QUICK START GUIDE

HiCUP is a bioinformatics pipeline for processing Hi-C data. The pipeline receives FASTQ data which is mapped against a reference genome and filtered to remove frequently encountered experimental artefacts. The pipeline produces paired read files in SAM/BAM format, each read pair corresponding to a putative Hi-C di-tag. HiCUP also produces summary statistics at each stage of the pipeline to assist with quality control and identify potential experimental problems and help refine the protocol for the future.

The HiCUP pipeline is not intended for determining contact probabilities between loci, which requires normalising the data for a range of biases.

Another Babraham Institute project, SeqMonk, enables the visualisation of Hi-C data (www.bioinformatics.babraham.ac.uk/projects/seqmonk).

REQUIREMENTS

HiCUP requires a working version of Perl and Bowtie installed on your machine. For further details go to:

http://www.perl.org/

http://bowtiebio.sourceforge.net/index.shtml

For full functionality HiCUP requires the R to be installed. (For more details of R go to http://www.r-project.org).

INSTALLATION

HiCUP is written in Perl and executed from the command line. To install HiCUP copy the hicup_v0.X.Y.tar.gz file into a HiCUP installation folder and extract all files by typing:

```
tar -xvzf hicup_v0.X.Y.tar.gz
```

Check after extracting that the Perl scripts are executable by all.

RUNNING HICUP

1) CREATE A DIGESTED REFERENCE GENOME

To filter out common experimental artefacts, HiCUP requires the positions at which the restriction enzyme(s) used in the protocol cut the genome. The script hicup_digester creates this reference genome digest file. The example below will digest all '.fa' files in the current working directory with HindIII. The digest output file will be labelled as the genome 'Human_GRCh37'. Provide the full path to hicup_digester or the sequence files to be digested if they are not in the current working directory.

Execute the script:

```
hicup_digester -g Human_GRCh37 -1 A^AGCTT,HindIII *.fa
```

2) CREATE BOWTIE INDICES

HiCUP uses the aligner Bowtie to map sequences to the reference genome, requiring the construction of Bowtie indices. **These indices need to be constructed from the same reference genome files as used by the hicup_digester script.** In the command line enter ‘bowtie-build’ to construct the indices, followed by a comma-separated list of the sequence files and then a space followed by the name of the output indices. For example:

```
bowtie-build 1.fa,2.fa,…,MT.fa Human_GRCh37
```

Further instructions on building a Bowtie index can be found at:

http://bowtie-bio.sourceforge.net/manual.shtml#the-bowtie-build-indexer

3) RUN THE HICUP PIPELINE

To start the pipeline, copy the configuration file hicup.conf and then modify the copy as required, adhering to the format below:

```
#Directory to which output files should be written (optional parameter)
#Set to current working directory by default
Outdir:

#Number of threads to use
Threads: 1

#Suppress progress updates (0: off, 1: on)
Quiet:0

#Retain intermediate pipeline files (0: off, 1: on)
Keep:0

#Compress outputfiles (0: off, 1: on)
Zip:1

#Path to the alignment program Bowtie (include the executable Bowtie filename)
Bowtie: /usr/local/bowtie/bowtie

#Path to R (required for full functionality)
R: /bi/apps/R/3.0.3/bin/R

#Path to the reference genome indices
#Remember to include the basename of the genome indices

Index: /data/public/Genomes/Mouse/NCBIM37/Mus_musculus.NCBIM37
```

```
#Path to the genome digest file produced by hicup_digester
Digest: Digest_Mouse_genome_HindIII_None_12-32-06_17-02-2012.txt

#FASTQ file format (phred33-quals, phred64-quals, solexa-quals or solexa1.3-quals)
#If not specified, HiCUP will try to determine the format automatically by analysing
#one of the FASTQ files. All input FASTQ will assumed to be in this format
Format: phred33-quals

#Maximum di-tag length (optional parameter)
Longest: 800

#Minimum di-tag length (optional parameter)
Shortest: 150

#FASTQ files to be analysed, paired files on adjacent lines
s_1_1_sequence.fastq
s_1_2_sequence.fastq

s_2_1_sequence.fastq
s_2_2_sequence.fastq

s_3_1_sequence.txt.fastq.gz
s_3_2_sequence.txt.fastq.gz
```

Enter the following text in the command line to run the whole HiCUP pipeline using the parameters specified in the configuration file:

```
hicup -c hicup.conf
```

The '-c' flag is used to specify the configuration filename. Also remember to provide the full path to the hicup script or the configuration file if they are not in the current working directory.

The HiCUP pipeline may take several hours to complete, so it may be preferable to run as a background job that will not terminate when ending the session. To do this, run the pipeline with the command:

```
nohup hicup -c hicup.conf &
```

TEST DATASET

To confirm HiCUP functions correctly on your system please download the test Hi-C dataset from the HiCUP homepage. The test files 'test_dataset1.fastq' and 'test_dataset2.fastq' both contain human Hi-C reads in Phred33 FASTQ format.

1) Extract the tar archive before processing:

```
tar -xvzf hicup_v0.X.Y.tar.gz
```

2) If necessary, create Bowtie indices of the Homo sapiens GRCh37 genome (chromosomes 1…22, X, Y and MT).

Example command:

```
bowtie-build 1.fa,2.fa,…,MT.fa human_GRCh37
```

3) Using hicup_digester create a reference genome of Homo sapiens GRCh37 all chromosomes (1…22, X, Y and MT) digested with HindIII (A^AGCTT).

Example command:

```
hicup_digester –g Human_GRCh37 -1 A^AGCTT,HindIII *.fa
```

4) Edit a copy of the hicup.conf configuration file so it has the following parameters:

```
Zip: 1
Keep: 0
Threads: 1

Bowtie: [Path to Bowtie on your system]
Digest: [Path to digest file on your system]
Index: [Path to Bowtie indices on your system]
R: [Path to R on your system]

Format: phred33-quals
Shortest: 150
Longest: 800

test_dataset1.fastq | test_dataset2.fastq
```

5) Run the pipeline using the command:

```
hicup –c hicup.conf
```