CONTRA: Copy Number Analysis for Targeted Resequencing

(Redirected from Main Page)

**** 21-3-2016 updates (v2.0.8): New workflows - Null Distribution Estimation (NDE) and Whole-gene analysis (WGCNV) with plots**

navigation
- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search
Go Search

toolbox
- What links here
- Related changes
- Special pages
- Printable version
- Permanent link

Contents [hide] [hide]

Introduction

CONTRA is a tool for copy number variation (CNV) detection for targeted resequencing data such as those from whole-exome capture data. CONTRA calls copy number gains and losses for each target region with key strategies include the use of base-level log-ratios to remove GC-content bias, correction for an imbalanced library size effect on log-ratios, and the estimation of log-ratio variations via binning and interpolation. It takes standard alignment formats (BAM/SAM) and output in variant call format (VCF 4.0) for easy integration with other next generation sequencing analysis package.

21-3-2016: *VCF has failed to become a standard format for Copy Number analysis output. So the tabular output format from contra.py is all you need. For versions prior to 2.0.8, ignore all warnings about reference ".fasta" file; it is required for VCF generation only. Newer versions do not require reference fasta any more. Use the largest tab-delimited output. Learn about our new workflow below regarding plots and gene-level calls.*

For support, please contact jason dot li at petermac dot org and richard dot lupat at petermac dot org.

Citing CONTRA

CONTRA has been widely used and cited since publication. If you have used contra.py or baseline.py, please cite the CONTRA paper:

> Li, J, Lupat, R, Amarasinghe, KC, Thompson, ER, Doyle, MA, Ryland, GL, Tothill, RW, Halgamuge, SK, Campbell, IG, Gorringe, KL (2012). "CONTRA: copy number analysis for targeted resequencing," Bioinformatics **28**(10): 1307-1313

.

If you have used the WGCNV or NDE workflows, please cite the CONTRA paper and state explicitly the use of WGCNV and/or NDE workflows.

Download

- Source File
  Download CONTRA source files
- User Guide
  Use **this web page (most up-to-date)**
- Test Files
  Test Files (see README.txt in the folder)

  Reference Genome (Ensembl)
  Reference Genome (1000genomes)
- Baseline Files
  There are seven baseline files that we have generated from either our samples or HapMap. Baseline file for the HapMap is generated from the 6 of the CEU samples that uses NimbleGen SeqCap EZ Exome V2 (Exome Center = BGI).

  Download Baseline files

Requirements

To run CONTRA, you need the following programs:

- Python 2.6+
  Most of the scripts for CONTRA are written in Python. It requires version 2.6 in order to use the multiprocessing module. Python Website
- R (now support all versions of R)
  R Website
- BEDTools (Included in CONTRA package. See Installation Guide)
  The original source of the BEDtools can be found in BEDtools source
- SAMtools
  SAMtools source
- [Optional] DNACopy (R-library that will be used for predicting large CNV)
  DNACopy Website

Installation Guide

Download CONTRA tarball and decompress it with the following command:

tar –xvzf CONTRA.<version>.tar.gz

For users who DO NOT have BEDTools installed, follow these steps:

cd CONTRA.<version>
tar -zxvf BEDTools.tar.gz
cd BEDTools
make clean
make all
sudo cp bin/ /usr/local/bin/*

Format of Input Files

The analysis itself requires several files as following:

- BAM or SAM files for the test and control samples.
  User can also provide a baseline file (in BED format) as the control sample. Please refer to " Baseline" section for example how to use the baseline script.
- Target File in BED format (UCSC BED FORMAT FAQ)
  A tab-delimited file specifying the target/capture regions. Four columns must be specified: Chromosome, ChromStart, ChromEnd and Name. Name is typically gene symbols. It is used to map regions or exons to a gene, and is required by WGCNV and NDE workflows (v2.0.8+). No header line is expected.
  The ChromStart is 0 base (the first base in a chromosome is numbered 0). The ChromEnd base is not included in the display of the feature. For example, chromStart= 0 and chromEnd = 100 is implying the first 100 bases of a chromosome (0-99).
  Below is the example of first few lines of a target.BED file:
  1 357512 357632 uc001aaw.1
  1 357571 357691 uc001aaw.1
  1 357633 357753 uc001aaw.1
  1 357694 357814 uc001aaw.1
  1 357756 357876 uc001aaw.1
  1 357816 357936 uc001aaw.1
- (DEPRECATED; Fasta not required in v2.0.8+) Fasta file for reference genome (e.g. human_g1k_v37.fasta)

CONTRA Workflow

21-3-2016: *New workflows added for CONTRA v2.0.8+*

Whole Exome:

For Whole-Exome data (i.e. having data points consistenly throughout the genome), ADTEx works brilliantly. Use it instead. See http://adtex.sourceforge.net/. For custom targets, CONTRA is the right tool.

Gene-level specific calls (WGCNV):

Noise is an intrinsic property of capture data, and high false positive rates at the exon-level is inevitable. Most capture is designed to target whole-genes consisting of multiple exons. We can take advantage of this fact to generate more specific calls at the gene-level (will miss exon scale event).

- Step-1
  Generate CONTRA results using a bed file in which the fourth column contains the gene symbols. These gene symbols will determine how exons are grouped.
- Step-2 (for a single-case; without NDE)
  WGCNV-SINGLE (see new package in CONTRA v2.0.8+) will summarise all exon-level log-ratios generated by CONTRA, and perform a student t-test against a hardcoded threshold of +/-0.2 to compute statistical significance. Use the adjusted p.value and "n" (number of windows) to filter the results. If somatic variants are provided, B-Allele Frequencies will be plotted alongside the log-ratios for visual inspection. These new plots make CONTRA much more user-friendly. If the program complaints your variant file not having the right format, contact us.
- OR Step-2 (for multiple-cases; with NDE)
  You would have first followed the Null Distribution Estimation workflow to generate background estimations. WGCNV-NDE (see new package in CONTRA v2.0.8+) will then summarise all exon-level log-ratios generated by CONTRA, and compute a z-score against the null distribution to derive the statistical significance. Use the adjusted p.value and "n" (number of windows) to filter the results. If somatic variants are provided, B-Allele Frequencies will be plotted alongside the log-ratios for visual inspection. These new plots make CONTRA much more user-friendly. If the program complaints your variant file not having the right format, contact us.

Null Distribution Estimation workflow:

Capture efficiency varies across target regions. This means some regions are associated with a lower variation in their depth of coverage, are hence are more reliable for copy number calling. Estimating the variance of DOC specific to each target region is possible given a large number (N > 50) of samples derived from the same capture design. This workflow can be applied in both scenarios where a pooled baseline is used or a matched control is used.

- Step-1
  Generate CONTRA results using a bed file in which the fourth column contains the gene symbols. These gene symbols will determine how exons are mapped to genes.
- Step-2
  The NullDistEstimation package (found in v2.0.8+) will perform the following: 1) re-normalise the DOC levels after excluding regions with very high amplification. This step is particularly useful for small captures (around 50 genes) as a high gain in one single gene can steal a significant amount of reads from the other target regions, affecting library size correction. 2) Estimate the null distribution of log-ratios from all given samples. This is done at both the exon-level and gene-level.
- Step-3
  Use WGCNV-NDE (see above) to call CNVs based on the estimated Null.

Standard CONTRA:

- Step-1
  CONTRA takes all the required files from the user [target, test BAM/SAM, control BAM/SAM/baseline file, fasta], and do all the pre-processing steps to ensure all the input files are compatible with the CONTRA script (such as files need to be in sorted order). If maxRegionSize is specified, large regions will be broken down into smaller regions depending on user specified parameters
- Step-2
  Short-read alignment information (BAM or SAM formats) from a test and a control sample is converted into read count per base pair for a list of target regions (BED format) using BEDTools. Target regions with too few reads in the control sample (by default, < 10 base pairs with read count > 10) are excluded from further analysis. The remaining read counts are then scaled based on the geometric mean of the total read counts of the two samples
- Step-3
  For each target region, a set of base-level log-ratios between test and control is calculated based on the scaled read counts, the mean of which is used to estimate the region's log-ratio. Library size bias is then removed based on a linear relationship between log-ratio and log-coverage estimated from the data. The regions are then binned based on their similarity in log-coverage
- Step-4
  Significance is then computed for each region and is adjusted to reduce false discovery rates. Results are reported with other details in either tab-delimited or the VCF4.0 format (Variant Call Format; see www.1000genomes.org)
- Step-5
  If large deletion option is specified, circular binary segmentation will be performed on region log-ratios, using different parameters to achieve different resolutions of segmentation. Segmentation results from different resolutions are combined to make the final call

Output Format

VCF (Deprecated: Not the best output format for copy number results)

VCF4.0 format (Use tabular file described below instead; v2.0.8+ does not support VCF any more)

Note:
QUAL = 10Log10 Adjusted p-value
Other columns are described in the VCF file header.

The details explanation for each column can be found in 1000Genomes VCF 4.0 Format

Tab Delimited

Three tab-delimited files will be generated:

1. Full details of the analysis, excluding target regions that do not pass the minimum read depth & number of bases thresholds (See Step 2 in CONTRA workflow)
2. Including only target regions that pass the p-value threshold
3. If largeDeletion option is specified, a summary of large CNV prediction with significant copy number changes will be presented in the tab-delimited file

The full details analysis file contains: target region ID, exon number, gene symbol, chromosome, original start coordinate, original end coordinate, mean, standard deviation and median of the log-ratios, number of bases included in the analysis for that target region, p-value, adjusted p-value, gain/loss, average test sample's scaled read depth, average control sample's scaled read depth, average test sample's original read depth, average control sample's original read depth, minimum and maximum log ratios on that target regions and the bin number.

Plot

If --plot option is specified when running the code, plot(s) for the distribution of log ratios will be included in the output folder.

Examples

Example 1: Running CONTRA with default optional parameters

We assume we have a target file *target_test.BED*, two BAM files *test_sample.BAM* and *control_sample.BAM* and a reference file *human_ref.fasta*. Our intended output folder is in *~/ContraTest/sampleName/*

To run the analysis on this sample, the command line argument is:

```
contra.py --target target_test.BED --test test_sample.BAM --control control_sample.BAM --fasta human_ref.fasta --outfolder ~/ContraTest/sampleName/
```

This will create a folder call *sampleName* inside *~/ContraTest/*. Inside the folder, there will be two subfolders, plot and table. The table folder will contain the VCF file and the analysis' details.

Note: CONTRA will always attempt to create the folder specified last (i.e. *sampleName* in this example). If the folder exists, there will be an error message when running the script. It is setup this way to avoid the data in the existing folder being overwrite. However, it will not create the parents folder (i.e. *ContraTest* in this example) with the assumption this folder has already existed. An attempt to put the result folders in a directory that has not been created will generate an error message.

Example 2: Running CONTRA with other optional parameters

We assume we have a target file *target_test.BED*, two BAM files *test_sample.BAM* and *control_sample.BAM* and a reference file *human_ref.fasta*. Our intended output folder is in *~/ContraTest/sampleName/*

The options we want to change for this example:

1. Number of bins to : 1,5,10,15 and 20 bins
2. Minimum read depth : 5
3. Minimum number of bases : 20
4. SampleName : sample123
5. Remove multi mapped reads

To run the analysis on this sample, the command line argument is:

```
contra.py --target target_test.BED --test test_sample.BAM --control control_sample.BAM --fasta human_ref.fasta --outfolder ~/ContraTest/sampleName/ --numBin 1,5,10,15,20 --minReadDepth 5 --minNBases 20 --sampleName sample123 --nomultimapped
```

Note1: The --nomultimapped option will use samtools to filter out alignment with mapping quality = 0.

Note2: With the --sampleName option, all the results' filename will be appended with the sample name in front of the default filename.

Example 3: Common workflows

Targetting ~1000 cancer genes, ongoing experiments. A case of tumour with matched germline:

RUN:

```
contra.py -t target_test.BED -s test_sample.bam -c control.bam -p --minExon=100 --maxRegionSize=100 --targetRegionSize=75 --removeDups --sampleName SampName -o ContraOutputFolder
```

THEN:

```
WGCNV-SINGLE/wgcnv.py ContraOutputFolder human VariantFile WGCNV_OUTDIR
```

**The format of *VariantFile* is described in README.txt under WGCNV-SINGLE.

**Without matched germline, one can use a known diploid sample as the germline (e.g. NA12878). Pooled baseline can be used if no germline is available at all. And then run WGCNV-SINGLE/wgcnv_noBAF.py instead of wgcnv.py

---

Having collected 30 to 50 cases, run above contra.py command on all, and then:

RUN:

```
NullDistEstimation/nde_wrapper.py ContraOut_List.txt NDE_OUTDIR BED_ANNOTATED T T
```

***ContraOut_List.txt* is a text file listing all samples' contra output directory to be used for null estimation, one sample per line. (parent directory to the 'table' directory generated by contra.py)

***BED_ANNOTATED* is the bed file appended with a fourth column containing gene symbols

For each current and future sample which you want CNV results for:

RUN normal contra.py first, THEN:

```
WGCNV-NDE/wgcnv_wrapper.py ContraOutputDir OUTDIR NDE_OUTDIR/thresh_wg.txt NDE_OUTDIR/thresh_ex.txt BED_ANNOTATED T human
```

***thresh_wg.txt* and *thresh_ex.txt* are gene-level and exon-level null estimates generated by nde_wrapper.py in the above steps

Data Simulator

TargetedSim Project Website - a tool to simulate targeted resequencing data. (by Kaushalya Amarasinghe 2011)

List of CONTRA parameters

Required

-t, --target

Target region definition file [BED format]

-s, --test

Alignment file for the test sample [BAM/SAM]

-c, --control Alignment file for the control sample [BAM/SAM/BED* – baseline file]

*--bed option has to be supplied for control with baseline file.

-f, --fasta

Reference genome [FASTA] (NOT REQUIRED since v2.0.8)

-o, --outFolder

the folder name (and its path) to store the output of the analysis (this new folder will be created – error message occur if the folder exists)

Optional

--numBin

Numbers of bins to group the regions. User can specify multiple experiments with different numbers of bins (comma separated). [Default: 20]

--minReadDepth

The threshold for minimum read depth for each bases (see Step 2 in CONTRA workflow) [Default: 10]

--minNBases

The threshold for minimum number of bases for each target regions (see Step 2 in CONTRA workflow) [Default: 10]

--sam

If the specified test and control samples are in SAM format. [Default: False] (It will always take BAM samples as default)

--bed

If specified, control will be a baseline file in BED format. [Default: False]
*Please refer to the Baseline Script section for instruction how to create baseline files from set of BAMfiles. A set of baseline files from different platform have also been provided in the CONTRA download page.

--pval

The p-value threshold for filtering. Based on Adjusted P-Values. Only regions that pass this threshold will be included in the VCF file. [Default: 0.05]

--sampleName

The name to be appended to the front of the default output name. By default, there will be nothing appended.

--nomultimapped

The option to remove multi-mapped reads (using SAMtools with mapping quality > 0). [default: FALSE]

-p, --plot

If specified, plots of log-ratio distribution for each bin will be included in the output folder [default: FALSE]

--minExon

Minimum number of exons in one bin (if less than this number, bin that contains small number of exons will be merged to the adjacent bins) [Default : 2000]

--minControlRdForCall

Minimum Control ReadDepth for call [Default: 5]

--minTestRdForCall

Minimum Test ReadDepth for call [Default: 0]

--minAvgForCall

Minimum average coverage for call [Default: 20]

--maxRegionSize

Maximum region size in target region (for breaking large regions into smaller regions. By default, maxRegionSize=0 means no breakdown). [Default : 0]

--targetRegionSize

Target region size for breakdown (if maxRegionSize is non-zero) [Default: 200]

-l, --largeDeletion

If specified, CONTRA will run large deletion analysis (CBS). User must have DNAcopy R-library installed to run the analysis. [False]

--smallSegment

CBS segment size for calling large variations [Default : 1]

--largeSegment

CBS segment size for calling large variations [Default : 25]

--lrCallStart

Log ratios start range that will be used to call CNV [Default : -0.3]

--lrCallEnd

Log ratios end range that will be used to call CNV [Default : 0.3]

--passSize

Size of exons that passed the p-value threshold compare to the original exons size [Default: 0.5]

--removeDups (new since 2.0.6)

if specified, will remove PCR duplicates [False]

--version

Returns version

Baseline

Creating a baseline control from multiple samples is can be useful when a matched control is not available. In the CONTRA download page, we have provided several baseline files for some of the platforms that we have tried. Alternatively, the “baseline.py” script that comes with CONTRA can be used to generate a custom baseline file.

List of parameters

-t, --target

Target region definition file [REQUIRED] [BED format]

-f, --files

Files to be converted to baselines [REQUIRED] [BAM]

-o, --output

Output folder [REQUIRED]

-c, --trim

Portion of outliers to be removed before calculating average [Default: 0.2]

-n, --name

Output baseline file name [Default: baseline]

Example

We assume we have a target file *target_test.BED* and four BAM files that we want to turn into baseline file *test1.BAM*, *test2.BAM* , *test3.BAM* and *test4.BAM*. Our intended output folder is in *~/Baseline/sampleBaseline/* with the final baseline file name *baseline_test*.

To run the baseline script on this sample, the command line argument is:

```
python baseline.py --target target_test.BED --files test1.BAM test2.BAM test3.BAM test4.BAM --output ~/Baseline/sampleBaseline/ --name baseline_test
```