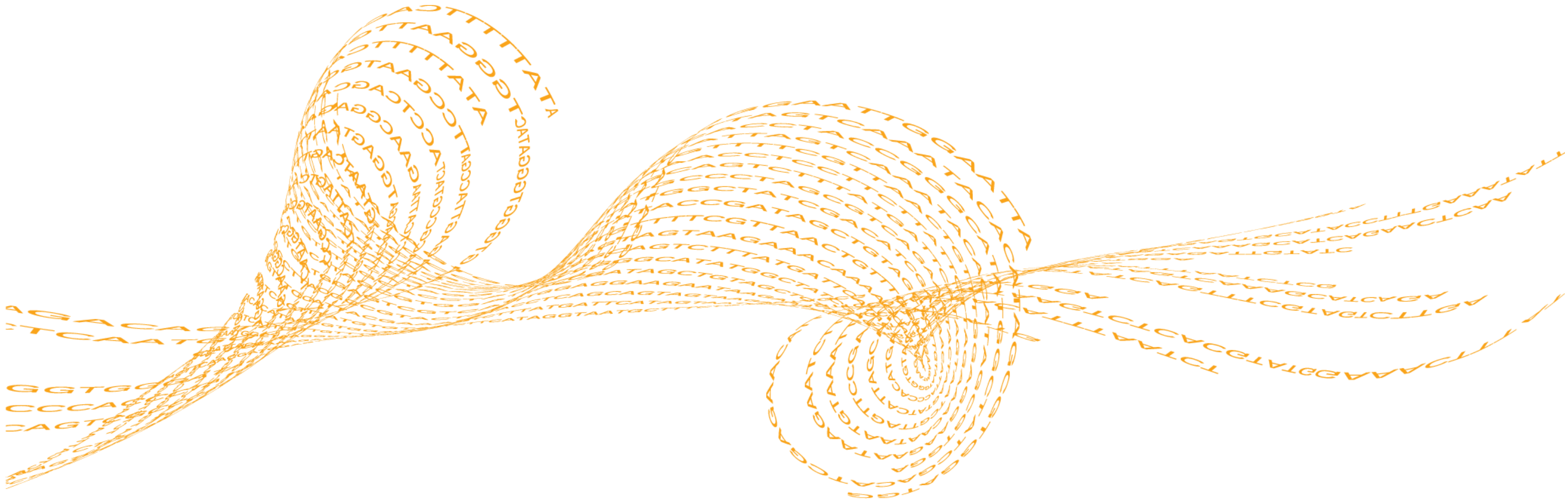
RNA Sequencing with TopHat and Cufflinks



ILLUMINA PROPRIETARY
15050962 Rev. A
January 2014

This document and its contents are proprietary to Illumina, Inc. and its affiliates ("Illumina"), and are intended solely for the contractual use of its customer in connection with the use of the product(s) described herein and for no other purpose. This document and its contents shall not be used or distributed for any other purpose and/or otherwise communicated, disclosed, or reproduced in any way whatsoever without the prior written consent of Illumina. Illumina does not convey any license under its patent, trademark, copyright, or common-law rights nor similar rights of any third parties by this document.

The instructions in this document must be strictly and explicitly followed by qualified and properly trained personnel in order to ensure the proper and safe use of the product(s) described herein. All of the contents of this document must be fully read and understood prior to using such product(s).

FAILURE TO COMPLETELY READ AND EXPLICITLY FOLLOW ALL OF THE INSTRUCTIONS CONTAINED HEREIN MAY RESULT IN DAMAGE TO THE PRODUCT(S), INJURY TO PERSONS, INCLUDING TO USERS OR OTHERS, AND DAMAGE TO OTHER PROPERTY.

ILLUMINA DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE IMPROPER USE OF THE PRODUCT(S) DESCRIBED HEREIN (INCLUDING PARTS THEREOF OR SOFTWARE) OR ANY USE OF SUCH PRODUCT(S) OUTSIDE THE SCOPE OF THE EXPRESS WRITTEN LICENSES OR PERMISSIONS GRANTED BY ILLUMINA IN CONNECTION WITH CUSTOMER'S ACQUISITION OF SUCH PRODUCT(S).

FOR RESEARCH USE ONLY

© 2014 Illumina, Inc. All rights reserved.

Illumina, **IlluminaDx**, **BaseSpace**, **BeadArray**, **BeadXpress**, **cBot**, **CSPro**, **DASL**, **DesignStudio**, **Eco**, **GAIIx**, **Genetic Energy**, **Genome Analyzer**, **GenomeStudio**, **GoldenGate**, **HiScan**, **HiSeq**, **Infinium**, **iScan**, **iSelect**, **MiSeq**, **MiSeqDx**, **Nextera**, **NextSeq**, **NuPCR**, **SeqMonitor**, **Solexa**, **TruGenome**, **TruSeq**, **TruSight**, **Understand Your Genome**, **UYG**, **VeraCode**, the pumpkin orange color, and the Genetic Energy streaming bases design are trademarks of Illumina, Inc. in the U.S. and/or other countries. All other names, logos, and other trademarks are the property of their respective owners.

Introduction

The BaseSpace apps *TopHat Alignment* and *Cufflinks Assembly and Differential Expression* are developed for analysis of RNA sequencing data. The main components of the TopHat and Cufflinks apps are derived from the Tuxedo suite. The two apps are designed to perform the following steps:

- The TopHat app runs Bowtie, TopHat, the Isaac Variant Caller, and Cufflinks. The app produces aligned reads, variant calls and FPKM abundance estimates of reference genes and transcripts. See *TopHat App Workflow* on page 27 for more information.
- The *Cufflinks Assembly and Differential Expression App* uses previous alignment results produced by the TopHat app as input, and performs novel transcript assembly and/or differential expression analysis. See *Cufflinks App Workflow* on page 28 for more information.

Versions

The following module versions are used in the TopHat and Cufflinks apps:

- TopHat2 v2.0.7
- Bowtie 0.12.9
- Cufflinks 2.1.1
- Isaac Variant Caller 2.0.5
- Picard tools 1.72

Current Limitations

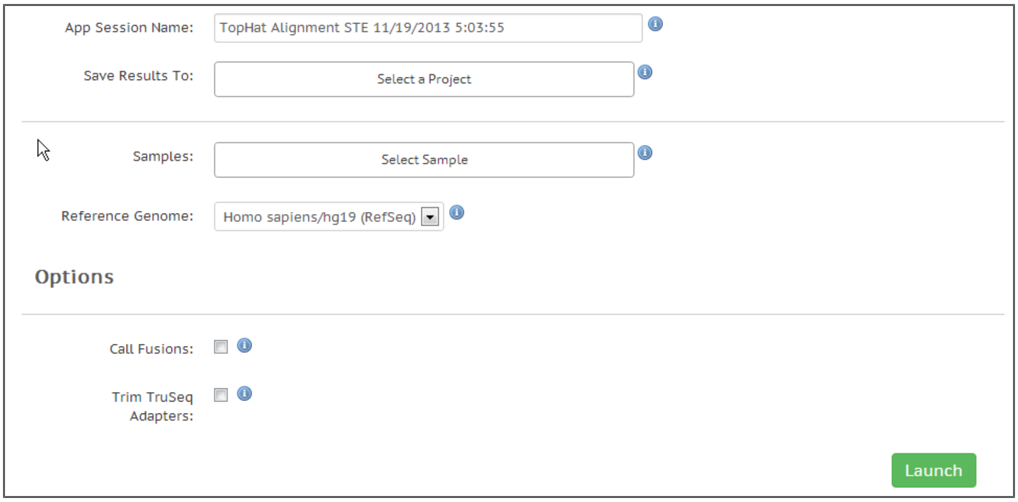
Before running the TopHat app, be aware of the following limitations:

- Reads must be at least 35 bp and no more than 125 bp in length.
- An individual sample should have between 100,000 and 400 million reads.
- Only UCSC hg19 (human), UCSC mm10 (mouse), and UCSC rn5 (rat) are currently supported.

Run TopHat App

1. Navigate to the project or sample that you want to analyze.
2. Click the **Launch app** button and select **TopHat Alignment** from the dropdown list.
3. Read the End User License Agreement and Permissions, and click **Accept** if you agree.
4. Set up the TopHat analysis:
    a. **App Session Name**: provide the app session name. Default name is the app name with the date and time the app session was started.
    b. **Save Results To**: select the project that stores the app results.
    c. **Samples**: browse to the sample you want to analyze, and select the check box. You can analyze multiple samples.
    d. Identify the samples prepared with a stranded sample kit by selecting the **Stranded** checkbox. Samples prepared with the TruSeq Stranded Total RNA Sample Prep Kit and TruSeq Stranded mRNA Sample Prep Kit are stranded.
    e. **Reference Genome**: select the reference genome.
5. Fill out the Options:
    a. **Call Fusions**: If selected, TopHat-fusion is used to detect gene fusions.
    b. **Trim TruSeq Adapters**: If selected, the application attempts to trim TruSeq adapters from the FASTQ sequence. Typically, this trimming will be unnecessary as adapter trimming is performed as part of demultplexing during sample upload. However, if the user did not specify adapter sequences in the sample sheet during upload, this provides a second opportunity to trim the adapters.

Figure 1 TopHat Input Form



App Session Name: TopHat Alignment STE 11/19/2013 5:03:55

Save Results To: Select a Project

Samples: Select Sample

Reference Genome: Homo sapiens/hg19 (RefSeq)

Options

Call Fusions: ☐

Trim TruSeq Adapters: ☐

6. Click **Launch**.

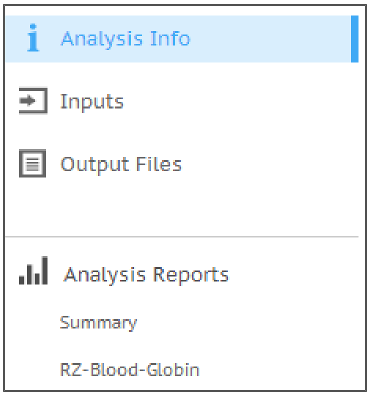
The TopHat app will now analyze your sample. When completed, the status of the app session is automatically updated, and you receive an email.

NOTE
Depending on the size and number of samples, the complete analysis may take between a few hours and several days.

TopHat App Output

This chapter describes the output that is produced by the TopHat app. To go to the results, click the **Projects** button, then the project, then the analysis.

Figure 2 TopHat Output Navigation Bar



When the analysis is completed, you can access your output through the left navigation bar, which provides the following:

- **Analysis Info**: an overview of the analysis. See *Analysis Info* on page 22 for a description.
- **Inputs**: an overview of the input samples and settings. See *TopHat Inputs Overview* on page 11 for a description.
- **Output Files**: access to the output files, organized by sample and app session. See *TopHat Output Files* on page 11 for descriptions.
- **Analysis Reports**:
  - **Summary**: access to analysis metrics for the aggregate results. See *Analysis Reports: Summary* on page 5 for a description.
  - **Sample Pages**: access to analysis metrics for each sample. See *Analysis Reports: Sample Pages* on page 7 for a description.

Analysis Reports: Summary

The TopHat app provides an overview for all samples on the Summary page. A brief description of the metrics is below.

Alignment Statistics

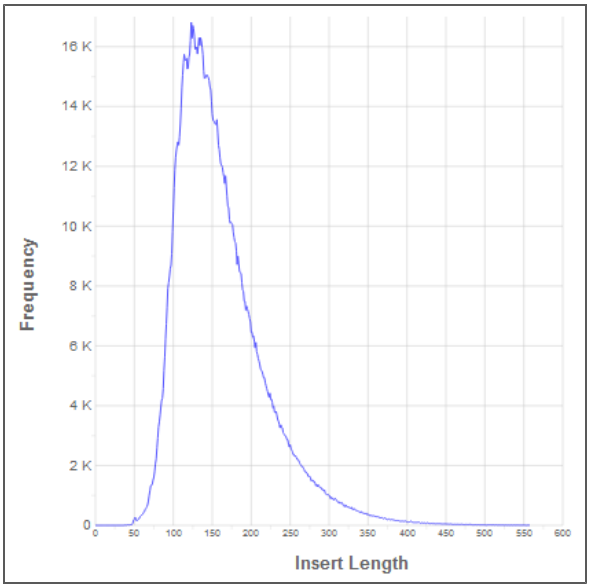
Statistic	Definition
Reads	Number and length of reads.
Number of Reads	Total number of reads passing filter for this sample.
% Total Aligned	Percentage of reads passing filter that aligned to the reference, including abundant reads.
% Abundant	Percentage of reads that align to abundant transcripts, such as mitochondrial and ribosomal sequences.

Statistic	Definition
% Unaligned	Percentage of passed filter reads that do not align to the reference.
Median CV Coverage Uniformity	The median coefficient of variation of coverage of the 1000 most highly expressed transcripts, as reported by the CollectRnaSeqMetrics utility from Picard tools. Ideal value = 0.
% Stranded	Percentage of reads that align to the correct strand, as reported by the CollectRnaSeqMetrics utility from Picard tools.

Insert Length Distribution

The Insert Length Distribution graphs the frequency of each insert length. This histogram is only available for paired-end reads.

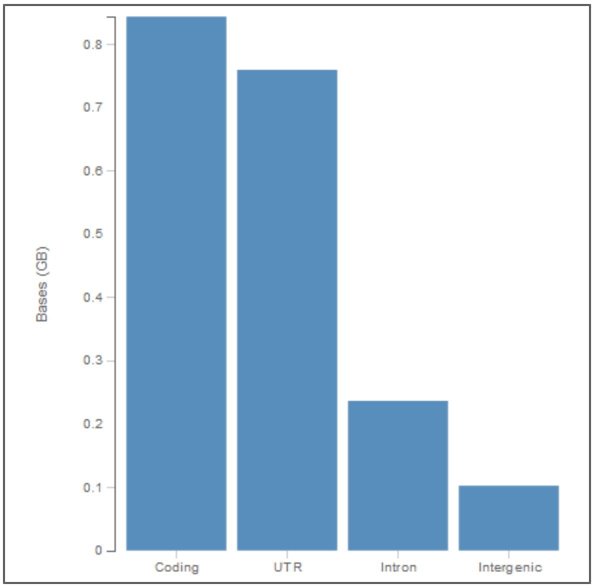
Figure 3 Insert Length Distribution



Alignment Distribution

The Alignment Distribution histogram graphs the number of bases by genomic region (coding, UTR, intron, or intergenic).

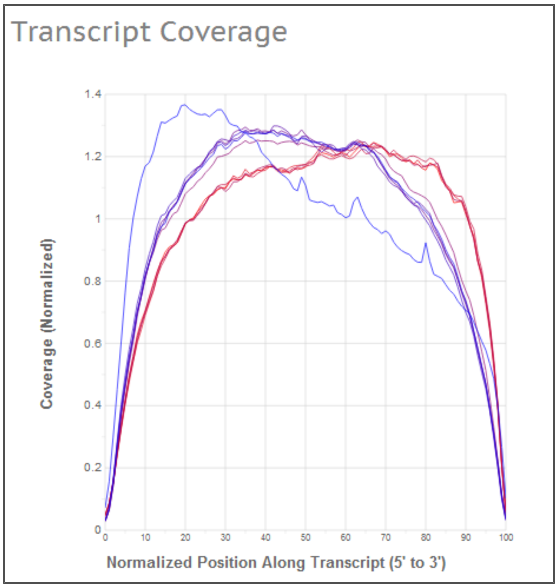
Figure 4 Alignment Distribution Histogram



Transcript Coverage

The Transcript Coverage graphs the coverage of a transcript position by that transcript position as reported by the CollectRnaSeqMetrics utility from Picard tools. The coverage is normalized for the coverage of that transcript. You can get the series names from the tooltips by hovering over the graph.

Figure 5 Transcript Coverage



Analysis Reports: Sample Pages

The TopHat app provides an overview of statistics per sample on the sample pages. A brief description of the metrics is below.

Primary Analysis Information

Statistic	Definition
Read Length	Length of reads.
Number of Reads	Total number of reads passing filter for this sample.
Bases (GB)	The total number of bases for this sample.
Q30 Bases (GB)	The total number of bases with a quality score of 30 or higher.

Insert Information

Only available for paired-end reads.

Statistic	Definition
Insert Length Median	Median length of the sequenced fragment. The fragment length is calculated based on the locations at which a read pair aligns to the reference. The read mapping information is parsed from the BAM files.
Insert Length S.D.	Standard deviation of the sequenced fragment length.
Duplicates (% Reads)	Percentage of paired reads that have duplicates., from a sub-sampled set of 4 M reads.

Alignment Quality

Statistic	Definition
Total Aligned Reads (% Reads)	Percentage of reads passing filter that align to the reference.
Abundant Reads (% Reads)	Percentage of reads that align to abundant transcripts, such as mitochondrial and ribosomal sequences.
Unaligned Reads (% Reads)	Percentage of reads passing filter that are not aligned to the reference.
Reads with spliced alignment (% Aligned Reads)	Percentage of aligned reads with a spliced alignment.
Reads aligned at multiple loci (% Aligned Reads)	Percentage of aligned reads to multiple loci.

Coverage Uniformity

Statistic	Definition
Median CV	The median coefficient of variation of coverage of the 1000 most highly expressed transcripts, as reported by the CollectRnaSeqMetrics utility from Picard tools. Ideal value = 0.

Statistic	Definition
Median 3'	The median uniformity of coverage of the 1000 most highly expressed transcripts at the 3' end, as reported by the CollectRnaSeqMetrics utility from Picard tools. 3' bias is calculated per transcript as: mean coverage of the 3' most 100 bases divided by the mean coverage of the whole transcript.
Median 5'	The median uniformity of coverage of the 1000 most highly expressed transcripts at the 5' end, as reported by the CollectRnaSeqMetrics utility from Picard tools. 5' bias is calculated per transcript as: mean coverage of the 5' most 100 bases divided by the mean coverage of the whole transcript.
Reads aligned to correct strand	Percentage of reads that align to the correct strand, as reported by the CollectRnaSeqMetrics utility from Picard tools.

Alignment Information

This table presents metrics for the categories in two columns:

- Fold Coverage: The total number of bases in the category divided by the size of the entire category.
- % Bases: The total number of bases aligned to this region relative to the total number of aligned bases.

These metrics are reported for the following categories, derived from RefSeq:

Statistic	Definition
Coding	Metrics based on coding bases.
UTR	Metrics based on bases in untranslated regions (UTR).
Intron	Metrics based on bases in introns.
Intergenic	Metrics based on bases in intergenic regions.

Variant Calls

Statistic	Definition
Homozygous Reference	Number of homozygous reference calls.
Heterozygous	Number of heterozygous variant calls.
Homozygous variant	Number of homozygous variant calls.
SNV	Total number of SNVs detected for the sample.
Indel	The number of indels detected for the sample.
T_n/T_v	Transition rate of SNVs that pass the quality filters divided by transversion rate of SNVs that pass the quality filters. Transitions are interchanges of purines (A, G) or of pyrimidines (C, T). Transversions are interchanges of purine and pyrimidine bases (for example, A to T).

Histograms

The sample pages also contain three histograms that are described in the following topics:

- *Insert Length Distribution* on page 6 (only available for paired-end reads).
- *Alignment Distribution* on page 6
- *Transcript Coverage* on page 7

Important Files

The Important Files section provides direct access to the most important output files. For detailed descriptions of these files, see:

- *BAM Files* on page 12
- *VCF Files* on page 12
- *gVCF Files* on page 12
- *FPKM Files* on page 24
- *Candidate Fusion List* on page 17

Analysis Info

This app provides an overview of the analysis on the Analysis Info page. A brief description of the metrics is below.

Row	Definition
Name	Name of the app session.
Application	App that generated this analysis.
Date started	Date the app session started.
Date completed	Date the app session completed.
Duration	Duration of analysis.
Status	Status of the app session.

Log Files

Clicking on the **Log Files** link at the bottom of the Analysis Info page provides access to TopHat and Cufflinks app log files.

The following files log information to help follow data processing and debugging:

- **WorkflowLog.txt**: Workflow standard output (contains details about workflow steps, command line calls with parameters, timing and progress).
- **WorkflowError.txt**: Workflow standard error output (contains errors messages created while running the workflow).
- **Logging.zip**: Contains all detailed workflow log files for each step of the workflow (content from Isis Logging folder)
- **IlluminaAppsService.log.copy**: Wrapper log file containing information about communication (get and post requests) between BaseSpace and AWS.

The following files contain additional information in case things (like mono) do not work as expected:

- **monoErr.txt**: Wrapper mono call error output (contains anything that is not caught by the WorkflowError.txt and will on most cases be empty, except one line).
- **monoOut.txt**: Wrapper mono call standard output (contains command calling the workflow and anything that is not caught by the WorkflowLog.txt).

NOTE
For explanation about mono, see www.mono-project.com.

TopHat Status

The status of the TopHat app session can have the following values:

- Preparing data
- Aligning
- Variant calling
- Estimating reference abundances
- Calculating metrics
- Finalizing results

Depending on the size and number of samples, the complete analysis may take between a few hours and several days.

TopHat Inputs Overview

The TopHat app provides an overview of the input samples and settings on the Inputs page. A brief description of the metrics is below.

Statistic	Definition
App Session Name	Name of the app session.
Save Results To	The project that stores the app results.
Samples	Samples selected for this analysis. Checkbox indicates if the samples are stranded.
Reference Genome	Reference genome and gene model selected.
Call Fusions	If selected, TopHat-fusion is used to detect gene fusions.
Trim TruSeq Adapters	If selected, the application trims TruSeq adapters from the FASTQ sequence.

TopHat Output Files

This section describes the main output files of the TopHat App:

- *BAM Files* on page 12
- *VCF Files* on page 12
- *gVCF Files* on page 12
- *FPKM Files* on page 24
- *Coverage.BedGraph.gz Files* on page 16
- *Junctions.bed Files* on page 16
- *Candidate Fusion List* on page 17

The output files are located in the following locations:

`<Analysis Result>|filtered`

- <sample name>.abundant_alignments.bam – Alignments of the reads against abundant sequences.

`<Analysis Result>|alignments`

- <sample name>.alignments.bam – Alignments of filtered reads against the genome (and transcriptome).
- <sample. name>.coverage.bedGraph.gz – Genome coverage from aligned RNA-seq reads
- <sample name>.junctions.bed – BED file describing junctions discovered in this data set in TopHat format.

`<Analysis Result>|cufflinks|reference`

- <sample name>.isoforms.fpkm_tracking – Abundance estimates for individual transcripts defined in genome reference.
- <sample name>.genes.fpkm_tracking – Abundance estimates for genes defined in genome reference.

`<Analysis Result>|metrics:`

- Contains CSV and json metrics files used to generate the summary and individual reports.

`<Analysis Result>|tophat_fusion`

- result.html – HTML summary page describing the discovered fusions.

`<Analysis Result>|variants`

- <sample_id>.genome.vcf.gz – Variant calls from starling in gVCF format.

BAM Files

The Sequence Alignment/Map (SAM) format is a generic alignment format for storing read alignments against reference sequences, supporting short and long reads (up to 128 Mb) produced by different sequencing platforms. SAM is a text format file that is human-readable. The Binary Alignment/Map (BAM) keeps exactly the same information as SAM, but in a compressed, binary format that is only machine-readable.

If you use an app in BaseSpace that uses BAM files as input, the app will locate the file when launched. If using BAM files in other tools, download the file to use it in the external tool.

Go to http://samtools.sourceforge.net/SAM1.pdf to see the exact SAM specification.

VCF Files

VCF is a text file format which contains information about variants found at specific positions in a reference genome. The file format consists of meta-information lines, a header line, and then data lines. Each data line contains information about a single variant.

If you use an app in BaseSpace that uses VCF files as input, the app will locate the file when launched. If using VCF files in other tools, download the file to use it in the external tool.

A detailed description of the VCF format is provided in the *BaseSpace User Guide*.

gVCF Files

This application also produces the genome Variant Call Format file (gVCF). gVCF was developed to store sequencing information for both variant and non-variant positions, which is required for human clinical applications. gVCF is a set of conventions applied to the standard variant call format (VCF) 4.1 as documented by the 1000 Genomes Project. These conventions allow representation of genotype, annotation, and other information across all sites in the genome in a compact format. Typical human whole genome sequencing results expressed in gVCF with annotation are less than 1 Gbyte, or about 1/100 the size of the BAM file used for variant calling. If you are performing targeted sequencing, gVCF is also an appropriate choice to represent and compress the results.

gVCF is a text file format, stored as a gzip compressed file (*.genome.vcf.gz). Compression is further achieved by joining contiguous non-variant regions with similar properties into single 'block' VCF records. To maximize the utility of gVCF, especially for high stringency applications, the properties of the compressed blocks are conservative -- thus block properties like depth and genotype quality reflect the minimum of any site in the block. The gVCF file can be indexed (creating a .tbi file) and used with existing VCF tools such as tabix and IGV, making it convenient both for direct interpretation and as a starting point for tertiary analysis.

For more information, see https://sites.google.com/site/gvcftools/home/about-gvcf.

The following conventions are used in the variant caller gVCF files.

Samples per File

There is only one sample per gVCF file.

Non-Variant Blocks Using END Key

Contiguous non-variant segments of the genome can be represented as single records in gVCF. These records use the standard 'END' INFO key to indicate the extent of the record. Even though the record can span multiple bases, only the first base is provided in the REF field to reduce file size.

The following is a simplified segment of a gVCF file, describing a segment of non-variant calls (starting with an A) on chromosome 1 from position 51845 to 51862.

```
##INFO=<ID=END,Number=1,Type=Integer,Description="End position
   of the variant described in this record">#CHROM POS ID REF
   ALT QUAL FILTER INFO FORMAT NA19238chr1 51845 . A . . PASS
   END=51862
```

Any fields provided for a block of sites, such as read depth (using the DP key), will show the minimum value observed among all sites encompassed by the block. Each sample value shown for the block, such as the depth (using the DP key), is restricted to a range where the maximum value is within 30% or 3 of the minimum, i.e. for sample value range [x,y], y <= x+max(3,x*0.3). This range restriction applies to each of the sample values printed out in the final block record.

Indel Regions

Note that sites which are "filled in" inside of deletions have additional changes:

All deletions:

- Sites inside of any deletion are marked with the deletion's filters, in addition to any filters which have already been applied to the site.
- Sites inside of deletions cannot have a genotype or alternate allele quality score higher than the corresponding value from the enclosing indel.

Heterozygous deletions:

- Sites inside of heterozygous deletions are altered to have haploid genotype entries (e.g. "0" instead of "0/0", "1" instead of "1/1").
- Heterozygous SNV calls inside of heterozygous deletions are marked with the "SiteConflict" filter and their genotype is unchanged.

Homozygous deletions:

- Homozygous reference and no-call sites inside of homozygous deletions have genotype "."
- Sites inside of homozygous deletions which have a non-reference genotype are marked with a “SiteConflict” filter, and their genotype is unchanged.
- Site and genotype quality are set to "."

The above modifications reflect the notion that the site confidence is bound by the enclosing indel confidence.

Also note that on occasion, the variant caller will produce multiple overlapping indel calls which cannot be resolved into two haplotypes. If this occurs all indels and sites in the region of the overlap will be marked with the “IndelConflict” filter (see below).

Genotype Quality for Variant and Non-variant Sites

The gVCF file uses an adapted version of genotype quality for variant and non-variant site filtration. This value is associated with the key GQX. The GQX value is intended to represent the minimum of {Phred genotype quality assuming the site is variant, Phred genotype quality assuming the site is non-variant}. The reason for using this is to allow a single value to be used as the primary quality filter for both variant and non-variant sites. Filtering on this value corresponds to a conservative assumption appropriate for applications where reference genotype calls must be determined at the same stringency as variant genotypes, i.e.:

- An assertion that a site is homozygous reference at GQX >= 30 is made assuming the site is variant.
- An assertion that a site is a non-reference genotype at GQX >= 30 is made assuming the site is non-variant.

Section Descriptions

The gVCF file contains the following sections:

- Meta-information lines start with ## and contain meta-data, config information, and define the values that the INFO, FILTER and FORMAT fields can have.
- The header line starts with # and names the fields that the data lines use. These are #CHROM, POS, ID,REF, ALT, QUAL, FILTER, INFO, FORMAT, followed by one or more sample columns.
- Data lines that contain information about one or more positions in the genome.

Note that if you extract the variant lines from a gVCF file, you produce a conventional variant VCF file.

Field Descriptions

The fixed fields #CHROM, POS, ID, REF, ALT, QUAL are defined in the VCF 4.1 standard provided by the 1000 Genomes Project, while the fields ID, INFO, FORMAT, and sample are described in the meta-information. Descriptions are provided below.

- **CHROM**: Chromosome: an identifier from the reference genome or an angle-bracketed ID String ("<ID>") pointing to a contig.
- **POS**: Position: The reference position, with the 1st base having position 1. Positions are sorted numerically, in increasing order, within each reference sequence CHROM. There can be multiple records with the same POS. Telomeres are indicated by using positions 0 or N+1, where N is the length of the corresponding chromosome or contig.
- **ID**: Semi-colon separated list of unique identifiers where available. If this is a dbSNP variant it is encouraged to use the rs number(s). No identifier should be present in more than one data record. If there is no identifier available, then the missing value should be used.
- **REF**: Reference base(s): A,C,G,T,N; there can be multiple bases. The value in the POS field refers to the position of the first base in the string. For simple insertions and deletions in which either the REF or one of the ALT alleles would otherwise be null/empty, the REF and ALT strings include the base before the event (which is reflected in the POS field), unless the event occurs at position 1 on the contig in which case they include the base after the event. If any of the ALT alleles is a symbolic allele (an angle-bracketed ID String "<ID>") then the padding base is required and POS denotes the coordinate of the base preceding the polymorphism.
- **ALT**: Comma separated list of alternate non-reference alleles called on at least one of the samples. Options are:
  - Base strings made up of the bases A,C,G,T,N
  - angle-bracketed ID String (”<ID>”)
  - breakend replacement string as described in the section on breakends.

  If there are no alternative alleles, then the missing value should be used.
- **QUAL**: Phred-scaled quality score for the assertion made in ALT. i.e. -10log_10 prob (call in ALT is wrong). If ALT is ”.” (no variant) then this is -10log_10 p(variant), and if ALT is not ”.” this is -10log_10 p(no variant). High QUAL scores indicate high confidence calls. Although traditionally people use integer phred scores, this field is permitted to be a floating point to enable higher resolution for low confidence calls if desired. If unknown, the missing value should be specified. (Numeric)
- **FILTER**: PASS if this position has passed all filters, i.e. a call is made at this position. Otherwise, if the site has not passed all filters, a semicolon-separated list of codes for filters that fail. gVCF files use the following values:
  - *PASS*: position has passed all filters.
  - *IndelConflict*: Locus is in region with conflicting indel calls.
  - *SiteConflict*: Site genotype conflicts with proximal indel call. This is typically a heterozygous SNV call made inside of a heterozygous deletion.
  - *LowGQX*: Locus GQX (minimum of {Genotype quality assuming variant position,Genotype quality assuming non-variant position}) is less than 30 or not present.
  - *HighDPFRatio*: The fraction of basecalls filtered out at a site is greater than 0.3.
  - *HighSNVSB*: SNV strand bias value (SNVSB) exceeds 10. High strand bias indicates a potential high false-positive rate for SNVs.
  - *HighSNVHPOL*: SNV contextual homopolymer length (SNVHPOL) exceeds 6.
  - *HighREFREP*: Indel contains an allele which occurs in a homopolymer or dinucleotide track with a reference repeat greater than 8.
  - *HighDepth*: Locus depth is greater than 3x the mean chromosome depth.
- **INFO**: Additional information. INFO fields are encoded as a semicolon-separated series of short keys with optional values in the format: <key>=<data>[,data]. gVCF files use the following values:

  - *END*: End position of the region described in this record.
  - *BLOCKAVG_min30p3a*: Non-variant site block. All sites in a block are constrained to be non-variant, have the same filter value, and have all sample values in range [x,y], y <= max(x+3,(x*1.3)). All printed site block sample values are the minimum observed in the region spanned by the block.
  - *SNVSB*: SNV site strand bias.
  - *SNVHPOL*: SNV contextual homopolymer length.
  - *CIGAR*: CIGAR alignment for each alternate indel allele.
  - *RU*: Smallest repeating sequence unit extended or contracted in the indel allele relative to the reference. RUs are not reported if longer than 20 bases.
  - *REFREP*: Number of times RU is repeated in reference.
  - *IDREP*: Number of times RU is repeated in indel allele.
- **FORMAT**: Format of the sample field. FORMAT specifies the data types and order of the subfields. gVCF files use the following values:
  - *GT*: Genotype.
  - *GQ*: Genotype Quality.
  - *GQX*: Minimum of {Genotype quality assuming variant position,Genotype quality assuming non-variant position}.
  - *DP*: Filtered basecall depth used for site genotyping.
  - *DPF*: Basecalls filtered from input before site genotyping.
  - *AD*: Allelic depths for the ref and alt alleles in the order listed. For indels this value only includes reads which confidently support each allele (posterior probability 0.999 or higher that read contains indicated allele vs all other intersecting indel alleles).
  - *DPI*: Read depth associated with indel, taken from the site preceding the indel.
- **SAMPLE**: Sample fields as defined by the header.

FPKM Files

In each output directory, this app will create two output files:

- ***genes.fpkm_tracking***, quantifying the expression of genes specified in the GTF annotation file.
- ***isoforms.fpkm_tracking***, quantifying the expression of transcripts specified in the GTF annotation file.

Expression is reported in terms of FPKM, or Fragments Per Kilobase of sequence per Million mapped reads. In simple terms, this measure normalizes the number of aligned reads by the size of the sequence feature and the total number of mapped reads.

Coverage.BedGraph.gz Files

The *.coverage.BedGraph.gz file has a summarised coverage useful for summary of coverage in browsers (IGV, UCSC genome browser), much smaller than the BAM file.

For an explanation of the bedgraph format, see:
http://genome.ucsc.edu/FAQ/FAQformat.html#format1.8.

Junctions.bed Files

The *.junctions.bed file is a bed file with junctions and the coverage of these for visualisation in a browser.

Candidate Fusion List

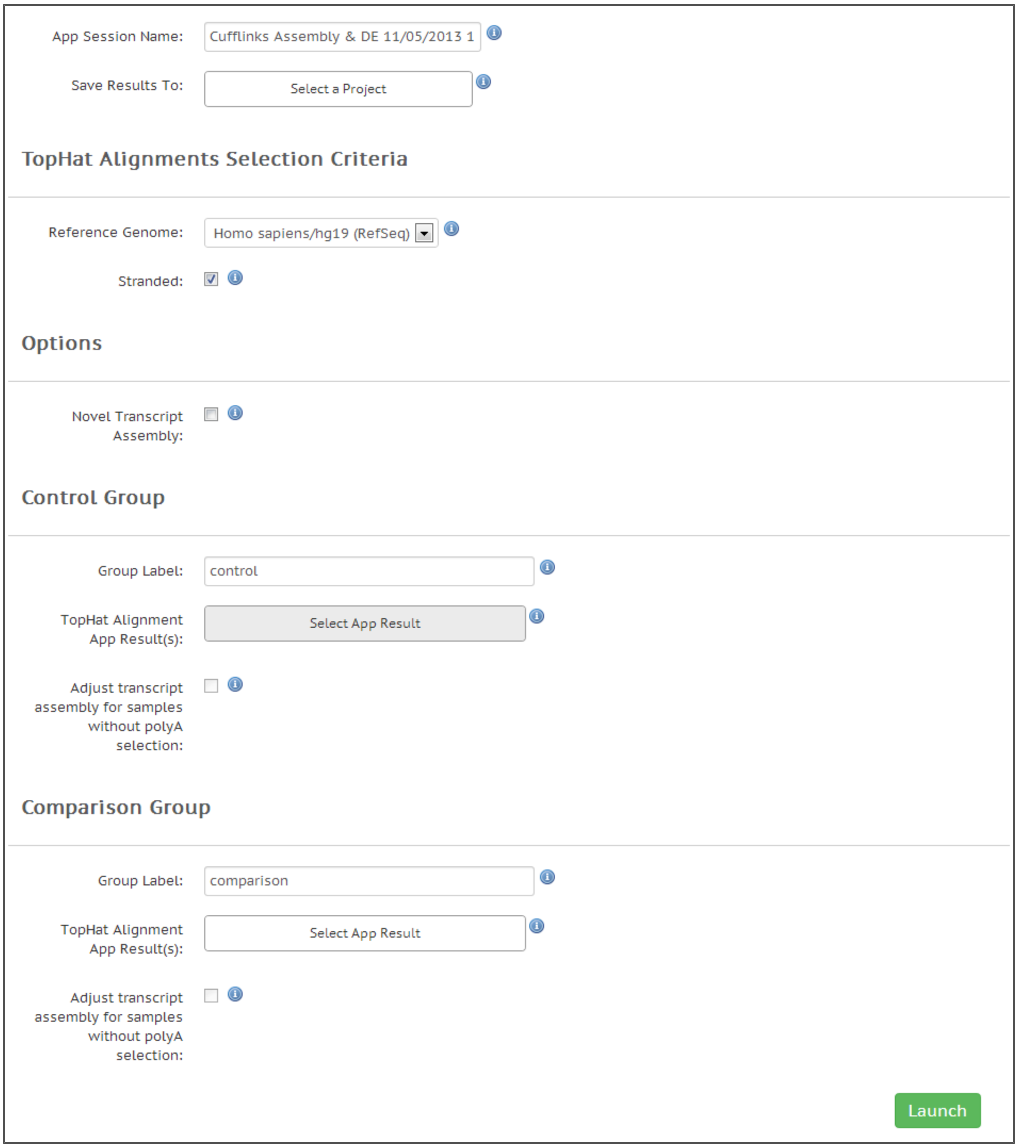
The candidate fusion list is an html page that provides an overview of all candidate fusions. The top tables provide the genes that are fused, the chromosomes and positions involved, as well as the number of reads that provide the evidence. This is explained in the table description. In addition, genomic and nucleotide blast searches are provided with the left and right sides of the fusion.

Run Cufflinks

1. Navigate to the project or sample that you want to analyze.
2. Click the **Launch Apps** button and select **Cufflinks Assembly & DE** from the dropdown list.
3. Read the End User License Agreement and permissions, and click **Accept** if you agree.
4. Fill out the app session storage information:
    a. **App Session Name**: provide the app session name. Default name is the app name with the date and time the app session was started.
    b. **Save Results To**: select the project that stores the app results.
5. Fill out the TopHat alignments selection criteria. These settings are used to filter the results available in the app result choosers below. This ensures that only compatible app results are selected for analysis.
    a. **Reference Genome**: select the reference genome.
    b. **Stranded**: indicate if samples were stranded.
6. Fill out the additional options:
    a. **Novel Transcript Assembly**: select if you want the Cufflinks app to perform novel transcript assembly.
7. Fill out the control group information:
    a. **Group Label**: provide the control group label. Default name is control.
    b. **Select App Result**: browse to the app result from the TopHat app that you want to use as control, and select the check box. You can use multiple app results as control.
    c. **Adjust transcript assembly for samples without polyA selection**: select if performing transcript assembly and your samples were generated without polyA selection. This will enable additional filtering of intronic alignments during transcript assembly to reduce the false identification of transcripts.
8. Fill out the comparison group information:
    a. **Group Label**: provide the comparison group label. Default name is comparison.
    b. **Select App Result**: browse to the app result from the TopHat app that you want to analyze against the control, and select the check box. You can analyze multiple app results.
    c. **Adjust transcript assembly for samples without polyA selection**: select if performing transcript assembly and your samples were generated without polyA selection.
9. Click **Launch**.

The Cufflinks app will now analyze your sample. When completed, the status of the app session is automatically updated, and you receive an email.

Figure 6 Cufflinks Input Form



App Session Name: Cufflinks Assembly & DE 11/05/2013 1

Save Results To: Select a Project

TopHat Alignments Selection Criteria

Reference Genome: Homo sapiens/hg19 (RefSeq)

Stranded: ☑

Options

Novel Transcript Assembly: ☐

Control Group

Group Label: control

TopHat Alignment App Result(s): Select App Result

Adjust transcript assembly for samples without polyA selection: ☐

Comparison Group

Group Label: comparison

TopHat Alignment App Result(s): Select App Result

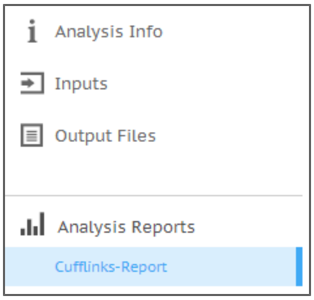
Adjust transcript assembly for samples without polyA selection: ☐

Launch

Cufflinks App Output

This chapter describes the output that is produced by the Cufflinks app. To go to the results, click the **Projects** button, then the project, then the analysis.

Figure 7 Cufflinks App Output Navigation Bar



When the analysis is completed, you can access your output through the left navigation bar, which provides the following:

- **Analysis Info**: an overview of the analysis. See *Analysis Info* on page 22 for a description.
- **Inputs**: an overview of the input samples and settings. See *Cufflinks Inputs Overview* on page 23 for a description.
- **Output Files**: access to the output files, organized by sample and app session. See *Cufflinks Output Files* on page 24 for descriptions.
- **Cufflinks Report**: access to analysis metrics for the aggregate results. See *Cufflinks Report* on page 20 for a description.

Cufflinks Report

The Cufflinks App Result Page provides an overview of statistics of the Cufflinks App Session. A brief description of the metrics is below.

Overview

Provides links to the control and comparison samples, as well as to the FPKM tables for genes and transcripts (see *FPKM Files* on page 24).

Assembly

The Assembly table provides metrics derived from the assembly of the control and comparison samples, as well as the two samples merged.

Statistic	Definition
Gene Count	Number of identified genes per sample.
Transcript Count	Number of identified transcipts per sample.
Link to gene models	Link to the GTF results (see *GTF Files* on page 25).
Relation to reference transcripts	Link to reference information.
Equal (=)	Complete match of intron chain.

Statistic	Definition
Potentially novel (j)	Potentially novel isoform (fragment): at least one splice junction is shared with a reference transcript.
Unknown, intergenic (u)	Unknown, intergenic transcript.
Overlap with opposite-strand exon (x)	Exonic overlap with reference on the opposite strand.
Other	Other types of transcripts.

Differential Expression

Statistic	Definition
Gene Count	Number of identified genes assessed in differential expression calculation.
ΔGene Count	Number of genes for which differential expression is significant.
Transcript Count	Number of identified transcripts assessed in differential expression calculation.
ΔTranscript Count	Number of transcripts for which differential expression is significant.
Selected CuffDiff results	Links to selected Cuffdiff results.

Sample Correlation

Sample correlation shows the similarity of the samples, based on the correlation of expression levels. This is determined the following way:

- Any gene with a reported confidence interval for the FPKM estimate that is greater than 25% of the estimated FPKM value is filtered out.
- Any gene with an FPKM value less than 1.0 is filtered out.
- For each sample pair, we calculate a correlation over the shared genes that were not filtered out in the above step from the natural log of the estimated FPKM values.

The sample correlation section includes a heat plot and clustering dendrogram.

Differential Expression Gene Browser

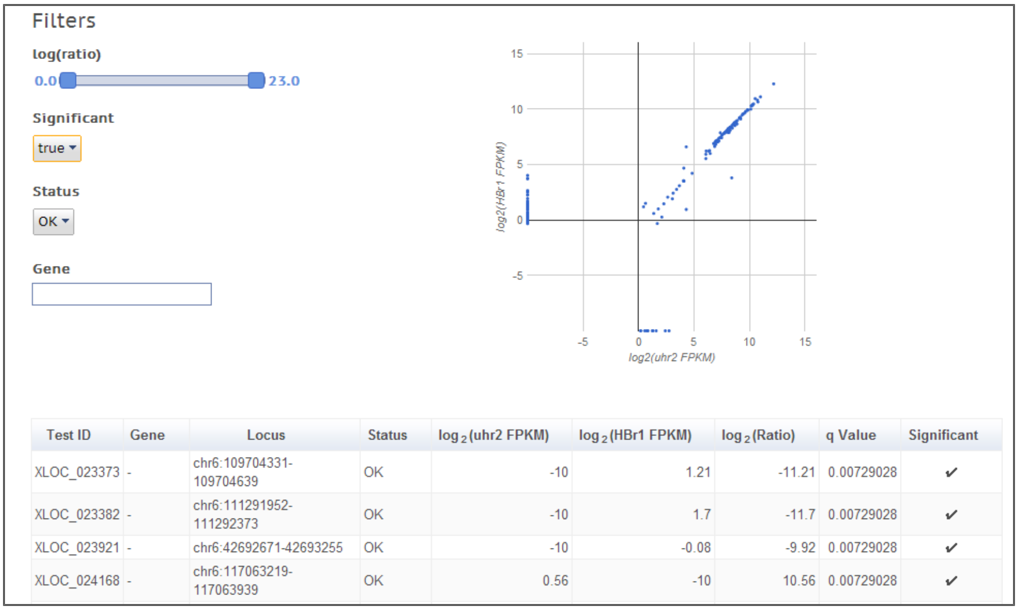
The Differential Expression Gene Browser shows an interactive scatter plot of the log2 (FPKM) counts of genes for two groups of samples. You can filter the results by the following metrics:

- The absolute value of log2(ratio) of expression levels between two sample groups.
- Significance of the differential expression.
- Test status:
  - OK: test successful.
  - NOTEST: not enough alignments for testing.
  - FAIL: when an ill-conditioned covariance matrix or other numerical exception prevents testing.
  - HIDATA: too many fragments in locus.
  - LOWDATA: The region being tested was either too complex or too shallowly sequenced to support a reliable calculation of abundance.

- Gene: allows you to search for a particular gene in the plot and the gene table below the scatter plot.

The gene table below the scatter plot shows the individual gene results. If you click on a gene, the corresponding dot is circled in the scatter plot. Vice versa, if you click on any of the dots in the scatter plot, the gene is highlighted in the gene table. You can also sort the data by clicking on a column header.

Figure 8 Differential Expression Gene Browser



Test ID	Gene	Locus	Status	$\log_2$(uhr2 FPKM)	$\log_2$(HBr1 FPKM)	$\log_2$(Ratio)	q Value	Significant
XLOC_023373	-	chr6:109704331-109704639	OK	-10	1.21	-11.21	0.00729028	✓
XLOC_023382	-	chr6:111291952-111292373	OK	-10	1.7	-11.7	0.00729028	✓
XLOC_023921	-	chr6:42692671-42693255	OK	-10	-0.08	-9.92	0.00729028	✓
XLOC_024168	-	chr6:117063219-117063939	OK	0.56	-10	10.56	0.00729028	✓

Analysis Info

This app provides an overview of the analysis on the Analysis Info page. A brief description of the metrics is below.

Row	Definition
Name	Name of the app session.
Application	App that generated this analysis.
Date started	Date the app session started.
Date completed	Date the app session completed.
Duration	Duration of analysis.
Status	Status of the app session.

Log Files

Clicking on the **Log Files** link at the bottom of the Analysis Info page provides access to TopHat and Cufflinks app log files.

The following files log information to help follow data processing and debugging:

- **WorkflowLog.txt**: Workflow standard output (contains details about workflow steps, command line calls with parameters, timing and progress).

- **WorkflowError.txt**: Workflow standard error output (contains errors messages created while running the workflow).
- **Logging.zip**: Contains all detailed workflow log files for each step of the workflow (content from Isis Logging folder)
- **IlluminaAppsService.log.copy**: Wrapper log file containing information about communication (get and post requests) between BaseSpace and AWS.

The following files contain additional information in case things (like mono) do not work as expected:

- **monoErr.txt**: Wrapper mono call error output (contains anything that is not caught by the WorkflowError.txt and will on most cases be empty, except one line).
- **monoOut.txt**: Wrapper mono call standard output (contains command calling the workflow and anything that is not caught by the WorkflowLog.txt).

NOTE
For explanation about mono, see www.mono-project.com.

Cufflinks Status

The status of the Cufflinks app session can have the following values:

- Preparing data
- Assembling transcripts
- Merging samples
- Calculating differential expression
- Calculating metrics
- Finalizing results

Depending on the size and number of samples, the complete analysis may take between a few hours and several days.

Cufflinks Inputs Overview

The Cufflinks app provides an overview of the input app results and settings on the Inputs page. A brief description of the metrics is below.

Statistic	Definition
App Session Name	Name of the app session.
Save Results To	The project that stores the app results.
Reference Genome	Reference genome selected.
Stranded	Indicates if samples were stranded.
Novel Transcript Assembly	Selected if the Cufflinks app was set to perform novel transcript assembly.
Group Label	The group label.
TopHat Alignment App Result(s)	App results selected for that group.
Adjust transcript assembly for samples without polyA selection	Selected if the samples for that group were generated without polyA selection.

Cufflinks Output Files

This section describes the main output files of the Cufflinks App:

- *FPKM Files* on page 24
- *GTF Files* on page 25
- *DIFF Files* on page 25

The output files for both apps are located in the following locations:

`<Analysis Result>/control` (if novel transcript assembly requested)

- cuffmerge/control.merged.gtf—Assembled and merged transcripts from designated control samples
- cuffmerge/quantification/<sample name>/genes.fpkm_tracking—Per-gene FPKM estimates for sample against newly assembled transcripts in control.merged.gtf
- cuffmerge/quantification/<sample name>/isoforms.fpkm_tracking—Per-transcript FPKM estimates for sample against newly assembled transcripts in control.merged.gtf
- metrics - Metrics on assembled control transcripts

`<Analysis Result>/comparison` (if novel transcript assembly requested)

- cuffmerge/comparison.merged.gtf—Assembled and merged transcripts from designated comparison samples
- cuffmerge/quantification/<sample name>/genes.fpkm_tracking—Per-gene FPKM estimates for sample against newly assembled transcripts in comparison.merged.gtf
- cuffmerge/quantification/<sample name>/isoforms.fpkm_tracking—Per-transcript FPKM estimates for sample against newly assembled transcripts in comparison.merged.gtf
- metrics—Metrics on assembled comparison transcripts

`<Analysis Result>/differential/cuffmerge` (if novel transcript assembly requested)

- control_vs_comparison.merged.gtf—Assembled and merged transcript for use in differential expression

`<Analysis Result>/differential/cuffdiff`—CuffDiff output from comparison of control and comparison samples

- control_vs_comparison.gene_exp.diff—Differential expression results for genes
- control_vs_comparison.isoform_exp.diff—Differential expression results for transcripts

FPKM Files

In each output directory, this app will create two output files:

- ***genes.fpkm_tracking***, quantifying the expression of genes specified in the GTF annotation file.
- ***isoforms.fpkm_tracking***, quantifying the expression of transcripts specified in the GTF annotation file.

Expression is reported in terms of FPKM, or Fragments Per Kilobase of sequence per Million mapped reads. In simple terms, this measure normalizes the number of aligned reads by the size of the sequence feature and the total number of mapped reads.

GTF Files

The ***merged.gtf*** file provides the set of merged transcripts. Each line contains an annotation field ("class_code") that describes the nature of the overlap of this transcript with transcripts from the reference annotation. The table below, taken from the cufflinks manual (http://cufflinks.cbcb.umd.edu/manual.html), provides a description of the possible class codes.

=	Match
j	New isoform
e	A single exon transcript overlapping a reference exon and at least 10 bp of a reference intron, indicating a possible pre-mRNA fragment
i	A single exon transcript falling entirely with a reference intron
r	Repeat, currently determined by looking at the reference sequence and applied to transcripts where at least 50% of the bases are lower case
p	Possible polymerase run-on fragment
u	Unknown, intergenic transcript
o	Unknown, generic overlap with reference
.	Tracking file only, indicates multiple classifications

Transcripts annotated with the 'i', 'j', 'u', or 'o' class codes represent novel transcripts of potential interest.

DIFF Files

The Cufflinks app creates several DIFF files that describe the differential expression. This tab delimited file lists the results of differential expression testing between samples for spliced transcripts, primary transcripts, genes, and coding sequences.

- *isoform_exp.diff*: Transcript differential FPKM.
- *gene_exp.diff*: Gene differential FPKM. Tests differences in the summed FPKM of transcripts sharing each `gene_id`.
- *tss_group_exp.diff*: Primary transcript differential FPKM. Tests differences in the summed FPKM of transcripts sharing each `tss_id`.
- *cds_exp.diff*: Coding sequence differential FPKM. Tests differences in the summed FPKM of transcripts sharing each `p_id` independent of `tss_id`.

The DIFF file has the following format:

Column number	Column name	Example	Description
1	Tested id	XLOC_ 000001	A unique identifier describing the transcipt, gene, primary transcript, or CDS being tested
2	gene	Lypla1	The gene_name(s) or gene_id(s) being tested
3	locus	chr1:4797771-4835363	Genomic coordinates for easy browsing to the genes or transcripts being tested.
4	sample 1	Liver	Label (or number if no labels provided) of the first sample being tested
5	sample 2	Brain	Label (or number if no labels provided) of the second sample being tested

Column number	Column name	Example	Description
6	Test status	NOTEST	Can be one of the following: • OK (test successful) • NOTEST (not enough alignments for testing) • LOWDATA (too complex or shallowly sequenced) • HIDATA (too many fragments in locus) • FAIL, when an ill-conditioned covariance matrix or other numerical exception prevents testing.
7	FPKMx	8.01089	FPKM of the gene in sample x
8	FPKMy	8.551545	FPKM of the gene in sample y
9	log2 (FPKMy/FPKMx)	0.06531	The (base 2) log of the fold change y/x
10	test stat	0.860902	The value of the test statistic used to compute significance of the observed change in FPKM
11	p value	0.389292	The uncorrected p-value of the test statistic
12	q value	0.985216	The FDR-adjusted p-value of the test statistic
13	significant	no	Can be either "yes" or "no", depending on whether p is greater then the FDR after Benjamini-Hochberg correction for multiple-testing

For more information, see the Cufflinks manual at http://cufflinks.cbcb.umd.edu/manual.html.

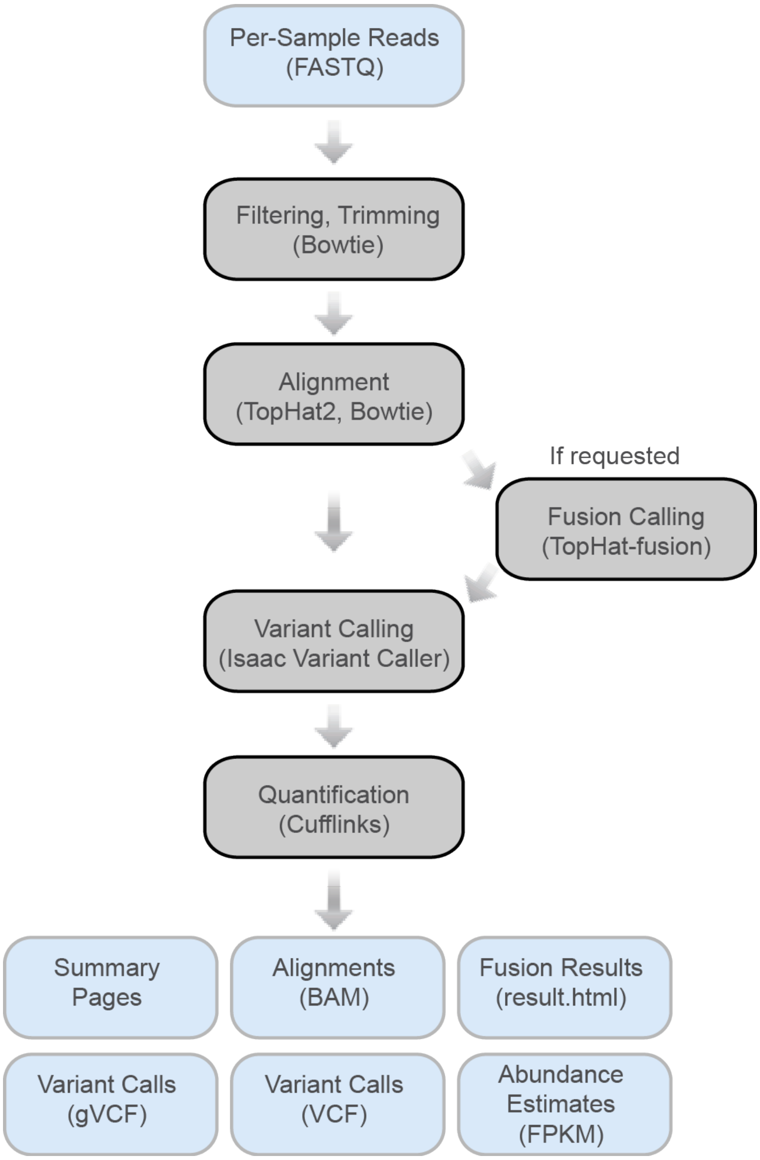
RNAseq Methods

This chapter describes the methods that are used in the TopHat and Cufflinks apps.

TopHat App Workflow

1. **Filtering**. The first step of the workflow is to filter the input reads against abundant sequences, such as mitochondrial or ribosomal sequences, as defined by iGenomes (support.illumina.com/sequencing/sequencing_software/igenome.ilmn). Also. The workflow has an internal library of such sequences that it uses for this purpose, including mitochondrial and ribosomal sequences. Bowtie (see *Bowtie* on page 29) performs this alignment. Only sequences that do not align against abundant sequences are passed through to the next phase of the analysis. Read pairs are filtered if at least one read aligns to an abundant sequence. This filtering step also performs trimming of 2 bases from the 5' end of the read. This is due to a consistent observation of a high mismatch rate from these two bases in RNA-seq libraries.
2. **Alignment**. TopHat2 (see *TopHat* on page 29), using the Bowtie 1 aligner, performs a spliced alignment of the filtered reads against the genome. Based on the user-specified genome, TopHat uses a list of known transcripts to align reads against known transcripts and splice junctions.
3. **Fusion Calling**: If requested, Tophat-fusion detects gene fusions. Fusion calling occurs in two steps. First, TopHat2 is run in a mode to allow the detection of fused alignments. Then, a post-alignment analysis script identifies candidate fusion genes from these fusion alignments. This step does not work with STAR alignments.
4. **Variant Calling**: The Isaac Variant Caller (see *Isaac Variant Caller* on page 29) performs variant calling, producing gVCF output. For stranded library preps, the strand bias filter will be disabled. In addition, the workflow makes use of the –`bsnp-diploid-het-bias` parameter to expand the allowable range for the heterozygous variant call, in order to account for allele-specific expression.
5. **Quantification**: Cufflinks (see *Cufflinks* on page 31) is used for quantification of reference genes and transcripts.

Figure 9 TopHat App Workflow

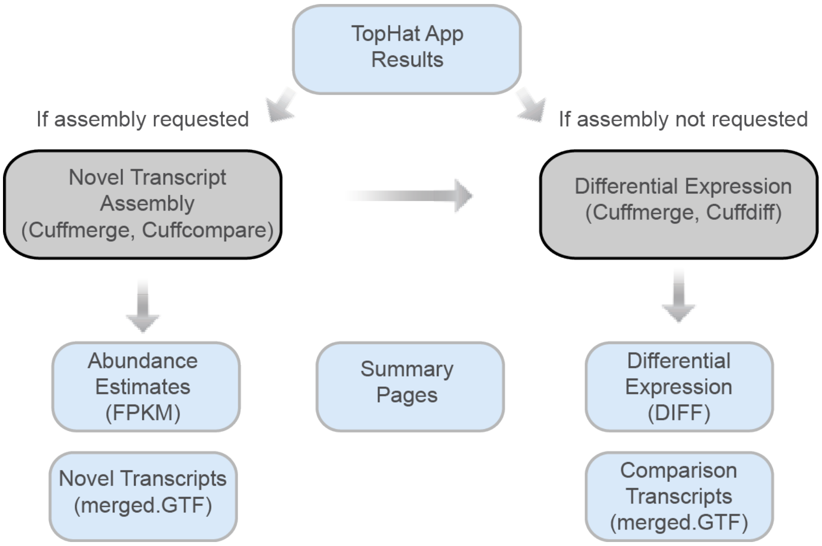


Cufflinks App Workflow

1. **Novel transcript assembly**. If novel transcript assembly is requested, transcripts are first assembled and quantified independently for each sample. Cuffmerge merges the assemblies between the samples of each group and combines them with the known gene models from annotation. Cuffcompare then provides annotations of novel transcripts, for example, if they overlap with a known transcript. Lastly abundances will be re-estimated for each transcript, known or novel.
2. **Differential expression**. Cuffdiff calculates differential expression between these two sample groups, estimating variance from the individual samples supplied for each sample group. Cuffdiff calculates a variety of information regarding differential expression, which are summarized in the on-line manual (cufflinks.cbcb.umd.edu/manual.html#cuffdiff_output) for that tool.
   Cuffdiff needs a consensus set of transcripts that is compared between different sample groups. If no novel transcript detection is performed, this consensus set is

simply the reference annotation. If novel transcript assembly is requested, then the set of transcripts from each sample must first be merged together. The Cuffmerge tool does this, while also merging in the known gene models from annotation.

Figure 10 TopHat and Cufflinks Workflow



Bowtie

Bowtie[1] is an ultrafast, memory-efficient aligner designed for quickly aligning large sets of short reads to large genomes. Bowtie indexes the genome to keep its memory footprint small: for the human genome, the index is typically about 2.2 GB for single-read alignment or 2.9 GB for paired-end alignment. Multiple processors can be used simultaneously to achieve greater alignment speed.

Bowtie forms the basis for other tools like TopHat, a fast splice junction mapper for RNA-seq reads, and Cufflinks, a tool for transcriptome assembly and isoform quantitation from RNA-seq reads.

TopHat

TopHat[2] is a fast splice junction mapper for RNA-Seq reads. It aligns RNA-Seq reads to mammalian-sized genomes using the ultra high-throughput short read aligner Bowtie, and then analyzes the mapping results to identify splice junctions between exons.

TopHat is a collaborative effort between the University of Maryland Center for Bioinformatics and Computational Biology and the University of California, Berkeley Departments of Mathematics and Molecular and Cell Biology.

Isaac Variant Caller

The Isaac Variant Caller (the algorithm is also referred to as Starling2) identifies single nucleotide polymorphisms (SNPs) and small indels using the following steps:

- **Read filtering**—Filters out reads failing quality checks.
- **Indel calling**—Identifies a set of possible indel candidates and realigns all reads overlapping the candidates using a multiple sequence aligner.
- **SNP calling**—Computes the probability of each possible genotype given the aligned read data and a prior distribution of variation in the genome.
- **Indel genotypes**—Calls indel genotypes and assigns probabilities.

- **Variant call output**—Generates output in a vcf and a compressed genome variant call (gVCF) file. See *VCF Files* on page 12 and *gVCF Files* on page 12 for details.

Indel Candidates

Input reads are filtered by removing any of the following:

- Reads that failed primary analysis quality checks.
- Reads marked as PCR duplicates.
- Paired-end reads not marked as a proper pair.
- Reads with a mapping quality less than 20.

Indel Calling

The variant caller proceeds with candidate indel discovery and generates alternate read alignments based on the candidate indels. As part of the realignment process, the variant caller selects a representative alignment to be used for site genotype calling and depth summarization by the SNP caller.

SNP Calling

The variant caller runs a series of filters on the set of filtered and realigned reads for SNP calling without affecting indel calls. First, any contiguous trailing sequence of N base calls are trimmed from the ends of reads. Using a mismatch density filter, reads having an unexpectedly high number of disagreements with the reference are masked, as follows:

- The variant caller treats each insertion or deletion as a single mismatch.
- Base calls with more than two mismatches to the reference sequence within 20 bases of the call are ignored.
- If the call occurs within the first or last 20 bases of a read, the mismatch limit is applied to a 41-base window at the corresponding end of the read.
- The mismatch limit is applied to the entire read when the read length is 41 or shorter.

Indel Genotypes

All bases marked by the mismatch density filter and any N base calls that remain after the end-trimming step are filtered out by the variant caller. These filtered base calls are not used for site-genotyping but appear in the filtered base call counts in the variant caller output for each site.

All remaining base calls are used for site-genotyping. To account for the possibility of error dependencies, the genotyping method heuristically adjusts the joint error probability that is calculated from multiple observations of the same allele on each strand of the genome. This method treats the highest quality base call from each allele and strand as an independent observation and leaves the associated base call quality scores unmodified. However, quality scores for subsequent base calls for each allele and strand are adjusted to increase the joint error probability of the given allele above the error expected from independent base call observations.

Variant Call Output

After the site and indel genotyping methods are complete, the variant caller applies a final set of heuristic filters to produce the final set of non-filtered calls in the output.

The output in the genome variant call (gVCF) file captures the genotype at each position and the probability that the consensus call differs from reference, which is expressed as a phred-scaled quality score.

Cufflinks

Cufflinks[3] assembles transcripts, estimates their abundances, and tests for differential expression and regulation in RNA-Seq samples. It accepts aligned RNA-Seq reads and assembles the alignments into a parsimonious set of transcripts. Cufflinks then estimates the relative abundances of these transcripts based on how many reads support each one.

Cufflinks is a collaborative effort between the Laboratory for Mathematical and Computational Biology, led by Lior Pachter at UC Berkeley, Steven Salzberg's group at the University of Maryland Center for Bioinformatics and Computational Biology, and Barbara Wold's lab at Caltech.

References

1 Langmead B, Trapnell C, Pop M, Salzberg SL. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol 10(3):R25.

2 Trapnell C, Pachter L, Salzberg SL.(2009) TopHat: discovering splice junctions with RNA-Seq. Trapnell C, Pachter L, Salzberg SL. Bioinformatics 25(9):1105-11.

3 Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, van Baren MJ, Salzberg SL, Wold BJ, Pachter L. (2010) Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nat Biotechnol 28(5):511-5.

Notes

Technical Assistance

For technical assistance, contact Illumina Technical Support.

Table 1 Illumina General Contact Information

Illumina Website	www.illumina.com
Email	techsupport@illumina.com

Table 2 Illumina Customer Support Telephone Numbers

Region	Contact Number	Region	Contact Number
North America	1.800.809.4566	Italy	800.874909
Austria	0800.296575	Netherlands	0800.0223859
Belgium	0800.81102	Norway	800.16836
Denmark	80882346	Spain	900.812168
Finland	0800.918363	Sweden	020790181
France	0800.911850	Switzerland	0800.563118
Germany	0800.180.8994	United Kingdom	0800.917.0041
Ireland	1.800.812949	Other countries	+44.1799.534000

Safety Data Sheets

Safety data sheets (SDSs) are available on the Illumina website at www.illumina.com/msds.

Product Documentation

Product documentation in PDF is available for download from the Illumina website. Go to www.illumina.com/support, select a product, then click **Documentation & Literature**.

Illumina
San Diego, California 92122 U.S.A.
+1.800.809.ILMN (4566)
+1.858.202.4566 (outside North America)
techsupport@illumina.com
www.illumina.com