Instruction on running cis-X

Table of Contents

Note: all the genomic coordinates should be in hg19 (GRCh37).

Please contact Yu Liu (liuyu@scmc.com.cn or yu.liu@sjtu.edu.cn) for questions.

1. Dependencies

cis-X requires the following tools. Please make sure they are in your `$PATH`.

- Perl (https://www.perl.org/). cis-X was tested with perl ver 5.10.1 or up.
  - Data::Compare (https://metacpan.org/pod/Data::Compare).
- R (https://www.r-project.org/). cis-X was tested with R ver 3.1.0 or up.
  - multtest (https://www.bioconductor.org/packages/release/bioc/html/multtest.html).
- Java (http://www.oracle.com/technetwork/java/javase/downloads/index.html). cis-X was tested with java ver 1.8.0_66 or up.
- FIMO[1] (http://meme-suite.org/tools/fimo). cis-X was developed with MEME ver 4.9.0.
- twoBitToFa (https://genome.ucsc.edu/goldenpath/help/twoBit.html).
- bedtools (https://bedtools.readthedocs.io/en/latest/).

The following files need to be prepared as reference files for cis-X run. These files are not distributed with cis-X, and need to be put in `refs/external` folder under cis-X home directory. We provide cis-X seed command as part of cis-X to assist preparing these reference files. See https://github.com/stjude/cis-x/tree/master/src/seed for more details.

- "GRCh37-lite.2bit". GRCh27-lite.fa.gz file was downloaded from http://genome.wustl.edu/pub/reference/GRCh37-lite/ and transformed into GRCh37-lite.2bit by faToTwoBit from http://hgdownload.soe.ucsc.edu/admin/exe/.
- "hg19_refGene" and "hg19_refGene.bed". RefSeq gene predictions in hg19 could be downloaded from UCSC table browser (http://genome.ucsc.edu/cgi-bin/hgTables). The "hg19_refGene.bed" file is a chopped version of "hg19_refGene", generated with script "hg19_refGene2bed.pl" provided along with cis-X.
- "HOCOMOCOv10_HUMAN_mono_meme_format.meme". Matrix for transcription factor binding motif models in MEME format from HOCOMOCO[2]. cis-X was developed with version 10 models at http://hocomoco11.autosome.ru/downloads_v10.
- "HOCOMOCOv10_annotation_HUMAN_mono.tsv". Annotation table for the motif models downloaded from http://hocomoco11.autosome.ru/downloads_v10.
- "hESC.combined.domain.hg19.bed". Topologically associating domains defined by Hi-C technology in Human ES Cell (H1) in previous study[3]. The topological domains are downloaded from http://chromosome.sdsc.edu/mouse/hi-c/download.html. The original TAD was in hg18 and need to be lifted to hg19 through liftOver from UCSC genome browser (http://genome.ucsc.edu/cgi-bin/hgLiftOver).
- "cancer_gene_census.txt". The Cancer Gene Census downloaded from COSMIC database[4] (https://cancer.sanger.ac.uk). Register was required for download this file. Please save the file into text format with the above file name after downloaded. cis-X was developed with COSMIC v87.
- "ImprintGenes.txt". Imprinting gene list could be downloaded from http://www.geneimprint.com/site/genes-by-species.
- "roadmapData.enhancer.merged.111.bed", "roadmapData.promoter.merged.111.bed" and "roadmapData.dyadic.merged.111.bed". These three files are DNaseI-accessible regulatory regions annotated from Roadmap Epigenomics project[5]. Files for individual cell lines are

downloaded from https://egg2.wustl.edu/roadmap/web_portal/DNase_reg.html#delieation and merged into these three files, for enhancer, promoter and dyadic separately. The merged files contain total 8 columns, with the first 6 columns as same as in the original bed files and additional 2 with the cell line name and tissue of origins.

2. Input files for cis-X

- List of single nucleotide markers. A tab delimitated text file contains both germline polymorphisms and somatic variants present in the tumor genome, A header line is mandatory with exact column name as listed below. This file could be output from multiple variant calling tools. The demo data set provided with cis-X was generated from Bambino[6].

Column name	Content
Chr	Chromosome location for the marker.
Pos	Genomic position for the marker.
Chr_Allele	Reference allele.
Alternative_Allele	Alternative allele.
reference_tumor_count	Reference allele count in tumor genome.
alternative_tumor_count	Alternative allele count in tumor genome.
reference_normal_count	Reference allele count in matched normal genome.
alternative_normal_count	Alternative allele count in matched normal genome.

- CNV/LOH regions. A tab delimitated text file following the bed format, contains all the genomic regions carrying copy number variation (CNV) or loss of heterozygosity (LOH) in the tumor genome. A header line is mandatory in the file with minimal 3 columns including “chrom”, “loc.start” and “loc.end”. A file contains only header line need to be provided if there is no copy number alterations or loss of heterozygosity in the genome under analysis. Any marker sit in the CNV/LOH region will be filtered out. CONSERTING[7] is used to generate the CNV/LOH regions in the demo data set.
- Somatic SNV/Indel list. A tab delimitated text file contains somatic sequence mutations present in the genome under analysis, including single nucleotide variants and small insertion/deletions. A header line is mandatory with the columns “chr”, “pos”, “ref allele”, “mutant allele” and “mutation type” (either snv or indel). In a case of simple indel, use ‘-‘ for the reference or alternative allele that is null or empty. The somatic mutations in the demo data set was analyzed by Bambino[6] and followed by post processing as previously described[8]. Note that the coordinate used in indel is "after" the inserted sequence. A file contains only header line is needed even if no somatic SNV/Indel is present in the sample under analysis.
- Somatic CNV. A tab delimitated text file contains the genomic regions with somatic acquired copy number aberrations in the cancer genome. A header line is required, with columns “chr”, “start”, “end” and “log2Ratio”. The somatic copy number aberration in the demo data set is generated with CONSERTING[7]. A file contains only header line is needed even if no somatic SNV/Indel is present in the sample under analysis.
- Somatic SV. A tab delimitated text file contains the somatic acquired structural variants present in the cancer genome. A header line is mandatory, with the following columns. The

structural variants used in the demo data set is generated by CREST[9]. A file contains only header line is needed even if no somatic SNV/Indel present is in the sample under analysis.

Column name	Content
chrA	Chromosome of left breakpoint.
posA	Genomic location of left breakpoint.
ortA	Strand orientation of left breakpoint.
chrB	Chromosome of right breakpoint.
posB	Genomic location of right breakpoint.
ortB	Strand orientation of right breakpoint.

- RNA-seq bam file and index file. The demo data set was aligned to hg19 with StrongArm (Rusch M, et al, manuscript in preparation), as previously described[10]. Alternatively, we recommend user to use STAR[11] for RNA-seq alignment.
- Gene expression table. A tab delimitated text file contains gene level expression for the tumor under analysis, in FPKM (the number of fragments per kilobase of transcript per million mapped reads). The FPKM should be generated with HTseq-count[12], following the script provided along with cis-X, as previously described[13]. This is important so that the data is comparable with the values presented in the gene specific reference expression matrix generated from a larger cohort. GENCODE v19 (https://www.gencodegenes.org/human/release_19.html) should be used as the gene model.
- Gene specific reference expression matrix. A set of 3 expression reference files for the outlier high expression test, including the genes expression reference for “white-list” genes, “bi-allelic” reference and expression from the whole cohort, as described in the online methods section of the manuscript. Currently, we have built this reference set for pediatric T-lineage acute lymphoblastic leukemia (T-ALL) and acute myeloid leukemia (AML) from previous study[14]. These files are packed along with cis-X. It is important to use a reference expression matrix matching the tissue type of the tumor under analysis. Please follow section 7 “Gene specific reference expression matrix” below on how to build customized reference expression matrix.

3. Output files for cis-X

The following files will be delivered as result. The “sample_id” provided by user is used as prefix for the output files by default. The results files are located in a directory named with [PREFIX], together with intermediate files for debugging purpose under [PREFIX]/working_space/.

File name	Content
[PREFIX].cisActivated.candidates.txt	cis-activated candidates in the genome under analysis by combining allelic specific expression analysis and outlier high expression analysis.
[PREFIX].cisActivated.candidates.byRuns.txt	cis-activated candidates in the genome revealed by ASE-runs and outlier high expression analysis.
[PREFIX].sv.candidates.txt	Structural variant candidates predicted as the causal for the cis-activated genes in the regulatory territory.
[PREFIX].cna.candidates.txt	Copy number aberrations predicted as the causal for the cis-activated genes in the regulatory territory.
[PREFIX].snvindel.candidates.txt	SNV/Indel candidates predicted as functional. The predicted transcription factors were listed here. The mutations were also annotated for known regulatory elements reported by Epigonomic Roadmap project by collecting 111 cell lines.
[PREFIX].OHE.results.txt	Raw results for outlier high expression test.
[PREFIX].ase.gene.model.fdr.txt	Raw results for gene level allelic specific expression test (all genes).
[PREFIX].ase.combine.WGS.RNAseq .goodmarkers.binom.txt	Raw results for single marker based allelic specific expression test.

Columns in each file are listed below.

1) [PREFIX].cisActivated.candidates.txt

Column	Name	Content
1	gene	Gene accession number.
2	gsym	Gene symbol.
3	chrom	Chromosome.
4	strand	Strand.
5	start	Transcription start position.
6	end	Transcription end position.
7	cdsStartStat	CDS status.

8	cdsEndStat	CDS status.
9	markers	Number of heterozygous markers in this gene.
10	ase_markers	Number of heterozygous markers showing ASE.
11	average_ai_all	Average B-allele frequency (BAF) difference between RNA and DNA for all heterozygous markers.
12	average_ai_ase	Average BAF difference between RNA and DNA for ASE markers.
13	pval_all_markers	P-value for each marker in ASE test.
14	pval_ase_markers	P-value for ASE markers in ASE test.
15	ai_all_markers	BAF difference between RNA and DNA for each marker.
16	ai_ase_markers	BAF difference between RNA and DNA for ASE marker.
17	tag_all_markers	If the marker is in diploid region or CNV/LOH region.
18	maf_rna_all_markers	BAF for each marker in RNA-seq.
19	comb.pval	Combined p-value for ASE test.
20	mean.delta	Avaerage BAF difference between RNA and DNA for all markers.
21	rawp	Raw p-value for ASE test.
22	Bonferroni	Adjusted p-value for ASE test (Bonferroni single-step).
23	ABH	Adjusted p-value for ASE test (Benjamini & Hochberg).
24	FPKM	FPKM value.
25	loo.source	Source of reference expression matrix used in outlier high expression test.
26	loo.cohort.size	Number of cases in the reference expression matrix for this gene.
27	loo.tstatistic	t-statistic from leave-one-out outlier high expression test.
28	loo.qval	Significance for OHE test corrected with the null distribution.
29	loo.rank	Ranking for the case under analysis among the reference cases.
30	impringting.stats	Imprinting status of the gene.
31	candidate.group	Status of the gene combining ASE and outlier test.
32	description	Status of the gene in COSMIC database.

2) [PREFIX].cisActivated.candidates.byRuns.txt

Column	Name	Content
1	Run_ID	ID for ASE-run.
2	Chrom	Chromosome for the ASE-run.
3	Start	Genomic coordinates where the ASE-run starts.
4	End	Genomic position where the ASE-run ends.
5	Length	Length of the ASE-run, in bp.

6	Num_Markers	Number of markers included in the ASE-run.
7	Tag_Markers	Tag for each marker in the ASE-run. Possible values include ‘e’ for ‘extreme’ markers with all the reads come from the same allele, or all but one read come from a single allele and the marker meets the significance of statistical test for imbalanced transcription; ‘E’ for ‘extreme’ markers with all but one read from a single allele but could not reach the significance of statistical test for imbalanced transcription; ‘s’ for markers with significant imbalanced transcription but not ‘e’ or ‘E’; ‘f’ for markers of all other kinds.
8	Genes_overlap	Genes with minimal 80% overlap with the ASE-run.
9	Candidates	Genes also reach the outlier high expression test to be nominated as cis-activated candidates.

3) [PREFIX].sv.candidates.txt

Column	Name	Content
1	left.candidate.inTAD	cis-activated candidate near left breakpoint.
2	right.candidate.inTAD	cis-activated candidate near right breakpoint.
3	chrA	Chromosome for left breakpoint.
4	posA	Genomic position for left breakpoint.
5	ortA	Strand for left breakpoint.
6	chrB	Chromosome for right breakpoint.
7	posB	Genomic position for right breakpoint.
8	ortB	Strand for right breakpoint.
9	type	Type of translocation.

4) [PREFIX].cna.candidates.txt

Column	Name	Content
1	candidate.inTAD	Candidate cis-activated by the CNA.
2	chr	Chromosome.
3	start	Left genomic position of the CNA.
4	end	Right genomic position of the CNA.
5	logR	Log ratio of the CNA.

5) [PREFIX].snvindel.candidates.txt

Column	Name	Content
1	chrom	Chromosome.
2	pos	Genomic position.
3	ref	Genotype for reference allele.
4	mut	Genotype for mutant allele.
5	type	Type of the mutation (SNV or Indel).

6	target	cis-activated candidate.
7	dist	Distance between mutation and the transcription start site of the target gene.
8	tf	Transcription factors predicted to have the binding motif introduced by the mutation.
9	EpiRoadmap_enhancer	Enhancer regions overlap with the mutation (Roadmap Epigenomics Project).
10	EpiRoadmap_promoter	Promoter regions overlap with the mutation (Roadmap Epigenomics Project).
11	EpiRoadmap_dyadic	Dyadic enhancer/promoter regions overlap with the mutation (Roadmap Epigenomics Project).
12	User_Annot	Annotation with user provided BED file.

6) [PREFIX].OHE.results.txt

Column	Name	Content
1	Gene	Gene symbol.
2	fpkm.raw	FPKM value.
3	size.bi	Number of cases in the “bi-allelic” reference cohort.
4	p.bi	P-value in outlier test using “bi-allelic” reference cohort.
5	rank.bi	Ranking of expression level in the case under analysis compared to the “bi-allelic” reference cohort.
6	tstatistic.bi	t-statistic from leave-one-out outlier high expression test, using “bi-allelic” as reference cohort.
7	qval.bi	Significance for OHE test corrected with the null distribution.
8	size.cohort	Number of cases in the “whole” reference cohort.
9	p.cohort	P-value in outlier test using “whole” reference cohort.
10	rank.cohort	Ranking of expression level in the case under analysis compared to the “whole” reference cohort.
11	tstatistic.cohort	t-statistic from leave-one-out outlier high expression test, using “whole” reference cohort.
12	qval.cohort	Significance for OHE test corrected with the null distribution.
13	size.white	Number of cases in the “white list” reference cohort.
14	p.white	P-value in outlier test using “white list” reference cohort.
15	rank.white	Ranking of expression level in the case under analysis compared to the “white list” reference cohort.
16	tstatistic.white	t-statistic from leave-one-out outlier high expression test, using “white list” reference cohort.
17	qval.white	Significance for OHE test corrected with the null distribution.

7) [PREFIX].ase.gene.model.fdr.txt

Column	Name	Content
1	gene	Gene accession number.
2	gsym	Gene symbol.
3	chrom	Chromosome.
4	strand	Strand.
5	start	Transcription start position.
6	end	Transcription end position.
7	cdsStartStat	CDS status.
8	cdsEndStat	CDS status.
9	markers	Number of heterozygous markers in this gene.
10	ase_markers	Number of heterozygous markers showing ASE.
11	average_ai_all	Average B-allele frequency (BAF) difference between RNA and DNA for all heterozygous markers.
12	average_ai_ase	Average BAF difference between RNA and DNA for ASE markers.
13	pval_all_markers	P-value for each marker in ASE test.
14	pval_ase_markers	P-value for ASE markers in ASE test.
15	ai_all_markers	BAF difference between RNA and DNA for each marker.
16	ai_ase_markers	BAF difference between RNA and DNA for ASE marker.
17	tag_all_markers	If the marker is in diploid region or CNV/LOH region.
18	maf_rna_all_markers	BAF for each marker in RNA-seq.
19	comb.pval	Combined p-value for ASE test.
20	mean.delta	Avaerage BAF difference between RNA and DNA for all markers.
21	rawp	Raw p-value for ASE test.
22	Bonferroni	Adjusted p-value for ASE test (Bonferroni single-step).
23	ABH	Adjusted p-value for ASE test (Benjamini & Hochberg).

8) [PREFIX].ase.combine.WGS.RNAseq.goodmarkers.binom.txt

Column	Name	Content
1	chrom	Chromosome.
2	pos	Genomic position.
3	ref	Genotype of reference allele.
4	mut	Genotype of non-reference allele.
5	cvg_wgs	Coverage of the marker from WGS.
6	mut_freq_wgs	Non-reference allele fraction in WGS.
7	cvg.rna	Coverage of the marker from RNA-seq.
8	mut_freq_rna	Non-reference allele fraction in RNA-seq.
9	ref.1	Read count of reference allele in RNA-seq.

10	var	Read count of non-reference allele in RNA-seq.
11	pvalue	P-value from binomial test.
12	delta.abs	Absolute difference of non-reference allele fraction between WGS and RNA-seq.

4. Running cis-X on local machine

- Download cis-X from https://www.stjuderesearch.org/site/lab/zhang/cis-x, and extract the package to a working directory. This directory will be referred to as `$CIS_X_HOME`.
- Install the dependencies and prepare the reference files as described in “Dependencies”. Tools are expected to be available in `PATH`.
- Set up the paths for both cis-X and its dependencies.

```
$ CIS_X_HOME=[Your cis-X source directory]
$ V2M_HOME=$CIS_X_HOME/vendor/variants2matrix
$ export PATH=$CIS_X_HOME/bin:$V2M_HOME/bin:$PATH
$ export PERL5LIB=$V2M_HOME/lib/perl:$PERL5LIB
$ export CLASSPATH=$(ls $V2M_HOME/lib/java/* | paste -sd ":" -)
```

- Use the `cis-X run` command to run cis-X.

```
$ cis-X run \
  -s $SAMPLE_ID \
  -o $WORKING_DIR \
  -l $MARKERS \
  -g $CNV_LOH_REGIONS \
  -b $BAM \
  -e $GENE_EXPRESSION_TABLE \
  -m $SOMATIC_SNV_INDEL \
  -v $SOMATIC_SV \
  -c $SOMATIC_CNV \
  -d $DISEASE \
  -a $CNV_LOH_ACTION \
  -w $MIN_COVERAGE_WGS \
  -r $MIN_COVERAGE_RNAseq \
  -f $FPKM_THRESHOLD_CANDIDATE \
  -u $USER_ANNOT \
  -h $CHR_STRING \
  -t $TAD_INFO
```

See "Input files for cis-X" for more details on how inputs are prepared. Please provide FULL path to the input files from the command line above.

`$DISEASE` can be one of "TALL" or "NBL", both which are distributed with cis-X under `$CIS_X_HOME/refs/diseases`. See "Gene specific reference expression matrix" to generate custom matrices.

`$CNV_LOH_ACTION` refers to the behavior of cis-X regarding the markers in CNV/LOH regions. Options are “keep” or “drop”. Default: keep

`$MIN_COVERAGE_WGS` and `$MIN_COVERAGE_RNAseq` refers to the minimal coverage in WGS and RNA-seq required for a marker to be included in the analysis Default: 10 for both.

`$FPKM_THRESHOLD_CANDIDATE` is the FPKM threshold for nominate cis-activated candidate. Only genes with FPKM greater or equal to this threshold will be nominated for further analysis. Default: 5.

`$USER_ANNOTFPKM` refers to the user provided annotation file in BED format for annotate the candidate snv/Indels. Default: NotSpecified.

`$CHR_STRING` refers to if a ‘chr’ prefix is included in the RNA-seq BAM file. User could check this information with ‘samtools view -H $BAM-File’. Options are “TRUE” or “FALSE”. Default: TRUE.

`$TAD_INFO` refers to a tab separated 3-column BED format file contains TAD information defining the regulatory territory used in noncoding variant analysis. The TAD structure predefined in human ES cell (H1) by Hi-C data was used as default and was provided along the cis-X package.

- Results are saved to `$WORKING_DIR/$SAMPLE_ID`. See "Output files for cis-X" for details of the results.

5. Running cis-X with demo dataset.

To test cis-X, a demo dataset is provided at https://www.stjuderesearch.org/site/lab/zhang/cis-x. It includes demo data for a single T-ALL (`demo/data`) and the necessary external references (`demo/refs`). It can be used after moving the demo data to their expected directories. Install the dependencies and references as described in “Dependencies” before the test.

```
$ tar xf cis-X-demo.tar.gz
$ mv demo/ref/* $CIS_X_HOME/refs/external
$ mv demo/data/* .
$ cis-X run \
  -s SJALL018373_D1 \
  -o $(pwd) \
  -l $(pwd)/SJALL018373_D1.test.wgs.markers.txt \
  -g $(pwd)/SJALL018373_D1.test.wgs.cnvloh.txt \
  -b $(pwd)/SJALL018373_D1.test.RNAseq.bam \
  -e $(pwd)/SJALL018373_D1.test.RNASEQ_all_fpkm.txt \
  -m $(pwd)/SJALL018373_D1.test.mut.txt \
  -v $(pwd)/SJALL018373_D1.test.sv.txt \
  -c $(pwd)/SJALL018373_D1.test.cna.txt \
  -d TALL \
  -a drop \
  -w 10 \
  -r 10 \
  -f 5
```

Upon completion, results are output to `$(pwd)/SJALL018373_D1` with the following files:

- `SJALL018373_D1.cisActivated.candidates.txt`
- `SJALL018373_D1.sv.candidates.txt`
- `SJALL018373_D1.cna.candidates.txt`
- `SJALL018373_D1.snvindel.candidates.txt`

6. cis-X in Docker

To avoid manually set up cis-X and its dependencies, a Dockerfile is also provided to run cis-X in a container via Docker. The container version of cis-X can be downloaded at https://www.stjuderesearch.org/site/lab/zhang/cis-x.

Install Docker (https://docs.docker.com/install) for your system first. After run Docker and follow the steps below. Note that cis-X requires at least 4 GiB of RAM. Larger memory may be required depend on the size of your input files. This resource can be increased for the desktop version of Docker by going to Docker preferences > Advanced > Memory.

- Install Docker for your platform: https://docs.docker.com/install/
- Start Docker. From `$CIS_X_HOME`, build the container image. This installs all the required dependencies and external references. This step can take 10~20 minutes, depends on the internet connection. This step only need to be run once.

```
$ docker build --tag cis-x .
```

- Prepare the required reference files as described in “Dependencies”. This could be done with cis-X seed, within the container image built from above command. Here we are using “refs” from current directory as the place for the downloaded references files as an example. After this step, cis-X is ready to run through Docker.

```
$ REFS_DIR=$(pwd)/refs
$ mkdir -p $REFS_DIR/external
$ docker run \
  --mount type=bind,source=$REFS_DIR/external,target=/refs/external \
  cis-x \
  seed \
  /refs/external
```

- To run the cis-X image with the demo dataset, run the following command from directory contains the demo data (parent of data/ and ref/).

```
$ docker run \
  --mount type=bind,source=$(pwd)/data,target=/data,readonly \
  --mount \
    type=bind,source=$(pwd)/ref,target=/app/refs/external,readonly \
  --mount type=bind,source=$(pwd),target=/results \
  cis-x \
  run \
   -s SJALL018373_D1 \
   -o /results \
   -l /data/SJALL018373_D1.test.wgs.markers.txt \
   -g /data/SJALL018373_D1.test.wgs.cnvloh.txt \
   -b /data/SJALL018373_D1.test.RNAseq.bam \
   -e /data/SJALL018373_D1.test.RNASEQ_all_fpkm.txt \
   -m /data/SJALL018373_D1.test.mut.txt \
   -v /data/SJALL018373_D1.test.sv.txt \
   -c /data/SJALL018373_D1.test.cna.txt \
   -d TALL \
   -a drop \
   -w 10 \
   -r 10 \
   -f 5
```

- From the previous example, running cis-X in Docker is very similar to running it in your own environment. The image can create a container that is not limited to just the demo dataset but your own data as well.

  In the three `mount` flags, the only changes needed are to the `source` directories (highlighted in red below). The three lines represent 1) directory contains your input files, 2) directory contains the reference files and 3) your output directory. They can point to any absolute path on the local filesystem and do not have to match the target name, e.g,

```
--mount type=bind,source=$HOME/data_dir,target=/data,readonly \
--mount type=bind,source=/tmp/references,target=/app/refs/external,readonly \
--mount type=bind,source=$(pwd)/cis-x-out,target=/results \
```

Note that the results directory must exist before running the command. The next arguments are the same as described in section "4. Running cis-X on local machine". Any paths here are relative to the target, not local filesystem path. For example, mounting `$HOME/research` and with an input located at `$HOME/research/sample-1/markers.txt`, the corresponding argument is `/data/sample-1/markers.txt`.

The following template is the entire command to run, with variables showing what needs to be set.

```
$ docker run \
    --mount type=bind,source=$DATA_DIR,target=/data,readonly \
    --mount type=bind,source=$REFS_DIR,target=/app/refs/external,readonly \
    --mount type=bind,source=$RESULT_DIR,target=/results \
    cis-x \
    run \
     -s $SAMPLE_ID \
     -o /results \
     -l /data/$MARKERS \
     -g /data/$CNV_LOH_REGIONS \
     -b /data/$BAM \
     -e /data/$GENE_EXPRESSION_TABLE \
     -m /data/$SOMATIC_SNV_INDEL \
     -v /data/$SOMATIC_SV \
     -c /data/$SOMATIC_CNV \
     -d $DISEASE \
     -a $CNV_LOH_ACTION \
     -w $MIN_COVERAGE_WGS \
     -r $MIN_COVERAGE_RNA \
     -f $FPKM_THRESHOLD_CANDIDATE
```

7. Gene specific reference expression matrix

The reference expression matrix is used for the outlier high expression test in cis-X. This includes a set of three files.

- `exp.ref.entire.txt`, includes all cases in the cohort without any filter.
- `exp.ref.bi.txt`, includes only cases showing bi-allelic expression for a given gene.
- `exp.ref.white.txt`, includes only wild type cases without known noncoding regulatory variants for a given gene.

cis-X will carry out independent test with the three reference expression matrixes. Only genes included in the matrix will be tested in each run. The reference expression matrix could be left empty with only a header line. For example, if you don't have any prior knowledge for noncoding regulatory variants in your cohort, the `exp.ref.while.txt` file could be left empty with only header line. The reference files provided along with cis-X could be used as template to prepare the customized files. Note that using unfiltered exp.ref.entire.txt file as the only reference file will result higher false negative rate for cis-activated candidate genes during analysis.

We provided scripts with cis-X to generate the `exp.ref.bi.txt` reference file from customized data. This requires a cohort of cases with both RNA-seq and DNA-seq (whole genome sequencing or whole exome sequencing) for each case. The steps are listed below.

a) Prepare cis-X as described above in section "4. Running cis-X on local machine".
b) Prepare a configuration file with the full path to the input files, with each line for one case, contains four columns with a header line: (1) sample id, (2) path to "List of single nucleotide markers" as described in cis-X input section, (3) path to RNA-seq bam file (the index file should be in the same directory with the bam file), (4) path to CNV/LOH file as described in cis-X input section.
c) Run on command line:

```
$ cis-X ref-exp prepare $CONFIG $WORKING_DIR $CHR_STRING
```

`$CONFIG` and `$WORKING_DIR` stand for the configure file generated above and output directory, with full path. `$CHR_STRING` stands for if 'chr' characters were present in the RNA-seq BAM file (either TRUE or FALSE). This will generate a file named "`cis-X.refexp.step1.commands.sh`" containing a batch of command lines. Depend on the working environment, user should submit these jobs to cluster. Wait till these jobs are completed before continuing to the next step.

d) Run on command line:

```
$ cis-X ref-exp generate $CONFIG $WORKING_DIR $GENE_EXPRESSION_TABLE
```

The gene expression table is the same as described in cis-X input section, but contains FPKM values for all the samples under analysis. Upon run completed, a folder called "`refexp`" will be generated under your working directory contains `exp.ref.bi.txt`, `precal.tvalue.bin_gt1.txt` and an empty `exp.ref.white.txt`. User can copy these to `$CIS_X_HOME/refs/dieseases/$DISEASE` as gene specific reference expression matrix. Please also copy the gene expression table provided in this step to the `$CIS_X_HOME/refs/diseases/$DISEASE` directory and rename as `exp.ref.entire.txt`, representing the matrix for the entire cohort.

8. Scripts in assist of cis-X analysis

- VariantsToTable (https://software.broadinstitute.org/gatk/documentation/tooldocs/3.8-0/org_broadinstitute_gatk_tools_walkers_variantutils_VariantsToTable.php) could be used to extract specific fields from a vcf file to tab delimited file for cis-X.
- hg19_refGene2bed.pl (under `$CIS_X_HOME/src/other/`). Transform the "`hg19_refGene`" file to "`hg19_refGene.bed`" file. Run this script with "`hg19_refGene`" in the same directory.
- mergeData_geneName.pl (under `$CIS_X_HOME/src/other/`). Used after HTseq-count to calculate the gene expression in FPKM. Usage: `perl -w mergeData_geneName.pl [counts.*.txt from HTseq-count] [v19 GTF file]`. The GTF file in v19 is also provided along with the demo data, at

`$demo_dir/ref/gencode.v19.annotation_level1and2_withChrM.gtf_gene_size_byGeneName.txt` and should be used as a template for user specified GTF files. The output file "`RNAseq_GENCODEv19_all_fpkm.txt`" should be used for cis-X.

9. Troubleshooting

- An error was noticed during installation of MEME suite with gcc > 5.4.0. We tested one solution for this is to add a patch during compiling meme 4.9.0, as below.

```
$ wget http://meme-suite.org/meme-software/4.9.0/meme_4.9.0_4.tar.gz
$ tar xf meme_4.9.0_4.tar.gz
$ cd meme_4.9.0
$ patch -p1 \
    < $CIS_X_HOME/src/other/meme_glam2_fix_new_gcc.patch
$ ./configure \
    --prefix=/usr/local \
    --with-url=http://meme-suite.org \
    --enable-build-libxml2 \
    --enable-build-libxslt
$ make
$ make install
```

- "realpath: command not found".

  This is due to realpath not exist in your system. Please follow the command line below.

```
cd $CIS_X_HOME
ENV_HOME=$CIS_X_HOME/vendor/env
mkdir -p $ENV_HOME/bin
gcc -O2 -o $ENV_HOME/bin/realpath src/other/realpath.c
PATH=$CIS_X_HOME/bin:$ENV_HOME/bin:$PATH
```

10. References

1. Grant, C.E., Bailey, T.L. & Noble, W.S. FIMO: scanning for occurrences of a given motif. *Bioinformatics* **27**, 1017-1018 (2011).
2. Kulakovskiy, I.V. et al. HOCOMOCO: expansion and enhancement of the collection of transcription factor binding sites models. *Nucleic Acids Res* **44**, D116-125 (2016).
3. Dixon, J.R. et al. Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature* **485**, 376-380 (2012).
4. Forbes, S.A. et al. COSMIC: somatic cancer genetics at high-resolution. *Nucleic Acids Res* **45**, D777-d783 (2017).
5. Roadmap Epigenomics, C. et al. Integrative analysis of 111 reference human epigenomes. *Nature* **518**, 317-330 (2015).
6. Edmonson, M.N. et al. Bambino: a variant detector and alignment viewer for next-generation sequencing data in the SAM/BAM format. *Bioinformatics* **27**, 865-866 (2011).
7. Chen, X. et al. CONSERTING: integrating copy-number analysis with structural-variation detection. *Nat Methods* **12**, 527-530 (2015).
8. Zhang, J. et al. The genetic basis of early T-cell precursor acute lymphoblastic leukaemia. *Nature* **481**, 157-163 (2012).
9. Wang, J. et al. CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nat Methods* **8**, 652-654 (2011).
10. Parker, M. et al. C11orf95-RELA fusions drive oncogenic NF-kappaB signalling in ependymoma. *Nature* **506**, 451-455 (2014).
11. Dobin, A. et al. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29**, 15-21 (2013).
12. Anders, S., Pyl, P.T. & Huber, W. HTSeq--a Python framework to work with high-throughput sequencing data. *Bioinformatics* **31**, 166-169 (2015).
13. Liu, Y. et al. The genomic landscape of pediatric and young adult T-lineage acute lymphoblastic leukemia. *Nat Genet* **49**, 1211-1218 (2017).
14. Ma, X. et al. Pan-cancer genome and transcriptome analyses of 1,699 paediatric leukaemias and solid tumours. *Nature* **555**, 371-376 (2018).