nature portfolio

https://doi.org/10.1038/s41596-021-00567-5

Supplementary information

Detecting chromosomal interactions in Capture Hi-C data with CHiCAGO and companion tools

nature portfolio

https://doi.org/10.1038/s41596-021-00567-5

Supplementary information

Detecting chromosomal interactions in Capture Hi-C data with CHiCAGO and companion tools

Supplementary Table 1. Description of CHiCAGO functions.

Function	Description	Parameters	
chicagoPipeline	Runs data through the CHiCAGO pipeline.	cd	chicagoData object
		outprefix	NULL, or a character string. If NULL, diagnostic plots are outputted to the current plotting device. If a character string, then pdfs will be generated for a series of diagnostic plots, in files of form "[outprefix]_[plotname]".
		printMemory	Set to TRUE for memory diagnostics.
copyCD	Copies a chicagoData object.	cd	chicagoData object
defaultSettings	Returns the default settings used for a CHiCAGO experiment.	NA	
estimateBrownian Component	Estimates the dispersion, and adds a Bmean column giving the expected number of Brownian reads. Usually, the dispersion is calculated on a subsample of baits. Its size is taken from brownianNoise.subset. To use the entire dataset set this to NA.	cd	chicagoData object
estimateDistFun	Estimates the function describing how the expected number of counts decreases with increasing distance. Default Settings: - 75 distance bins - a cubic fit. For distances outside of the bin boundaries, the function is log-linear, with continuity of f and its first derivative on the log-scale.	cd	chicagoData object
		method	Choice of method: "cubic" fits a cubic function with linear extrapolation, on a log-log scale.
		plot	Logical – if TRUE output a diagnostic plot.
		outfile	Optional file name to save plot on disk.
estimateTechnicalNoise	Calculates the expected technical noise based on trans read pairs.	cd	chicagoData object
		plot	Logical - if TRUE output a diagnostic plot.
		outfile	Optional file name to save plot on disk.
exportResults	Exports the results from a chicagoData object to disk, or to a Genomic Interactions object.	cd	chicagoData object
		outfileprefix	Prefix for output files.
		scoreCol	Column of intData(cd) that contains the score.
		cutoff	The score cutoff.
		b2bcutoff	Optional alternative score cutoff for bait-to-bait interactions.

		format	File format(s) to output. If multiple formats are supplied as a vector, then all of these formats will be outputted. Supported formats are: "seqMonk", "interBed", "washU_text" and "washU_track" (advanced users).
		order	Option to ordered by position or by score.
		removeMT	Logical – if TRUE, discards interactions with mitochondrial DNA.
getPvals	Calculates a P-value for each observation using a Delaporte model	cd	chicagoData object
getScores	Converts p-values into CHiCAGO scores, using p-value weighting.	cd	chicagoData object
		method	Either "unweighted" or "weightedRelative" (recommended).
		includeTrans	Logical – if FALSE, discards trans interactions.
		plot	Plot a diagnostic plot.
		outfile	Optional file name to save plot on disk.
getSkOnly	Finds s_k scaling factors for multiple samples.	files	Path to .chinput files
		cd	Blank chicagoData object for reference, usually created with setExperiment.
mergeSamples	Merges several chicagoData objects together, summarising their counts into a normalised value.	cdl	List of chicagoData objects.
		normalise	Logical – if TRUE, uses the normalisation procedure specified by mergeMethod; if FALSE, takes the mean number of reads.
		NcolOut	Column identifying normalised counts.
		NcolNormPrefix	Column prefix for sample specific normalised counts.
		mergeMethod	Options: - “weightedMean” - NcolOut is the weighted mean of the sample-wise counts adjusted by the samples’ respective scaling factors s_k; - “mean” - sample-specific counts are first normalised by dividing by s_k, and NcolOut is computed as their mean.
		repNormCounts	Report normalised counts for each replicate This option is on by default when mergeMethod = "mean” but it is also available with "weightedMean".
modifySettings	Modifies the settings in a chicagoData object.	cd	chicagoData object.
		designDir	Path to designDir.
		settings	Named list with settings.

		settingsFile	Path to a settings file. One row per setting: <name> <whitespace> <value>.
normaliseBaits	Calculate normalisation factors s_j for each bait.	cd	chicagoData object
		normNcol	Column identifying normalised counts.
		plot	Logical – if TRUE, output a diagnostic plot.
		outfile	Optional file name to save plot on disk.
normaliseOtherEnds	Compute s_i normalisation factors for other ends, and normalised counts.	cd	chicagoData object
		Ncol	Column intData(cd) identifying counts normalised by bait (i.e. output from normaliseBaits).
		normNcol	Column identifying counts normalised by other ends .
		plot	Logical – if TRUE, output a diagnostic plot.
		outfile	Optional file name to save plot on disk.
overlapFragWithFeatures	Computes overlap between other-ends from a chicagoData object and a set of genomic features.	x	chicagoData object or data table (data.table) containining other end IDs.
		folder	Path to folder with feature files.
		list_frag	List of feature files (e. g. H3K4me1, CTCF, DHS etc.). These files must have a bed format, with no header. Each element of the list must be named.
		position_otherEnd	Path to file with restriction fragments coordinates and the corresponding IDs. position_otherEnd must be specified when x is not a chicagoData object.
		sep	Field separator character in file containing the coordinates of the restriction fragments (called by position_otherEnd).
peakEnrichment4Features	Computes how many other ends from a chicagoData object, that engage in SI, overlap with a set of genomic features. As a control, this function also samples different sets of interactions from the non-significant pool and computes their overlap with genomic features. Results are returned in a table and plotted in a barplot. Samples have the same size as the number of SI called and they follow the same distribution of bait-other end distances.	x1	chicagoData object or data table (data.table) containining other end IDs.
		folder	Path to folder with feature files.
		list_frag	List of feature files (e. g. H3K4me1, CTCF, DHS etc.). These files must have a bed format, with no header. Each element of the list must be named.
		no_bins	Number of bins to divide the range of colname_dist (after colname_dist has been trimmed according to min_dist and max_dist). This determines the number of interactions sampled according to distance from bait.

		sample_number	Number of samples to be used in the permutation test. Recommended: ≥100.
		position_otherEnd	Path to file with restriction fragments coordinates and the corresponding IDs. position_otherEnd must be specified when x is not a chicagoData object.
		colname_dist	Column identifying bait-other distances. Needs to be specified if x is not a chicagoData object.
		score	score cutoff.
		colname_score	Column identifying interaction scores.
		min_dist	Minimum bait-other end distance required. When set to NULL and trans is set to TRUE, cis interactions are discarded from the analysis.
		max_dist	Maximum bait-other distance required.
		sep	Field separator character in file containing the coordinates of the restriction fragments (called by position_otherEnd).
		filterB2B	Logical – if TRUE bait-to-bait interactions are discarded.
		b2bcol	Column identifying bait-to-bait interactions.
		unique	Logical – if TRUE duplicated other ends are discarded.
		plot_name	Optional file name to save plot on disk.
		trans	Logical – if TRUE enrichment is computed for trans interactions. Additionally, if min_dist is NULL, cis interactions are discarded.
		plotPeakDensity	Logical – if TRUE, plot the density of interactions with distance. Only applies to cis interactions.
plotBaits	Plots the read counts around baits.	cd	chicagoData object.
		pcol	Column identifying interaction scores.
		Ncol	Column identifying counts.
		n	Number of baits to plot.
		baits	Bait IDs to plot.
		plotBaitsNames	Logical – if TRUE, the bait names, rather than IDs, are plotted.

		plotBprof	Logical – if TRUE, displays line with expected Brownian noise per distance.
		plevel1, plevel2	cutoff used on the pcol column. plevel1 should be the most stringent.
		outfile	Optional file name to save plot on disk.
		removeBait2bait	Logical – if TRUE, bait-to-bait interactions are not plotted.
		width, height,	Passed through to pdf
		maxD	Maximum (linear) distance each side of the bait to plot. If NULL, include the whole chromosome.
		bgCol, lev1Col, lev2Col	Colours to be used for background points, and for the two stringency levels defined by plevel1 and plevel2, respectively.
		bgPch, lev1Pch, lev2Pch	Plotting character for background points, and for points exceeding the two stringency levels defined by plevel1 and plevel2, respectively. Specified as per pch in points.
		…	Additional arguments passed to plot
plotDistFun	Estimates the function describing how the expected number of counts decreases with increasing distance.	cd	chicagoData object.
		…	Additional arguments passed to plot
readAndMerge	Wraps readSample() and mergeSamples() on multiple files.	files	Character vector containing the locations of the files to read in.
		cd	chicagoData object.
		…	Further arguments passed to mergeSamples.
readSample	Reads input data from a file, into a chicagoData object.	file	Path of input file.
		cd	chicagoData object.
setExperiment	Creates a template CHiCAGO experiment object. This should be the first function called.	designDir	Path to design directory.
		settings	Named list with settings. Overrides both settingsFile, or def.settings.
		settingsFile	Path to a settings file. One row per setting: <name> <whitespace> <value>. Overrides def.settings.
		def.settings	Default settings

Supplementary Table 2: Downsampled data on OSF platform for testing the CHiCAGO pipeline

Data Type	**File name(s)**	**File type**	**Link to Directory**	**Description**
Raw CHi-C data, HaCaT cells	HaCaT_unst_rep1_CHiC_DS20M_R1.fastq.gz HaCaT_unst_rep2_CHiC_DS20M_R1.fastq.gz HaCaT_unst_rep2_CHiC_DS40M_R1.fastq.gz HaCaT_unst_rep2_CHiC_DS40M_R2.fastq.gz	FASTQ	https://osf.io/kmfj4/	Raw data for testing alignment (e.g. HiCUP)
Raw CHi-C data, MyLa cells	MyLa_rep1_CHiC_DS20M_R1.fastq.gz MyLa_rep1_CHiC_DS20M_R2.fastq.gz MyLa_rep2_CHiC_DS20M_R1.fastq.gz MyLa_rep2_CHiC_DS20M_R2.fastq.gz	FASTQ	https://osf.io/xm9an/	Raw data for testing alignment (e.g. HiCUP)
CHi-C BAM files, HaCaT	HaCaT_unst_rep1_CHiC_DS20M_R1_2.hicup.bam HaCaT_unst_rep2_CHiC_DS40M_R1_2.hicup.bam	BAM	https://osf.io/g8qcm/	Aligned and filtered CHi-C reads. Can be used to test bam2chicago.sh
CHi-C BAM files, MyLa	MyLa_rep1_CHiC_DS20M_R1_2.hicup.bam MyLa_rep2_CHiC_DS20M_R1_2.hicup.bam	BAM	https://osf.io/hq4zj/	Aligned and filtered CHi-C reads. Can be used to test bam2chicago.sh
CHiCAGO input files, HaCaT	HaCaT_unst_rep1_CHiC_DS20M_HindIII_GWAS_sharedChr.chinput HaCaT_unst_rep2_CHiC_DS40M_HindIII_GWAS_sharedChr.chinput	Chinput	https://osf.io/wsc69/	Chinput files for testing full CHiCAGO pipeline
CHiCAGO input files, MyLa	MyLa_rep1_CHiC_DS20M_HindIII_GWAS_sharedChr.chinput MyLa_rep2_CHiC_DS20M_HindIII_GWAS_sharedChr.chinput	Chinput	https://osf.io/wsc69/	Chinput files for testing full CHiCAGO pipeline
CHiCAGO Design files	GRCh37_HindIII_sharedChr.rmap HindIII_GWAS_sharedChr.baitmap	rmap; baitmap	https://osf.io/sx7fu/	Design files accompanying the CHi-C data; used to make the npbp, npb and poe files that are required for the CHiCAGO pipeline
CHiCAGO data files, HaCaT	HaCaT_rep1_DS20M_HindIII_GWAS_sharedChr.Rds HaCaT_rep2_DS40M_HindIII_GWAS_sharedChr.Rds HaCaT_DS_reps_HindIII_GWAS_sharedChr.Rds	Rds	https://osf.io/b9p3v/	CHiCAGO output for individual replicates (rep1, rep2) or for both replicates processed together (reps). Can be used to test CHiCAGO functions.
CHiCAGO data files, MyLa	MyLa_rep1_DS20M_HindIII_GWAS_sharedChr.Rds MyLa_rep2_DS20M_HindIII_GWAS_sharedChr.Rds MyLa_DS_reps_HindIII_GWAS_sharedChr.Rds	Rds	https://osf.io/b9p3v/	CHiCAGO output for individual replicates (rep1, rep2) or for both replicates processed together (reps). Can be used to test CHiCAGO functions.
Feature files	all_feat.txt E047_CD8_H3K27ac_hg19.bed E047_CD8_H3K27me3_hg19.bed E047_CD8_H3K4me1_hg19.bed ENCFF151HKM_NHEK_H3K27me3_hg19.bed ENCFF898SZF_NHEK_H3K4me1_hg19.bed ENCFF943CBQ_NHEK_H3K27ac_hg19.bed	txt; bed	https://osf.io/ajtpv/	Histone modifications from ENCODE (NHEK) or Roadmap (CD8) to test for feature enrichment against HaCaT and MyLa, respectively. Accompanying all_feat.txt specifies file names of features.