stat-mean-insert	Durchschnittliche Matrizenlänge.		0–65535
pe-stat-mean-read-len	Durchschnittliche Read-Länge.		0–65535
pe-stat-quartiles-insert	Ein durch Kommas getrenntes Zahlentrio für das 25., 50. und 75. Perzentil der Matrizenlängen.		0–65535
pe-stat-stddev-insert	Standardabweichung der Matrizenlängenverteilung.		0–65535

Varianten-Caller-Optionen

Die folgenden Optionen sind im Abschnitt „Variant Caller“ (Varianten-Caller) der Konfigurationsdatei enthalten. Weitere Informationen hierzu finden Sie unter *Varianten-Caller-Optionen* auf Seite 29.

Name	Beschreibung	Befehlszeilenentsprechung	Wert
cnv-blacklist-bed	Legt eine BED-Datei mit Intervallen fest, die aus der endgültigen Ausgabe ausgeschlossen werden.	--cnv-blacklist-bed	
cnv-cbs-alpha	Signifikanzniveau für den Test hinsichtlich der Akzeptanz von Changepoints. Der Standardwert ist 0.01.	cnv-cbs-alpha	
cnv-cbs-eta	Typ-1-Fehlerrate des sequenziellen Grenzwerts für ein frühzeitiges Beenden bei Verwendung des Permutationsverfahrens. Der Standardwert ist 0.05.	--cnv-cbs-eta	
cnv-cbs-kmax	Maximale Breite des kleineren Segments für die Permutation. Der Standardwert ist 25.	--cnv-cbs-kmax	
cnv-cbs-min-width	Mindestanzahl von Markern für ein geändertes Segment. Der Standardwert ist 2.	--cnv-cbs-min-width	
cnv-cbs-nmin	Mindestdatenlänge für eine maximale statistische Näherung. Der Standardwert ist 200.	--cnv-cbs-nmin	
cnv-cbs-nperm	Anzahl an Permutationen für die Berechnung des p-Werts. Der Standardwert ist 10000.	--cnv-cbs-nperm	
cnv-cbs-trim	Anteil der Daten, die für die Varianzberechnungen gekürzt werden müssen. Der Standardwert ist 0.025.	--cnv-cbs-trim	
cnv-counts-method	Zählungsmethode für ein Alignment, für das eine Zählung in einer Target-Klasse erstellt werden soll. Die Standardeinstellung für einen Ansatz mit Normalgruppe ist „overlap“. Die Standardeinstellung für eine Selbstnormalisierung ist „start“.	--cnv-counts-method	midpoint/start/overlap
cnv-enable-gcbias-correction	Aktiviert/deaktiviert die Korrektur der GC-Verzerrung.	--cnv-enable-gcbias-correction	true/false

Name	Beschreibung	Befehlszeilenentsprechung	Wert
cnv-enable-gcbias-smoothing	Aktiviert/deaktiviert eine Glättung der Korrektur der GC-Verzerrung über benachbarte GC-Klassen mit exponentiellem Kernel. Die Standardeinstellung ist „true“.	--cnv-enable-gcbias-smoothing	true/false
cnv-enable-plots	Aktiviert/deaktiviert die Erstellung von Plots. Die Standardeinstellung ist „false“.	--cnv-enable-plots	true/false
cnv-enable-ref-calls	Bei „true“ werden Calls ohne Einfluss auf die Kopienzahl (REF) in die Ausgabe-VCF aufgenommen.	--cnv-enable-ref-calls	true/false
cnv-enable-self-normalization	Aktiviert/deaktiviert die Selbstnormalisierung.	--cnv-enable-self-normalization	true/false
cnv-enable-tracks	Aktiviert/deaktiviert die Erstellung von Verfolgungsdateien, die zur Anzeige in IGV importiert werden können. Die Standardeinstellung ist „true“.	--cnv-enable-tracks	true/false
cnv-extreme-percentile	Extremwert der mittleren Perzentile, ab dem Proben herausgefiltert werden. Der Standardwert ist 2.5.	--cnv-extreme-percentile	
cnv-filter-bin-support-ratio	Filtert ein Kandidatenereignis heraus, wenn die Anzahl unterstützender Klassen weniger als das angegebene Verhältnis in Bezug auf die Ereignisgesamtlänge beträgt. Das Standardverhältnis ist 0.2 (20 % Unterstützung).	--cnv-filter-bin-support-ratio	
cnv-filter-copy-ratio	Schwellenwert für das über 1,0 gemittelte Kopienverhältnis, ab dem ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 0.2.	--cnv-filter-copy-ratio	
cnv-filter-de-novo-quality	Phred-skalierter Schwellenwert für ein De-novo-Ereignis-Calling in der Probandenprobe.	--cnv-filter-de-novo-quality	
cnv-filter-length	Minimale Ereignislänge in Basen, ab der ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 10000.	--cnv-filter-length	
cnv-filter-qual	Der QUAL-Wert, ab dem ein gemeldetes Ereignis in der VCF-Ausgabedatei mit PASS gekennzeichnet wird. Der Standardwert ist 10.	--cnv-filter-qual	
cnv-fpop-penalty	Abzugsparameter für die Erkennung von Changepoints. Der Standardwert ist 0.03.	--cnv-fpop-penalty	
cnv-input	Probeneingabe.	--cnv-input	
cnv-interval-width	Gibt die Breite des Sampling-Intervalls für die CNV-WGS-Verarbeitung an. Der Standardwert ist 1000.	--cnv-wgs-interval-width	
cnv-matched-normal	Target-Zählungsdatei der zugeordneten Normalprobe.	--cnv-matched-normal	

Name	Beschreibung	Befehlszeilenentsprechung	Wert
cnv-max-percent-zero-samples	Schwellenwert für das Herausfiltern von Targets mit zu vielen Proben, die eine Coverage von null aufweisen. Der Standardwert ist 5%.	--cnv-max-percent-zero-samples	
cnv-max-percent-zero-targets	Schwellenwert für das Herausfiltern von Proben mit zu vielen Targets, die eine Coverage von null aufweisen. Der Standardwert ist 2.5%.	--cnv-max-percent-zero-targets	
cnv-merge-distance	Legt die Mindestanzahl an Basenpaaren zwischen zwei Segmenten fest, ab der eine Zusammenfassung zulässig ist. Der Standardwert ist 0, d. h., es muss sich um direkt benachbarte Segmente handeln.	--cnv-merge-distance	
cnv-merge-threshold	Der maximale Unterschied zwischen den Segmentmittelwerten, bis zu dem eine Zusammenfassung von zwei benachbarten Segmenten zulässig ist. Der Segmentmittelwert wird als linearer Kopienverhältniswert dargestellt. Der Standardwert ist 0.2. Legen Sie den Wert auf 0 fest, wenn Sie die Zusammenfassung deaktivieren möchten.	--cnv-merge-threshold	
cnv-min-mapq	Aktiviert bzw. deaktiviert die Aufteilung aller Target-BED-Intervalle in zwei Intervalle mit gleichem Abstand. Bei Aktivierung müssen alle Proben (Fallproben und Normalgruppe) mit dieser aktivierten Option ausgeführt werden. Die Standardeinstellung ist „false“.	--cnv-min-mapq	true/false
cnv-normals-file	Eine einzelne Datei zur Verwendung in der Normalgruppe. Kann mehrmals angegeben werden, für jede Datei einmal.	--cnv-normals-file	
cnv-normals-list	Eine Normalgruppendatei.	--cnv-normals-list	
cnv-num-gc-bins	Anzahl der Klassen für die Korrektur der GC-Verzerrung. Jede Klasse repräsentiert den Prozentsatz des GC-Gehalts. Der Standardwert ist 25.	--cnv-num-gc-bins	10/20/25/50/100
cnv-ploidy	Der normale Ploidiewert. Zur Schätzung des Kopienzahlwerts, der in der VCF-Ausgabedatei ausgegeben wird. Der Standardwert ist 2.	--cnv-ploidy	
cnv-qual-length-scale	Gewichtungsfaktor der Verzerrung, um QUAL-Schätzungen für längere Segmente anzupassen. Diese erweiterte Option sollte nicht geändert werden. Der Standardwert ist 0.9303 (2-0.1).	--cnv-qual-length-scale	
cnv-qual-noise-scale	Gewichtungsfaktor der Verzerrung, um QUAL-Schätzungen auf Grundlage der Probenvarianz anzupassen. Diese erweiterte Option sollte nicht geändert werden. Der Standardwert ist 1.0.	--cnv-qual-noise-scale	

Name	Beschreibung	Befehlszeilenentsprechung	Wert
cnv-segmentation-mode	Auszuführender Segmentierungsalgorithmus. Der Standardwert ist „slm“ oder „cbs“, abhängig davon, ob es sich bei den Intervallen um Gesamtgenomintervalle oder gezielte Sequenzierungsintervalle handelt.	--cnv-segmentation-mode	cbs/slm/hslm/fpop
cnv-skip-contig-list	Gibt eine kommagetrennte Liste mit Contig-Bezeichnern an, die beim Generieren von Intervallen für die WGS-Analyse übersprungen werden. Wenn nicht anders angegeben, werden standardmäßig die Contigs „chrM“, „MT“, „m“ und „chrm“ übersprungen.	--cnv-wgs-skip-contig-list	
cnv-slm-eta	Ausgangswahrscheinlichkeit für eine Änderung des Mittelwertprozess-Werts. Der Standardwert ist 1e-5.	--cnv-slm-eta	
cnv-slm-fw	Minimale Anzahl von Datenpunkten für die Ausgabe einer CNV. Der Standardwert ist 0.	--cnv-slm-fw	
cnv-slm-omega	Skalierungsparameter für die relative Gewichtung von experimenteller/biologischer Varianz. Der Standardwert ist 0.3.	--cnv-slm-omega	
cnv-slm-stepeta	Parameter für die Distanznormalisierung. Der Standardwert ist 10000. Nur gültig für „HSLM“.	--cnv-slm-stepeta	
cnv-target-bed	Eine korrekt formatierte BED-Datei, die die Target-Intervalle für das Coverage-Sampling angibt. Wird für die WES-Analyse verwendet.	--cnv-target-bed	
cnv-target-factor-threshold	Der Prozentsatz des mittleren Target-Faktor-Schwellenwerts, ab dem verwendbare Targets herausgefiltert werden. Der Standardwert ist 1% für die Gesamtgenomverarbeitung und 10% für die gezielte Sequenzierungsverarbeitung.	--cnv-target-factor-threshold	
cnv-truncate-threshold	Extreme Ausreißer werden auf Grundlage dieses Prozentschwellenwerts gekürzt. Der Standardwert ist 0.1%.	--cnv-truncate-threshold	
cosmic	Eine COSMIC-VCF-Eingabedatei.	--cosmic	
dn-cnv-vcf	Joint-VCF mit strukturellen Varianten aus dem CNV-Calling-Schritt. Wenn ausgelassen, werden Prüfungen mit überlappenden Kopienzahlvarianten übersprungen.	--dn-cnv-vcf	
dn-input-vcf	Zu filternde Joint-VCF mit kleinen Varianten aus dem De-novo-Calling-Schritt.	--dn-input-vcf	
dn-output-vcf	Dateispeicherort, an dem die gefilterte VCF gespeichert werden soll. Wenn nicht angegeben, wird die VCF-Eingabedatei überschrieben.	--dn-output-vcf	

Name	Beschreibung	Befehlszeilenentsprechung	Wert
dn-sv-vcf	Joint-VCF mit strukturellen Varianten aus dem SV-Calling-Schritt. Wenn ausgelassen, werden Prüfungen mit überlappenden strukturellen Varianten übersprungen.	--dn-sv-vcf	
enable-cnv-tracks	Aktiviert/deaktiviert die Erstellung von bigwig- und gff-Dateien.	--enable-cnv-tracks	true/false
enable-combinegvcfs	Aktiviert/deaktiviert die Zusammenführung von gVCF-Dateien.	--enable-combinegvcfs	true/false
enable-joint-genotyping	Legen Sie diese Option auf „true“ fest, wenn Sie gVCF-Dateien zusammenführen möchten.	--enable-joint-genotyping	true/false
enable-multi-sample-gvcf	Aktiviert/deaktiviert die Erstellung einer Mehrproben-gVCF-Datei. Ist diese Option auf „true“ festgelegt, muss die Eingabe als kombinierte gVCF-Datei erfolgen.	--enable-multi-sample-gvcf	true/false
enable-sv	Aktiviert/deaktiviert den Structural Variant Caller. Die Standardeinstellung ist „false“.	--enable-sv	true/false
enable-vlrd	Aktiviert/deaktiviert Virtual Long Read Detection.	--enable-vlrd	true/false
enable-vqsr	Aktiviert/deaktiviert das VQSR-Nachverarbeitungsmodul.	--enable-vqsr	true/false
panel-of-normals	Der Pfad zur Normalgruppen-VCF-Datei.	--panel-of-normals	
pedigree-file	Joint-Calling-spezifisch. Der Pfad zu einer PED-Stammbaumdatei, die eine strukturierte Beschreibung der familiären Beziehungen zwischen den Proben enthält. Die Stammbaumdatei kann Trios enthalten. Es werden nur Stammbaumdateien mit Trios unterstützt.	--pedigree-file	
qc-snp-DeNovo-quality-threshold	Der Schwellenwert für die Zählung und Berichterstellung von De-novo-SNP-Varianten.	--qc-snp-DeNovo-quality-threshold	
qc-indel-DeNovo-quality-threshold	Der Schwellenwert für die Zählung und Berichterstellung von De-novo-INDEL-Varianten.	--qc-indel-DeNovo-quality-threshold	
sv-call-regions-bed	Gibt eine BED-Datei an, die den Satz mit Regionen für das Calling enthält. Die Datei muss bgzip-komprimiert und tabix-indiziert sein.	--sv-call-regions-bed	
sv-denovo-scoring	Aktiviert/deaktiviert das De-novo-Qualitäts-Scoring struktureller Varianten für das Joint Diploid-Calling. Die Stammbaumdatei muss ebenfalls angegeben werden.	--sv-denovo-scoring	
sv-exome	Wenn dieser Wert auf „true“ festgelegt ist, wird der Varianten-Caller für gezielte Sequenzierungseingaben konfiguriert. Filter mit großer Tiefe werden deaktiviert. Die Standardeinstellung ist „false“.	--sv-exome	true/false

Name	Beschreibung	Befehlszeilenentsprechung	Wert
sv-output-contigs	Bei Festlegung auf „true“ werden assemblierte Contig-Sequenzen in einer VCF-Datei ausgegeben. Die Standardeinstellung ist „false“.	--sv-output-contigs	true/false
sv-quiet	Bei Festlegung auf „true“ wird die Protokollausgabe auf stderr verhindert. (Es wird jedoch weiterhin eine entsprechende Protokolldatei ausgegeben.) Die Standardeinstellung ist „true“.	--sv-quiet	true/false
sv-reference	Gibt eine Referenzdatei im FASTA-Format an.	--sv-reference	
sv-region	Schränkt für das Debugging die Analyse auf eine festgelegte Region des Genoms ein. Kann wiederholt angegeben werden, um eine Liste mit Regionen zu erstellen.	--sv-region	Muss in folgendem Format eingegeben werden: „chr:startPos-endPos“.
variant	Der Pfad zu einer einzelnen gVCF-Datei. In der Befehlszeile können mehrere *--variant*-Optionen verwendet werden, eine für jede gVCF-Datei. Es werden bis zu 500 gVCF-Dateien unterstützt.	--variant	
variant-list	Der Pfad zu einer Datei, die eine Liste der gVCF-Eingabedateien enthält (mit einer Datei pro Zeile), die kombiniert werden müssen.	--variant-list	
vc-alt-allele-in-normal-filter	Legen Sie diese Option auf „false“ fest, um den Filter „alt-allele-in-normal“ zu deaktivieren. Die Standardeinstellung ist „true“.	--vc-alt-allele-in-normal-filter	true/false
vc-decoy-contigs	Der Pfad zu einer kommagetrennten Liste mit Contigs, die während des Varianten-Callings übersprungen werden sollen.	--vc-decoy-contigs	
vc-emit-ref-confidence	Zur Aktivierung der gVCF-Erstellung für Basenpaare auf BP_RESOLUTION festlegen. Zur Aktivierung von gVCF-Erstellung mit Banding auf GVCF festlegen.	--vc-emit-ref-confidence	BP_RESOLUTION/GVCF
vc-enable-baf	Aktiviert bzw. deaktiviert die Ausgabe der B-Allelfrequenz. Die Standardeinstellung ist „true“ (aktiviert).	--vc-enable-baf	
vc-enable-clustered-events-filter	Aktiviert/deaktiviert den Clusterereignisfilter. Die Standardeinstellung ist „true“.	--vc-enable-clustered-events-filter	true/false
vc-enable-decoy-contigs	Aktiviert/deaktiviert Varianten-Calls bei Decoy-Contigs. Die Standardeinstellung ist „false“.	--vc-enable-decoy-contigs	true/false
vc-enable-gatk-acceleration	Aktiviert/deaktiviert die Ausführung des Varianten-Callers im GATK-Modus.	--vc-enable-gatk-acceleration	true/false
vc-enable-homologous-mapping-filter	Aktiviert/deaktiviert den Ereignisfilter für homologes Mapping. Die Standardeinstellung ist „true“.	--vc-enable-homologous-mapping-filter	true/false

Name	Beschreibung	Befehlszeilenentsprechung	Wert
vc-enable-orientation-bias-filter	Aktiviert/deaktiviert den Ausrichtungsverzerrungsfilter.	--vc-enable-orientation-bias-filter	true/false
vc-enable-phasing	Aktiviert die Phasierung von Varianten, sofern möglich. Die Standardeinstellung ist „true“.	--vc-enable-phasing	true/false
vc-enable-roh	Aktiviert bzw. deaktiviert ROH-Caller und -Ausgabe. Die Standardeinstellung ist „true“ (aktiviert).	--vc-enable-roh	
vc-enable-triallelic-filter	Aktiviert den Filter für mehrere Allele. Die Standardeinstellung ist „true“.	-vc-enable-triallelic-filter	
vc-forcegt-vcf	Erzwingt die Genotypisierung für das Keimbahn-Calling kleiner Varianten. Es ist eine .vcf- oder .vcf.gz-Datei mit einer Liste kleiner Varianten erforderlich.	--vc-forcegt-vcf	Eine vcf- oder .vcf.gz-Datei mit kleinen Varianten, für die die Genotypisierung erzwungen werden soll.
vc-frd-beta-max	Die maximale Allelfrequenz für die Foreign Read-Hypothese.	--vc-frd-beta-max	
vc-gvcf-gq-bands	Definiert GQ-Folgen für die gVCF-Ausgabe. Die Standardwerte sind 10, 20, 30, 40, 60, 80.	--vc-gvcf-gq-bands	
vc-hard-filter	Boolescher Ausdruck für die Filterung von Varianten-Calls. Der Standardausdruck lautet: DRAGENHardQUAL:all: QUAL < 10.4139;LowDepth:all: DP < 1		Mögliche Parameter im Ausdruck: QD, MQ, FS, MQRankSum, ReadPosRankSum, QUAL, DP und GQ.
vc-limit-genomecov-output	Beschränkt die Größe der erstellten .genomecov.bed-Datei. Die Standardeinstellung ist „false“.	--vc-limit-genomecov-output	true/false
vc-max-alternate-alleles	Die maximale Anzahl der ALT-Allele, die in einer VCF- oder gVCF-Datei ausgegeben werden können. Der Standardwert ist 6.	--vc-max-alternate-alleles	
vc-max-reads-per-active-region	Die maximale Anzahl der Reads pro aktiver Region zum Downsampling. Der Standardwert ist 1000.	--vc-max-reads-per-active-region	
vc-max-reads-per-active-region-mito	Die maximale Anzahl der Reads, die eine bestimmte aktive Region abdecken, für das Mitochondrien-Calling kleiner Varianten.	--vc-max-reads-per-active-region-mito	
vc-max-reads-per-raw-region	Die maximale Anzahl der Reads pro Rohregion zum Downsampling. Der Standardwert ist 1000.	--vc-max-reads-per-raw-region	
vc-max-reads-per-raw-region-mito	Die maximale Anzahl der Reads, die eine bestimmte Rohregion abdecken, für das Mitochondrien-Calling kleiner Varianten.	--vc-max-reads-per-raw-region-mito	
vc-min-base-qual	Minimal zulässige Basenqualität für das Varianten-Calling. Der Standardwert ist 10.	--vc-min-base-qual	

Name	Beschreibung	Befehlszeilenentsprechung	Wert
vc-min-call-qual	Minimal zulässige Varianten-Call-Qualität für die Ausgabe eines Calls. Der Standardwert ist 30.	--vc-min-call-qual	
vc-min-read-qual	Minimal zulässige Read-Qualität (MAPQ) für das Varianten-Calling. Der Standardwert ist 20.	--vc-min-read-qual	
vc-min-reads-per-start-pos	Die minimale Anzahl der Reads pro Startposition für das Downsampling. Der Standardwert ist 5.	--vc-min-reads-per-start-pos	
vc-min-tumor-read-qual	Minimal zulässige Tumor-Read-Qualität (MAPQ) für das Varianten-Calling.		
vc-orientation-bias-filter-artifacts	Zu filternder Artefakttyp. Ein Artefakt (bzw. ein Artefakt und sein umgekehrtes Komplement) kann nur einmal aufgeführt werden.	--vc-orientation-bias-filter-artifacts	C/T, G/T oder C/T, G/T, C/A
vc-remove-all-soft-clips	Ist diese Option auf „true“ festgelegt, werden die Varianten vom Varianten-Caller nicht anhand von Reads mit Soft Clipping bestimmt.	--vc-remove-all-soft-clips	true/false
vc-roh-blacklist-bed	Blacklist-BED-Datei für ROH.	--vc-roh-blacklist-bed	
vc-target-bed	BED-Datei mit Zielregionen.	--vc-target-bed	
vc-target-bed-padding	Kann verwendet werden, um alle BED-Zielregionen mit dem festgelegten Wert aufzufüllen (optional). Wird, sofern festgelegt, vom Caller für kleine Varianten verwendet.	--vc-target-bed-padding	
vc-target-coverage	Ziel-Coverage für das Downsampling. Der Standardwert ist 2000.	--vc-target-coverage	
vc-target-coverage-mito	Die maximale Anzahl der Reads mit einer Startposition, die mit einer bestimmten Position für das Mitochondrien-Calling kleiner Varianten überlappt.	--vc-target-coverage	
vc-tlod-filter-threshold	Legt den Schwellenwert für den TLOD-Filter fest Der Standardwert ist 6.5.	--vc-tlod-filter-threshold	
vqsr-annotation	Eine kommagetrennte Zeichenfolge, die die für das Erstellen von Modellen zu verwendenden Annotationen festlegt. Das Format der Zeichenfolge ist *<Modus>,<Annotation>,<Annotation>...*, wobei „Modus“ für INDEL oder SNP steht. Weitere Informationen finden Sie unter *Variant Quality Score Recalibration* auf Seite 102.	--vqsr-annotation	
vqsr-config	Der Pfad zur VQSR-Konfigurationsdatei.	--vqsr-config	
vqsr-filter-level	Gibt für die Filterung von Varianten-Calls das Sensitivitätslevel für die Echtheit in Prozent an.	--vqsr-filter-level	
vqsr-input	Legt die zu verarbeitende VCF-Eingabedatei für VQSR fest.	--vqsr-input	

Name	Beschreibung	Befehlszeilenentsprechung	Wert
vqsr-lod-cutoff	Gibt den LOD-Schwellenwert für die Auswahl der Varianten-Call-Stellen an, die für die Erstellung des negativen Modells zu verwenden sind. Der Standardwert ist -5.0.	--vqsr-lod-cutoff	
vqsr-num-gaussians	Die Anzahl der Gaußschen Normalverteilungen zur Erstellung der positiven und negativen Modelle, die als kommagetrennte Zeichenfolge der folgenden vier ganzzahligen Werte angegeben ist: <SNP positiv>,<SNP negativ>,<INDEL positiv>,<INDEL negativ>. Sofern nicht anders angegeben, lauten die Standardwerte 8, 2, 4, 2.	--vqsr-num-gaussians	
vqsr-resource	Legt die Trainingsressourcendateien fest, mit deren Hilfe echte Varianten-Call-Stellen bestimmt werden. Mit dieser Option werden der Betriebsmodus, der Wert für die A-priori-Wahrscheinlichkeit zum Gewichten dieser Ressource und schließlich der Pfad der Ressourcendatei als kommagetrennte Zeichenfolge angegeben. Beispiel: <Modus>,<A-priori>,<Ressourcendatei>.	--vqsr-resource	
vqsr-tranche	Gibt für die Berechnung der LOD-Schwellenwerte in Prozent die Sensitivitätslevel für die Echtheit an. Diese Option kann mehrfach mit jeweils unterschiedlichem Sensitivitätslevel angegeben werden. Sofern nicht anders festgelegt, lauten die Standardwerte 100.0, 99.99, 99.90, 99.0 und 90.0.	--vqsr-tranche	

Optionen für die Repeat-Expansion-Bestimmung

Die folgenden Optionen können im Abschnitt RepeatGenotyping der Konfigurationsdatei oder in der Befehlszeile festgelegt werden. Weitere Informationen finden Sie unter *Repeat-Expansion-Bestimmung mit Expansion Hunter* auf Seite 73.

Parametername	Beschreibung	Befehlszeilenentsprechung	Bereich
enable	Zur Aktivierung bzw. Deaktivierung der Repeat-Expansion-Bestimmung.	--repeat-genotype-enable	true/false
specs	Der vollständige Pfad zur JSON-Datei, die den Repeat-Variantenkatalog (Spezifikation) mit der Beschreibung der Loci für das Calling enthält.	--repeat-genotype-specs	

Technische Unterstützung

Wenn Sie technische Unterstützung benötigen, wenden Sie sich bitte an den technischen Support von Illumina.

Website: www.illumina.com
E-Mail: techsupport@illumina.com

Telefonnummern des Illumina-Kundendiensts

Region	Gebührenfrei	Regional
Nordamerika	+1.800.809.4566	
Australien	+1.800.775.688	
Belgien	+32 80077160	+32 34002973
China	400.066.5835	
Dänemark	+45 80820183	+45 89871156
Deutschland	+49 8001014940	+49 8938035677
Finnland	+358 800918363	+358 974790110
Frankreich	+33 805102193	+33 170770446
Großbritannien	+44 8000126019	+44 2073057197
Hongkong, China	800960230	
Irland	+353 1800936608	+353 016950506
Italien	+39 800985513	+39 236003759
Japan	0800.111.5011	
Neuseeland	0800.451.650	
Niederlande	+31 8000222493	+31 207132960
Norwegen	+47 800 16836	+47 21939693
Österreich	+43 800006249	+43 19286540
Schweden	+46 850619671	+46 200883979
Schweiz	+41 565800000	+41 800200442
Singapur	1.800.579.2745	
Spanien	+34 911899417	+34 800300143
Südkorea	+82 80 234 5300	
Taiwan, China	00806651752	
Andere Länder	+44.1799.534000	

Sicherheitsdatenblätter (SDS, Safety Data Sheets) sind auf der Illumina-Website unter support.illumina.com/sds.html verfügbar.

Die **Produktdokumentation** steht unter support.illumina.com zum Herunterladen zur Verfügung.

Dokument-Nr. 1000000112459 v00 DEU

Illumina
5200 Illumina Way
San Diego, Kalifornien 92122, USA
+1.800.809.ILMN (4566)
+1.858.202.4566 (außerhalb von Nordamerika)
techsupport@illumina.com
www.illumina.com

Nur für Forschungszwecke. Nicht zur Verwendung in Diagnoseverfahren.