Version

1

DAVIDE PRANDI

Centre for Integrative Biology, University of Trento, Italy

CLONET User Manual

Table of Contents

Chapter 1

DISCLAIMER

CLONET code is intended for research use only. Current (Apr 2014) version has NO WARRANTY.

Under no circumstances CLONET code can be redistributed.

INTRODUCTION

CLONET is a collection of R scripts for the computation of global DNA admixture (1-purity), ploidy, and clonality of tumor DNA samples (each with matched normal sample) from sequencing data.

Each script has the following syntax:

```
>CLONET.scriptName.R ConfigurationFile.R
```

Folders are organized as follows:

```
CLONET
-> CLONET.R
-> Docs
-> Examples
-> Functions
-> Tools
```

CLONET.R is the main R script required to compute global DNA admixture, ploidy and clonality of segmented data.

The Docs folder contains this document.

The Examples folder contains a folder Small that includes a complete run of all the CLONET scripts.

The Functions folder contains all the functions used by CLONET.

The Tools folder contains R scripts to perform point mutations (PM) analysis, structural rearrangement (RR) analysis and tumor evolution path analysis.

REQUIREMENTS

CLONET requires Linux kernel >= 2.6.15.

CLONET requires R >= 2.7 and the following packages, *parallel*, *dgof*, *sets*, and *pso*, *igraph*, *reshape2*.

CLONET requires global folder names.

CLONET requires ASEQ tool to generate initial pileup analysis (binaries provided).

CLONET ON SMALL EXAMPLE

A precompiled example with already generated data is accessible. To re-run it, the user needs to run the following commands from the CLONET folder:

```
>./CLONET.R  Examples/Small/ConfigurationFile/CLONET.Config.R

>./Tools/CLONET.analyzePM.R \
Examples/Small/ConfigurationFile/CLONET.analyzePM.Config.R

>./Tools/CLONET.analyzeRR.R \
Examples/Small/ConfigurationFile/CLONET.analyzeRR.Config.R

>./Tools/CLONET.analyzeGeneSCNA.R                                    \
Examples/Small/ConfigurationFile/CLONET.analyzeGeneSCNA.Config.R

>./Tools/CLONET.LoadAndCLeanDataForTEP.R \
Examples/Small/ConfigurationFile/CLONET.LoadAndCLeanDataForTEP.Config.R

>./Tools/CLONET.createEvolutionGraph.R \
Examples/Small/ConfigurationFile/CLONET.createEvolutionGraph.Config.R
```

Successful run will generate a series of file in the folder `Examples/Small/Results/`:

- betaTable.txt Tab delimited file containing estimated percentage of neutral reads for each input genomic segment;
- globalAdmTable.txt Tab delimited file containing per sample estimate of global DNA admixture and its variability range;
- ploidyTable.txt Tab delimited file containing per sample estimate of ploidy;
- clonalityTable.txt Tab delimited file containing estimated clonality for each input genomic segment;
- genesSCNA_clonalityTable.txt Tab delimited file containing estimated clonality for gene;
- PM_clonalityTable.txt Tab delimited file containing estimated clonality for point mutation;
- RR_clonalityTable Tab delimited file containing estimated clonality for structural rearrangement;
- TumorEvolutionPath.graphml Computed tumor evolution path in graphml format.

CLONET

This script takes a set of samples pileups and a corresponding set of SCNAs and computes

1. the percentage of neutral reads within each SCNA
2. the ploidy of each sample
3. the global DNA admixture of each sample
4. the clonality of each input genomic segment

Configuration file requires the following:

- *path_to_CLONET_functions* The absolute path to CLONET folder
- *output_DIR* The output directory to save the result tables
- *sampleInfoFile* Information about sample names
- *segmentListFile* List of segment for each sample in a 5 column tab separated file without header. Columns are chromosome, segment start position, segment end position, log R value of the segment, sample name
- *pileup_dir* Folder with informative SNPs for each sample
- *PaPI_Suffix* Suffix of the informative SNPs pileup
- *errorTable_file* Path to pre-computed error table
- *minCoverage* Minimum tumor coverage to consider an informative SNP as valid
- *NsamplesToProcessInParallel* Number of samples to process in parallel
- *perSampleCores* Number of cores assigned to the analysis of each sample
- *min_nsnps* Minimum number of informative SNPs for a genomic segment to be considered
- *min_cov* Minimum mean coverage of a genomic segment to be considered
- *equalCN.betaThr* Minimum value of beta above which the two alleles are present in the same number (cn equal to 1+1 or 2+2 or 3+3 ...) to account for ref map bias
- *maxHomoDels* Homozygous deletions threshold (change only if you know what you are doing)
- *deletionsLog2Levels* Log ratio (R) values for a putative valid mono-allelic deletion used to compute global DNA admixture
- *alphaPar* Percentage of deletions to compute Adm.global variability interval (change only if you know what you are doing)
- *clonalityThreshold* Threshold on clonality to call a genomic segment clonal or subclonal (default value is recommended)
- *betaThreshold* Threshold on beta value to asses copy number (default value is recommended)

The configurations file in Examples/Small reports default configuration parameters that were tested in many challenging cases.

CLONET TOOLS

The main CLONET.R script performs mandatory analysis on copy number data. Additional, analyses can be run for point mutations and/or rearrangements.

ANALYSIS OF POINT MUTATIONS

The script in `Tools/CLONET.analyzePM.R` allows a user to submit a list of PMs with reference and alternative read counts and returns a table with clonality of each PMs. Configuration file requires to specify:

- *path_to_CLONET_functions* The absolute path to CLONET.R folder
- *PMreadCounts_file* Tab separated file containing the list of PM with the tumor read counts
- *ClonalityTable_file* Clonality table produced by CLONET.R script
- *OutTable_file* Output PM clonality table file name
- *p.val* alpha value for binomial test
- *observedAF* observed peak of the binomial distribution to account reference mapping bias
- *minCov* minimum coerage to consider a PM valid
- *minAltReads* minimum coverage of the alternative base to consider a PM valid
- *ncores* number of cores to use

ANALYSIS OF STRUCTURAL REARRANGEMENTS

The script in `Tools/CLONET.analyzeRR.R` allows a user to submit a list of RRs with the total number of reads that span both sides of each breakpoint (i.e., dRanger/Breakpointer output) and returns a table with clonality of each RRs. Configuration file requires to specify:

- *path_to_CLONET_functions* The absolute path to CLONET.R folder
- *RR_readCount_file* Tab separated file containing the list of RR with RRs with the total number of reads that span both sides of each breakpoint
- *ClonalityTable_file* Clonality table produced by CLONET.R script
- *OutTable_file* Output RR clonality table file name
- geneList_file table with genes genomic coordinates to associate to each breakpoint
- *p.val* alpha value for binomial test
- *ncores* number of cores to use

ANALYSIS PER GENE SOMATIC COPY NUMBER ABERRATIONS

The script in `Tools/CLONET.analyzeGeneSCNA.R` computes integer copy number status and associated clonality value for a list of genes. The output is a tab delimited file that for each gene reports its integer copy number value and the associated clonality plus information about the genomic segments (identified by segmentation) intersecting the gene. Configuration file requires specifying:

- *path_to_CLONET_functions* The absolute path to CLONET.R folder
- *ClonalityTable_file* Clonality table produced by CLONET.R script
- geneList_file table with genes genomic coordinates to analyze

- adm.global_file Global DNA admixture table produced by CLONET.R script
- *OutTable_file* Table to save the result
- *ncores* number of cores to use

CLEAN DATA

The three tables produced by the scripts described above feed the tumor evolution engine of CLONET. Before building the evolution path the data has to be cleaned and organized. The script in `Tools/CLONET.LoadAndCLeanDataForTEP.R` creates an RData object that constitutes the basic input to build the tumor evolution path. Configuration file requires specifying:

- *path_to_CLONET_functions* The absolute path to CLONET.R folder
- *cnData_file* Clonality table output of CLONET.R
- *pmData_file* Clonality table output of CLONET.analyzeRR.R
- *rrData_file* Clonality table output of CLONET.analyzeRR.R
- *geneTableSCNA_file* Clonality table output of CLONET.analyzeGeneSCNA.R
- *roundSig* number of significative decimals
- *lossCNs* definition of CN loss status
- *gainCNs* definition of CN gain status
- *Adm.global.max* maximum allowed global DNA admixture
- *samplesToRemove* List of bad samples by visual inspection of beta vs logR plot
- *maxError* Maximum valid variability range of clonality
- *minError* Minimum valid variability range of clonality
- *minSNPs* Minimum number of informative SNPs for a genomic segment to be considered valid
- *minCov* Minimum mean coverage of a genomic segment to be considered valid
- *minCov.BP* Minimum allowed coverage for RRs
- *minCov.PM* minimum coerage to consider a PM valid
- *minAltCov.PM* minimum coverage of the alternative base to consider a PM valid
- Adm.global_file Global DNA admixture table produced by CLONET.R script
- *OutRdata_file* Out Rdata file
- *ncores* number of cores to use

TO CREATE A TUMOR EVOLUTION PATH

The script in `Tools/ CLONET.createEvolutionGraph.R` computes the tumor evolution path for a given set of aberrations analyzed with CLONET. The output is a standard graphml object that can be visualized and analyzed with many tools. Configuration file requires specifying:

- *path_to_CLONET_functions* The absolute path to CLONET.R folder

- *CleanData_RData* Data cleaned for tumor evolution path construction with CLONET.LoadAndCLeanDataForTEP.R tool
- OutGraph_file Path to output graphml file
- ListOfGenes_file List of genes to analyze (one column without header)
- Tab separated file with information to annotate each gene in the graphml object (e.g., cytoband, GO terms)
- clonalStatus Define clonality status of early events: clonal or clonal plus uncertain.clonal
- subclonalStatus Define clonality status of late events: subclonal or subclonal plus uncertain.subclonal
- *ncores* number of cores to use