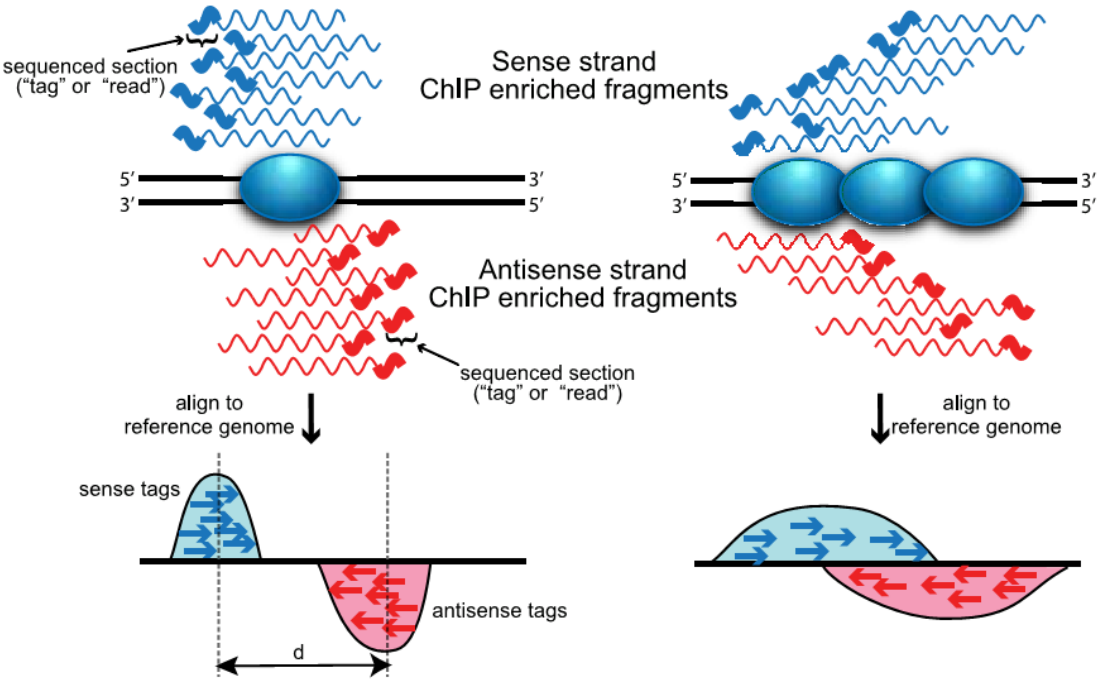
A protocol using homer to analyze Hi-C data

Created by Bao-Wen Yuan (yuanbaowen521@126.com)



Though this is a picture from CHIP-seq, I want to show that, reads=tags

Before step 1， you should have *.sam files generated by mapping. # Unlike paired end sequencing for other techniques like genomic resequencing or RNA-Seq, where you might expect the 2nd read to be located within the general vicinity of the first read, read-pairs from Hi-C should be processed independently (e.g. do not use Tophat, bowtie, bwa or any short read alignment software in "paired-end" mode - each read should be mapped independently!)

I'd recommend mapping the same way you would ChIP-Seq data, allowing only unique genomic matches

Download sequences for use with homer, use

the configureHomer.pl script. http://homer.salk.edu/homer/introduction/configure.html

To get a list of available packages: **perl /path-to-homer/configureHomer.pl -list**

Install packages: **perl /path-to-homer/configureHomer.pl -install hg19** (to download the hg19 version of the human genome)

STEP 1：Creating pair end Tag Directories, quality control, and read filtering for Hi-C data (makeTagDirectory) http://homer.salk.edu/homer/interactions/HiCtagDirectory.html

(1)makeTagDirectory <HiC-Unprocessed> <reads1-1.sam,reads1-2.sam reads2-1.sam,reads2-2.sam> [-illuminaPE] [-tbp 1]

#complete the lengthy process of merging all the paired end reads into an "unprocessed" tag directory, and remove all clonal reads

#<HiC-Unprocessed> 是 <OutputTagDirectory>， 就是把输出结果放入的地方，是个文件夹.文件夹中包含了：*.tags.tsv (extended format to store the paired end reads, only each read will get 2 entries in the appropriate *.tags.tsv so that the reads can be indexed on each chromosome in one file); tagInfo.txt (inform HOMER to treat the experiment as a paired-end file. *note: the tagInfo.txt file will report the "Total Tags" value as 2x the total number interactions. This is because HOMER is storing each PE tag twice - the software later adjusts this number to reflect this redundancy, but keep this in mind when interpreting this number*); petag.LocalDistribution.txt (shows the relationship between the 5' ends of the paired reads, to determine the fragment size of the Hi-C fragments used for sequencing); petag.FreqDistribution.txt contains a histogram describing the fraction of paired-end reads that are found at different distances from one another

#<reads1-1.sam,reads1-2.sam reads2-1.sam,reads2-2.sam> 是 mapping 结果文件(sam 格式),pair-end 用逗号隔开，无空格；不同 pair-end 对之间用空格隔开(在实际操作中，需要在含有 sam 文件的目录中运行此程序，写全路径不行)

#[-illuminaPE] the original bowtie output files for Illumina sequencing often contains a 0 or 1 at the end to delineate the read source(HOMER matches reads between alignment files by matching their names), so it need to remove the last character before trying to match. currently, data ususally downloaded from SRA, they often have the same name for each paired end read, and this parameter is always needn't.

#[-tbp 1] only consider read pairs with the exact same ends once(if reads-pairs start from the exact same places twice in the data, they are likely clonal and a result of over-sequencing your library or some other artifact)

(2)cp -r HiC-unprocessed HiC-processed # make a copy of the directory

(3)makeTagDirectory <HiC-processed> <-update> <-genome hg19> <-restrictionSite AAGCTT> [-rsmis #] <-both> [-removePEbg] [-removeSelfLigation] [-removeRestrictionEnds] [-removeSpikes 10000 5]

filter reads based on restriction sites

HiC-processed: contain the files to be handled

 # -update: tells the program not to look for new data files but reprocess the existing tag directory

-genome hg19: specifying the genome, so that homer can figure out where the restriction sites are located

-restrictionSite AAGCTT: to see the distribution of reads around restriction site. During Hi-C, "background" ligation events may occur after sonication as they seem to occur between random fragments of DNA with no regard to restriction site used for the Hi-C assay. In theory, all reads should be in the vicinity of the restriction site used for the assay, otherwise they are noise.

-removeSelfLigation: remove reads if their ends form a self ligation with adjacent restriction sites

-removeRestrictionEnds : Removes reads if one of their ends starts on a restriction site

#-rsmis <#> how many mismatches are tolerated in the sites if the restriction site has a lot of star activity. By default it only looks for perfect restriction sites.

#both: Only keep reads if both ends of the paired-end read have a restriction site within the fragment length estimate 3' to the read

-removePEbg: remove paired-end reads that are likely continuous genomic fragments or re-ligation events. read pair separated less than **1.5x** the sequencing insert fragment length are removed.
-removeSpikes <#size> <# fold>: remove PE reads that originate from regions of unusually high tag density
#produce: petagRestrictionDistribution.<seq>.mis<#>.txt: display the distribution of reads relative to the restriction site

STEP 2: Hi-C Background Models: Normalizing Genomic Interactions for Linear Distance and Read Depth http://homer.salk.edu/homer/interactions/HiCBackground.html

Primary Source of Technical Bias in Hi-C Interaction Counts

1) Read depth per region: Since Hi-C is an unbiased assay of genomic structure, we expect to observe equal read coverage across the genome. However, factors such as the ability to map reads uniquely (e.g. density of genomics repeats), the number of restriction sites, and genomic duplications/structural variation in the experimental sample will all influence the number of reads.

2) Linear distance between loci along the chromosome: If two loci are along the same polymer/chromosome of DNA, the loci are constrained with respect to one another independent of any specific structures adopted by the chromosome. More to the point, loci closely spaced along a chromosome are almost guaranteed to be 'near' one another if for no other reason their maximal separation is the length of DNA between them. As a result, closely spaced loci will have very high Hi-C read counts, regardless of their specific conformation. This is generally true of all 3C-based assays.

3) Sequencing Bias: GC% bias, ligation preferences during library construction, normal sequencing problems.

4) Chromatin compaction: This source of bias isn't really an artifact as it reflects biologically meaningful configuration of chromatin fibers. The idea is that different types of chromatin environments tend to "fold" in different ways, with heterochromatic/inactive/inert chromatin displaying a different average interaction frequencies as a function of distance than euchromatin/active/permissive regions. However, when looking for specific interaction in a particular chromatin environment, it can be useful to understand the properties of your local region of DNA to help interpret what is meaningful or just normal for that type of chromatin environment.

Normalizing Hi-C Data-- normalize the data for sequencing depth and distance between loci

$e_{ij} = f(i-j)\ (n^*_i)(n^*_j)/N^*$

#eij: the expected Hi-C reads between two given regions (i&j)

f: the expected frequency of Hi-C reads as a function of distance

N*: estimated total number of reads

n*: the estimated total number of interaction reads at each region

(1) **analyzeHiC <HiC Tag Directory> -res <#> -bgonly -cpu <#>**

creating hi-c background models that saves important parameters from normalization to speed up analysis. The background model only has to be computed once for a given resolution.

-res <#> represents how frequent the genome is divided up into regions, namely the binned size

-superRes <#> represents how large the region is expanded when counting reads. Usually the "-res <#>" should be smaller than the "-superRes <#>"

the principle advantage to this (-res and -superRes) is that you don't penalize features that span boundaries

-force force the creation of a new model

-bgonly create a background model and then quit. the background model can be custom

-cpu <#> use multiple processors, because the process is very slow, so better to use multiple processors.

#the output file has only one: <HiC Tag Directory>/HiCbackground_#res_bp.txt

(1) analyzeHiC <HiC Tag Directory> <-createModel model.output.file> <-p peak.file> -vsGenome <-res #> <-cpu 8> <-bgonly>

creating custom background models to see the general properties in certain regions

-createModel: command to creat model

-p: a peak file that specify the custom region

-vsGenome: compare to the rest of the genome

STEP 3 Creating and Normalizing Hi-C interaction Matrices http://homer.salk.edu/homer/interactions/HiCmatrices.html

(1) mergePeaks CTCF.peaks <-d #> > newCTCFPeaks.txt

#this step used to matching resolution of peak/BED file and resolution of analyzeHiC

In the case when distance between peaks are located less than the resolution apart from one another, the two peaks will give essentially the same results and cause redundancy

(2) analyzeHiC <Hi-C Tag Directory> [-res #] [-superRes #] { [-chr chr.name] [-start #] [-end #] [-pos chr:start-end] [-chr2 chr.name] [-start2 #] [-end2 #] [-pos2 chr:start-end] -vsGenome] } { [-p peak/bed.file] [-p2 peak/bed.file]} -raw/-simpleNorm/-norm/-logp/-expected/-rawAndExpected expectedMatrix.filename/-corr/-nomatrix > outputMatrixFile.txt

-res: default is 10Mb, use # to specify; the same as -superRes

#specifying specific regions to analyze:

-chr <chr name> : will restrict analysis to this chromosome

-start <#> : starting position for analysis

-end <#> : end position for analysis

-pos <chr:start-end> : UCSC browser formatted position, can instead the -chr/-start/-end

-chr2 <chr name> : will restrict analysis to this chromosome

-start2 <#> : starting position for analysis

-end2 <#> : end position for analysis

-pos2 <chr:start-end> : UCSC browser formatted position, instead the -chr2/-start2/-end2

-vsGenome : compare to the rest of the genome

For this option (-chr/-start/-end/-pos), HOMER will chop up the region into resolution sized chunks and perform the analysis. I.e. you provide a locus you want a detailed picture of. But for peak file, HOMER will only consider the center of the peak file and the surrounding "resolution-sized" region. It will not chop-up your peaks into resolution-sized chunks (unless you specify the "**-chopify**" option).

-p <peak/BED file> : peak/BED file to use to search for interactions between

-p2 <peak/BED file> : A second peak/BED file for non-symetrical matricies

#output information options:

-raw: Outputs the raw interactions counts between the regions

-simpleNorm: Outputs the ratio of observed to expected interactions (normalizing for sequencing coverage only)

-norm: (default) Outputs the ratio of observed to expected interactions (normalizing for both sequencing coverage and linear distance between loci) only). This attempts to take into account the "proximity ligation" effect, where adjact regions are expected to have interactions regardless of the specific 3D genomic structure.

-logp: Outputs the natural log of the p-value describing the likelihood of observeing the actual number of interactions relative to the expected number of interactions between to two loci. This is calculated conservatively as a cumulative binomial distribution.

-expected: Outputs the expected number of interactions instead of the observed number of interactions. To get the "simpleNorm" version of the expected interactions, include "-simpleNorm" too.

-rawAndExpected <expectedMatrix filename>: Outputs both a raw interaction matrix and the expected number of interaction matrix. The raw interactions are sent to *stdout* (or the file specified with "-o") and the expected interactions are sent to the given filename.

-corr (can be used with any of the above options): Instead of outputing the matrix as is, the value of each cell is replaced with the Peason's Correlation Coefficient between the row and column. This can be useful as it adds transitive information to the problem. Instead of just using the number of interaction

that directly span between to loci, the correlation option will consider how each region interacts with all of the other loci too. If they have similar interaction profiles, the correlation will be high (i.e. 1). If "-logp" or "-expected" is used, those values are the ones that will be used for the correlation calculation. The matrix must be symmetric for this option to work.

-nomatrix: Don't create a matrix

If viewing normalized data of observed vs. expected log2 ratios, in the matrix interacting regions are positive (one color) and non-interacting regions are negative (a different color); By default, analyzeHiC sorts the chromosomes numerically and places the X, Y, and M at the end of the list

(3) **analyzeHiC <HiC Tag Directory> <-size #> <-hist #> <-p peak/BED.file> > output2dHistogram.txt**

this step used to create 2D Histograms of Interactions

<-size #>: a total size of the interaction matrix

<-hist #>: the histogram resolution

STEP 4 Sub-nuclear Compartment Analysis (PCA/Clustering)

http://homer.salk.edu/homer/interactions/HiCpca.html

Principal Compartment Analysis (PCA) of Hi-C Data: the basic idea is to redefine the coordinate system such the data can be "described" with as few dimensions as possible. This axes of the coordinate system are referred to as the principle components, and the "1st" component is found such that it can be used to describe or discriminate as much of the system as possible, with the "2nd" component describing as much of the remaining variance as possible, and so on. We can then consider each region with respect to their values along the first couple principal components to get a simplified view of the data.

(1) runHiCpca.pl <outputPrefix> <HiC Tag Directory> [-res #] [-superRes #] [-pc #] [-active peak/BED.file]/[-inactive peak/BED.file]/[-genome genome.name] [-std #] [-corrDepth #] [-min #] [-cpu #]

[-res #] (default: 50000) / [-superRes #] (default: 100000)

[-pc #]: number of principal components to return (default: 1) .

[-active peak/BED.file]: seed regions of "active" chromatin to use when assessing the proper sign of the PC1 results

[-inactive peak/BED.file]: seed regions of "inactive" chromatin to use when assessing the proper sign of the PC1 results

[-genome genome.name]: If no seed regions given, this will use the TSS file as active seeds

[-corrDepth #]: number of expected reads needed per data point when calculating correlation, default: 3

[-std #]: exclude regions with sequencing depth exceeding # std deviations, default: 4. Remove regions that may throw off the PCA calculation

[-min #]: exclude regions with sequencing depth less than this fraction of mean, default: 0.15. Remove regions with low coverage that don't behave as well during the PCA analysis

[-cpu #]: number of CPUs to use, default: 1

#generate two files:

<prefix>.PC1.txt - peak file containing coordinates along the first 2 principal components

<prefix>.PC1.bedGraph - UCSC upload file showing PC1 values across the genome

Usually the values for the PC1 are the most informative. PC2 usually describes the the location along the chromosome (e.g. near telomere vs. centromere, or different chromosome arms). Normally, positive PC1 regions reflecting "active/permissive" chromatin and negative PC1 regions indicative of "inactive/inert" chromatin

(2) getHiCcorrDiff.pl <outputPrefix> <HiC Tag Directory1> <HiC Tag Directory2> [-res #] [-superRes #] [-corrDepth #] [-std #] [-min #] [-maxDist #] [-cpu #]

this step used to directly comparing two Hi-C experiments, parameters almost the same as above

[-min #]: default: 0.1

[-maxDist #]: maximum distance around regions to calculate similarity metrics, default: none

Output two files:

<prefix>.corrDiff.txt - peak file containing correlation values for each

<prefix>.corrDiff.bedGraph - UCSC upload file showing correlation values across the genome

Because the precise qualitative nature of this association may differ slightly between experiments, it is recommended to directly compare the interaction profiles between experiments rather than simply looking at PC1 values

(3) annotatePeaks.pl <peak.name> <genome> <-size #> [-hist #] <-bedGraph exp1.PC1.bedGraph exp2.PC1.bedGraph> > output.histogram.txt

this used to see distribution of PC1 values around certain peak

(4) findHiCCompartments.pl *.PC1.txt [-opp] [-thresh #] [-peaks] > compartments.txt

this used to find PC1 based compartments. Regions of continuous positive or negative PC1 values set the stage for identifying "compartments"

[-opp]: return inactive/negative regions. Without this, return active/positive regions.

[-thresh #]: threshold for active regions, default: 0

[-peaks]: output as peaks/regions, not continuous domains)

(4) findHiCCompartments.pl <-bg *.PC1.txt > <-diff #> <-corr corrDiff.txt> <-corrDiff #> >compartments.txt

this used to identify regions that change their compartment between experiments

<-bg *.PC1.txt>: specify a background experiment's PC1 results

<-diff #>: minimum difference between PC1 values to define region as different, default: 50

<-corr corrDiff.txt>: specify a correlation difference file

<-corrDiff #>: maximum correlation for a region to be defined as different, default: 0.4

---

(1) Clustering Regions Based on their Interactions

analyzeHiC has two types of clustering routines to help identify sets of regions that are "related" in 3D-space:

-cluster : Clustering of regions regardless of genomic locations (pure clustering based on interaction frequency)

-clusterFixed : Clustering of regions based on adjacent, linear regions on the chromosome (for finding "linear domains")

analyzeHiC <HiC Tag Directory> <-chr #> <-res #> [-raw/-simpleNorm/-norm/-logp/-corr] [-o out.name] -cluster/-clusterFixed > outputmatrix.txt

[-o out.name]: use <filename> as the prefix for the clustering files instead of "out"

this can output files "out.cdt" and "out.gtr". Java Tree View can open the "out.cdt" file and view the clustering result

---

STEP 5 Finding Significant Interactions in Hi-C Data

http://homer.salk.edu/homer/interactions/HiCinteractions.html

The premise behind finding significant interactions is simple enough: Look for pairs of regions that have more Hi-C reads between them than would be expected by chance

(1) analyzeHiC <HiC Tag Directory> [-res #] [-superRes #] [-chr/-start/-end/-pos/-p] [-maxDist #/-minDist #] [-center] [-pvalue] [-zscore #] <-interactions significantInteractions.txt> [-nomatrix] [-cpu #]

[-chr/-start/-end/-pos/-p]: specify the regions to analyze, same as before

[-maxDist #/-minDist #]: limiting the space that is searched

[-center]: re-center the regions on the average of the position of the Hi-C reads participating in the interaction, thus replace the coordinates of the regions in the output file. Namely, increasing accuracy by centering interactions.

[-pvalue #]: default report all interactions with a p-value less than 0.001. modify the threshold.

[-zscore #]: modify z-score cutoff

(2) analyzeHiC <HiC Tag Directory1> <-res #> <-ped <HiC Tag Directory2>> <-interactions significantInteractions.txt> [-nomatrix]

this used to find differential interactions between two Hi-C experiments

(3) findHiCInteractionsByChr.pl <HiC Tag Directory> <-res #> <-superRes #> [-cpu #] > outputInteractions.txt

this used to find intra-chromosomal interactions genome-wide at high resolution

<-res #> : default: 2000/<-superRes #> : default: 10000

[-minDist #] : Minimum distance between regions to consider for an interaction (default: -superRes value)

[-maxDist #] : Maximum distance between regions to consider for an interactions (default: 10,000,000)
[-pvalue #] : default: 0.01/[-zscore #] : default: 1.5
[-cpu #] : default: 1
[-ped <background HiC directory>] : Will quantify background experiment reads at significant interactions.
[-std #] : exclude regions with sequencing depth exceeding # std deviations, default: 4)
[-min #] : exclude regions with sequencing depth less than this fraction of mean, default: 0.2)
(4)

Continue updating