HiTC - Exploration of High Throughput 'C' experiments

Nicolas Servant

May 3, 2016

1 Introduction

Chromosome Capture Conformation (3C) was first introduced by **?** ten years ago. The 3C technique aims in detecting physical contact between pairs of genomic loci and is now widely used to detect intrachromosomal (cis) and interchromosomal (trans) interactions between genes and regulatory elements. The development of the 3C-based techniques has changed our vision of the nulcear oragnization (see **?** for a review).
With the development of high throughput analyses, and in particular second-generation sequencing, the 3C has been adapted to study in parallel physical interactions between many loci, and thus increase the scale at which interactions between genomic loci can be detected (4C - Circular 3C, **?**, **?**; 5C - 3C Carbone Copy, **?**). More recently, this technique was further extended to obtain detailed insights into the general three-dimensional arrangements of complete genomes (Hi-C, **?**).

While the use of high-throuput 'C' techniques is expected to increase in the coming years, it also creates some new statistical and bioinformatics challenges. In this way, publicly available bioinformatics tools, as well as clear analysis strategy are still lacking. The my5C web browser was proposed by **?** to visualize, transform and analyze 5C data. However, the my5C webtool is targeted to end-users and biologists to prepare their 5C experiments and to handle their data but is not dedicated to the development of new statistical algorithms.

The *HiTC* R package has been developed to offer a bioinformatic environment to explore high-troughput 'C' data. One advantage of this package is that it operates within the open source Bioconductor framework, and thus, offers new opportunities for futur development in this field. The current version of the package provides the basic visualization, transformation and normalization functions described in **?**, but also some new functionnalities such as data import, new visualization functions, annotation and other data transformation. Our goal is also to provide a flexible basis for further development, aiming at the integration of new analysis algorithm that are being developped (**?**, **?**, **?**)

This document briefly describes how to use the *HiTC* R package. The package is built on the functionality of Bioconductor packages such as *IRanges* and *GenomicRanges*, and provides new classes and methods to handle with high-throughput 'C' data. It is especially suited to 5C and Hi-C data handling, but can also in principle be used for 4C, though specific needs of 4C users may be best met by *r3Cseq* R package.
Even if the 5C and Hi-C approaches are derived from the same 3C technique, strong differences in their protocol can also be noticed. While 5C enables analysis of interactions between many loci, it also required an extensive number of primers, which is not suitable for a genome-wide analysis as the Hi-C. Thus, the pre-processing of these two types of data is totally different with, for instance, two different mapping strategies.

If you use *HiTC* for analyzing your data, please cite:

- Servant N., Lajoie B.R., Nora E.P., Giorgetti L., Chen C., Heard E., Dekker J., Barillot E. (2012) HiTC : Exploration of High-Throughput 'C' experiments. *Bioinformatics*.

2 Getting started

The current version of the *HiTC* package was developped to work on processed 5C, Hi-C or other high-throughput 3C data.
The *HTCexp* (High-Throughput 'C' experiment) class aims at representing a single 'C' experiment, characteriez by :

- An interaction map (i.e a *Matrix*)
- Two *GRanges* objects that describe each features of the interaction matrix, respectively, the x (i.e. columns) and y (i.e. rows) labels of the interaction matrix. Basically, in the context of 5C, these objects will be the forward and reverse primers, and for the Hi-C the binned genomic intervals.

Note that *HiTC* was not designed to processed chromatine conformation capture from raw reads, but takes contact maps as input. In order to process Hi-C data from raw sequencing reads, you can use the HiC-Pro pipeline **?** which is freely available at https://github.com/nservant/HiC-Pro. Data processed with HiC-Pro can then be loaded into the R environment using the importC function of *HiTC*.

Whereas a 5C dataset can be composed of a single cis interaction map (i.e. *HTCexp* object), a complete Hi-C dataset is composed of a list of cis and trans interaction maps, characterized by the physical interactions of each pair of chromosomes. The *HTClist* class represents a list of *HTCexp* objects and provides dedicated methods and visualization functions.

```
> library(HiTC)
> showClass("HTCexp")
```

```
Class "HTCexp" [package "HiTC"]

Slots:

Name:  intdata     xgi     ygi
Class:  Matrix GRanges GRanges
```

```
> showClass("HTClist")
```

```
Class "HTClist" [package "HiTC"]

Slots:

Name:  .Data
Class:  list

Extends:
Class "list", from data part
Class "vector", by class "list", distance 2
Class "AssayData", by class "list", distance 2
Class "vectorORfactor", by class "list", distance 3
```

3 Working with 'C' Data

HTCexp or *HTClist* objects can be easily created using the dedicated constructors. Additional functions to import data from files are also available.

3.1 A simple example

```
> require(Matrix)
> ## Two genome intervals objects with primers informations
> reverse <- GRanges(seqnames=c("chr1","chr1"),
+                    ranges = IRanges(start=c(98831149, 98837507),
+                    end=c(98834145, 98840771),
+                    names=c("REV_2","REV_4")))
> forward <- GRanges(seqnames=c("chr1","chr1"),
+                    ranges = IRanges(start=c(98834146, 98840772),
+                    end=c(98837506, 98841227),
+                    names=c("FOR_3","FOR_5")))
> ## A matrix of interaction counts
> interac <- Matrix(c(8463, 7144, 2494, 8310), ncol=2)
> colnames(interac) <- c("REV_2","REV_4")
> rownames(interac) <- c("FOR_3","FOR_5")
> z <- HTCexp(interac, xgi=reverse, ygi=forward)
> detail(z)
```

```
HTC object
Focus on genomic region [chr1:98831149-98841227]
CIS Interaction Map
Matrix of Interaction data: [2-2]
2  genomic ranges from 'xgi' object
2  genomic ranges from 'ygi' object
Total Reads =  26411
Number of Interactions =  4
Median Frequency =  7727
Sparsity =  1
```

```
> ## Access to the slots
> x_intervals(z)
```

```
GRanges object with 2 ranges and 0 metadata columns:
        seqnames              ranges strand
           <Rle>           <IRanges>  <Rle>
  REV_2     chr1 [98831149, 98834145]      *
  REV_4     chr1 [98837507, 98840771]      *
  -------
  seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

```
> y_intervals(z)
```

```
GRanges object with 2 ranges and 0 metadata columns:
        seqnames              ranges strand
           <Rle>           <IRanges>  <Rle>
  FOR_3     chr1 [98834146, 98837506]      *
  FOR_5     chr1 [98840772, 98841227]      *
```

```
  -------
  seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

```
> intdata(z)
```

```
2 x 2 Matrix of class "dgeMatrix"
      REV_2 REV_4
FOR_3  8463  2494
FOR_5  7144  8310
```

```
> ## Methods
> range(z)
```

```
GRanges object with 1 range and 0 metadata columns:
      seqnames               ranges strand
         <Rle>            <IRanges>  <Rle>
  [1]     chr1 [98831149, 98841227]      *
  -------
  seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

```
> isBinned(z)
```

```
[1] FALSE
```

```
> isIntraChrom(z)
```

```
[1] TRUE
```

```
> seqlevels(z)
```

```
[1] "chr1"
```

3.2 Import/Export Data

The HiTC package provides the `importC` and `exportC` functions to import/export data from a list file. Only non null values have the written in this file allowing a efficient storage of Hi-C (sparse) data. The format is defined as follow :

- A list file (tab-separated) with per line, the name of both interactors and the number of associated sequencing reads (i.e. `I1 I2 Count1-2`).
- The associated BED files describing the x and y intervals of the HTCexp object. For 5C experiment, it can be the forward and reverse primers location, whereas for Hi-C experiment, it can be a description of the genomic bins. The name of these intervals must match with the name of the interactors in the list file.

In addition, the package is fully compatible with the my5C web browser. The interaction counts matrices can be imported/exported from a matrix file format. The matrix format summarizes all the informations with genomic coordinates as row and column names (ex: `HIC_bin1|hg18|chr14:1-999999`). The row and column names are splitted to create the HTCexp object.

The *HiTC* package includes a sample of the Human Hi-C dataset (GSE18199) published by **?**. The interaction map of chromosome 12 to 14 is used to illustrate the capabilities of the *HiTC* package to explore Hi-C data.

```
> ## Load Lieberman et al. Chromosome 12 to 14 data (from GEO GSE18199)
> exDir <- system.file("extdata", package="HiTC")
> l <- sapply(list.files(exDir, pattern=paste("HIC_gm06690_"), full.names=TRUE),
+             import.my5C)
```

```
> hiC <- HTClist(l)
> show(hiC)

HTClist object of length 6
3 intra / 3 inter-chromosomal maps

> names(hiC)

[1] "chr12chr12" "chr12chr13" "chr12chr14" "chr13chr13" "chr13chr14" "chr14chr14"

> ## Methods
> ranges(hiC)

GRangesList object of length 6:
$chr12chr12
GRanges object with 1 range and 0 metadata columns:
      seqnames         ranges strand
         <Rle>      <IRanges>  <Rle>
  [1]    chr12 [1, 132349533]      *

$chr12chr13
GRanges object with 2 ranges and 0 metadata columns:
      seqnames         ranges strand
  [1]    chr12 [1, 132349533]      *
  [2]    chr13 [1, 114142979]      *

$chr12chr14
GRanges object with 2 ranges and 0 metadata columns:
      seqnames         ranges strand
  [1]    chr12 [1, 132349533]      *
  [2]    chr14 [1, 106368584]      *

...
<3 more elements>
-------
seqinfo: 3 sequences from an unspecified genome; no seqlengths

> range(hiC)

GRanges object with 3 ranges and 0 metadata columns:
      seqnames         ranges strand
         <Rle>      <IRanges>  <Rle>
  [1]    chr12 [1, 132349533]      *
  [2]    chr13 [1, 114142979]      *
  [3]    chr14 [1, 106368584]      *
  -------
  seqinfo: 3 sequences from an unspecified genome; no seqlengths

> isBinned(hiC)

chr12chr12 chr12chr13 chr12chr14 chr13chr13 chr13chr14 chr14chr14
      TRUE       TRUE       TRUE       TRUE       TRUE       TRUE

> isIntraChrom(hiC)

chr12chr12 chr12chr13 chr12chr14 chr13chr13 chr13chr14 chr14chr14
      TRUE      FALSE      FALSE       TRUE      FALSE       TRUE
```

```
> isComplete(hiC)
```

```
[1] TRUE
```

```
> seqlevels(hiC)
```

```
[1] "chr12" "chr13" "chr14"
```

```
> summary(hiC)
```

```
            seq1  seq2 nbreads nbinteraction averagefreq medfreq sparsity
chr12chr12 chr12 chr12 1179512         17418      67.718      20   0.0153
chr12chr13 chr12 chr13   58509         12495      4.6826       4   0.1831
chr12chr14 chr12 chr14   57593         11535      4.9929       5   0.1894
chr13chr13 chr13 chr13  779070          9518     81.8523      28   0.2803
chr13chr14 chr13 chr14   41230          8455      4.8764       5   0.3129
chr14chr14 chr14 chr14  755139          7913     95.4302      30   0.3088
```

4 Quality Control

The first step after data pre-procesing is a quality control to check weither the data are likely to reflect cis and/or trans chromosomal interactions rather than just random collisions. Quality control for the percentage of reads aligned to interchromosomal and intrachromosomal interactions is available, as well as distribution of the interaction frequency against the genomic distance between two loci, and simple statistics (see Figure **??**).

```
> par(mfrow=c(2,2))
> CQC(hiC, winsize = 1e+06, dev.new=FALSE, hist.dist=FALSE)
```

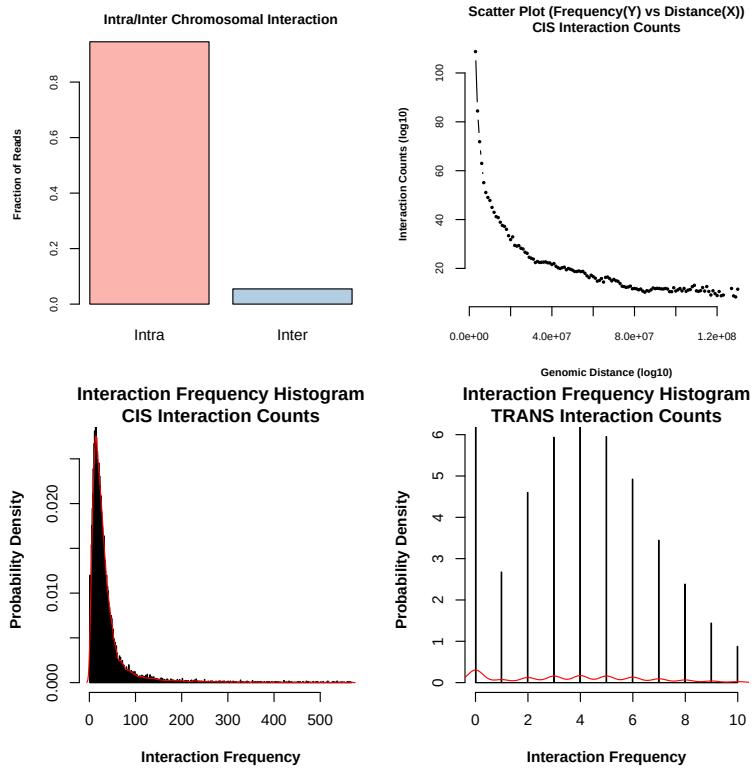

Figure 1: **Quality Control of `hiC` data.** From top-left to bottom-right : proportion of intra/inter chromosomal interactions, scatter-plot of interaction counts versus genomic distance between two loci, histogram of interaction counts for intra (CIS) and inter (TRANS) interactions, histogram of distances between two intrachromosomal loci.

5 HTCexp : single 'C' map experiment

5.1 Attached 5C data

The *HiTC* package includes a 5C dataset (GSE35721) published by **?**, from which we choose two different Mouse samples, male undifferentiated ES cells (E14, GSM873935) and male embryonic fibroblasts (MEF, GSM873924). This dataset is mainly used to describe the available functionalities of the package.

```
> ## Load Nora et al 5C dataset
> data(Nora_5C)
> show(E14)

HTClist object of length 1
1 intra / 0 inter-chromosomal maps

> show(MEF)

HTClist object of length 1
1 intra / 0 inter-chromosomal maps
```

5.2 Visualization of Interaction Maps

The interaction map represents the frequency at which each pair of restriction fragments have been ligated together during the 3C procedure. The goal is to visualize at once these counts for many pairs of restriction fragments across a large genomic region. Each entry in the matrix corresponds to a count information, i.e.,

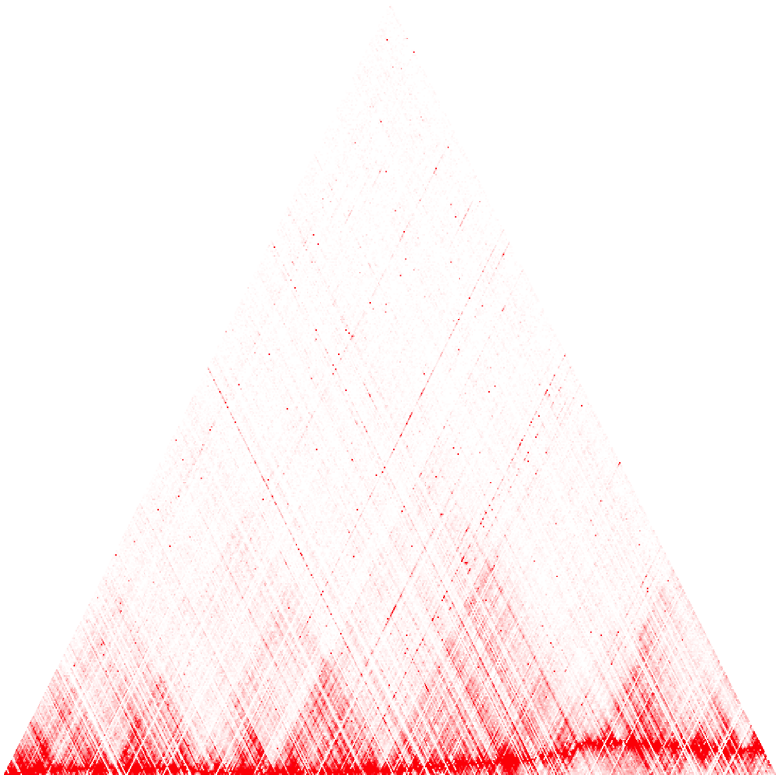

Figure 2: **Visualization of HTCexp object (1).** Raw 5C interaction map of chromosome X.

number of times two restriction fragments have been sequenced as a pair.
In the *HiTC* package, the *HTCexp* object are represented as a triangle view (see Figure **??**). This view is particulary useful for interaction maps comparison and alignment with genomic or epigenomic features on a small region. The mapC function proposes a list of options to play with data visualization, such as contrast, color, or trimming.

```
> mapC(E14$chrXchrX)
```

5.3 Data Transformation

5.3.1 Windowing

Each pixel of an interaction map can correspond either to a single restriction fragment, several restriction fragments or genomic intervals of any given size (and therefore various restriction fragment numbers). 5C allows assessing interaction frequencies for each pair of restriction fragments. The Hi-C protocol, on contrary, does not necessarily yields counts for every single pair of restriction fragments, especially when working with large genomes. Results are thus typically displayed for genomic bins of an arbitrary size.
To produce an interaction map, the genomic range of the display should be divided into appropriately size loci. This size depends on the resolution desired for the analysis. For instance, 5C data can be visualized at the primers resolution, or segmented into 100Kb or 1Mb bins that can be partially overlap or not. Such binned interaction map is symmetrical around the diagonal. For the following example, we decided to focus on a subset of the original dataset (see Figure **??**).

```
> ## Focus on a subset chrX:100295000:102250000
> E14subset<-extractRegion(E14$chrXchrX, c(1,2),
+                          chr="chrX", from=100295000, to=102250000)
> ## Binning of 5C interaction map
```

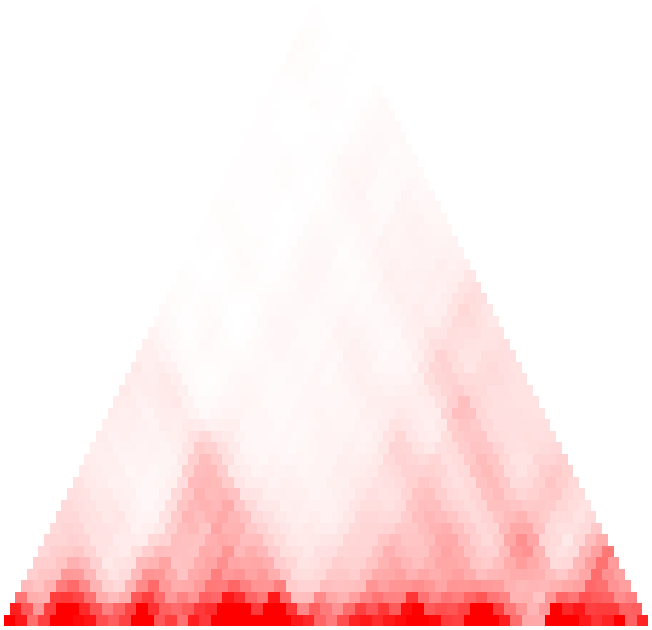

Figure 3: **Visualization of HTCexp object (2).** Binned 5C interaction map of chrX:100295000-102250000.

```
> E14subset.binned <- binningC(E14subset, binsize=100000, method="median", step=3)
> mapC(E14subset.binned)
```

5.3.2 Data Normalization

Due to the polymer nature of chromatin, at small genomic distances, pairs of restriction fragments that are close to each other in the linear genome will give higher signal than fragments that are further apart. Such property leads to strongest counts falling on the heatmap diagonal. When considering any given pair of restriction fragments, it is therefore informative to assess whether the observed counts are above what is expected given the genomic distance that separate them.
Different ways of normalization have been proposed. Here, we propose to estimate the expected interaction counts as presented in **?**. The expected value is the interaction frequency between two loci that one would expect based on a sole dependency on the genomic proximity of these fragments in the linear genome. This can be estimated using a Loess regression model (see Figure **??**). Note that another model based on mean counts at each genomic distance can also be used (method=mean)

```
> ## Look at exptected counts
> E14exp <- getExpectedCounts(E14subset, method="loess", stdev=TRUE, plot=TRUE)
```

Interaction frequencies can be then normalized for distance by dividing the observed value by the expected value (`normPerExpected`). The variability between the interaction counts and the genomic distance between pairs of loci can be calculated if specified. These normalization methods can be easily applied using the methods `normPerReads` and `normPerExpected`.

```
> E14norm <- normPerExpected(E14subset, method="loess", stdev=TRUE)
> E14norm.binned <- binningC(E14norm, binsize=50000, method="median", step=3)
> mapC(E14norm.binned)
```

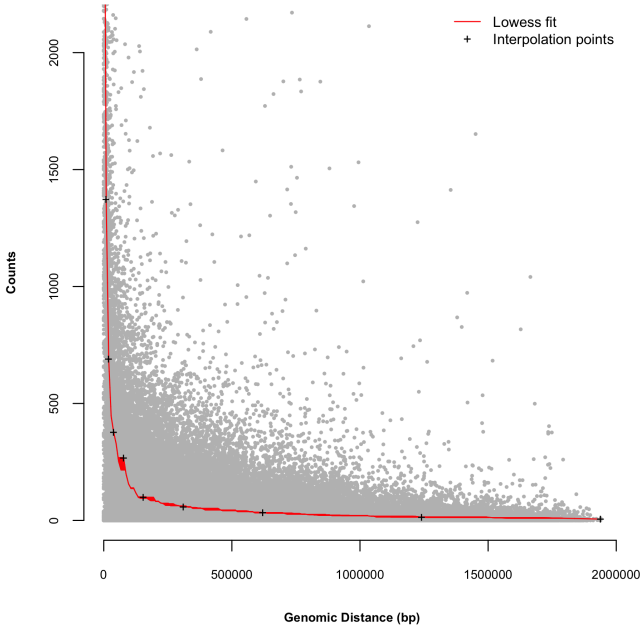

Figure 4: **Estimation of expected count using a Loess smoothing.** The crosses represent the interpolation points.

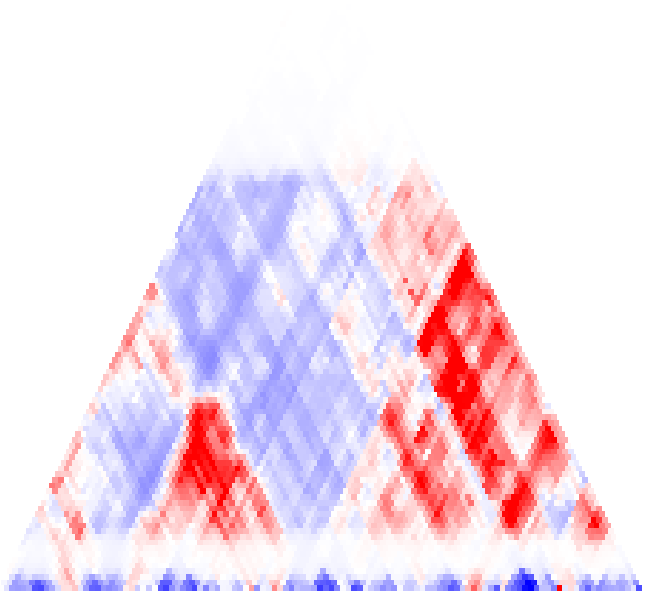

Figure 5: **Normalized 5C data.** Interaction map of data normalized from the background level of interactions.

5.3.3 Annotation of Interaction Maps

The *HiTC* package contains functions for visualizing genomic regions with interaction maps (see Figure **??**). The annotation objects have to belong to the *GRanges* class, cand can be loaded from BED files using the *rtracklayer* package. For instance, the following example displays the CTCF enriched regions (**?**) and RefSeq genes over the interaction map of the E14 sample.

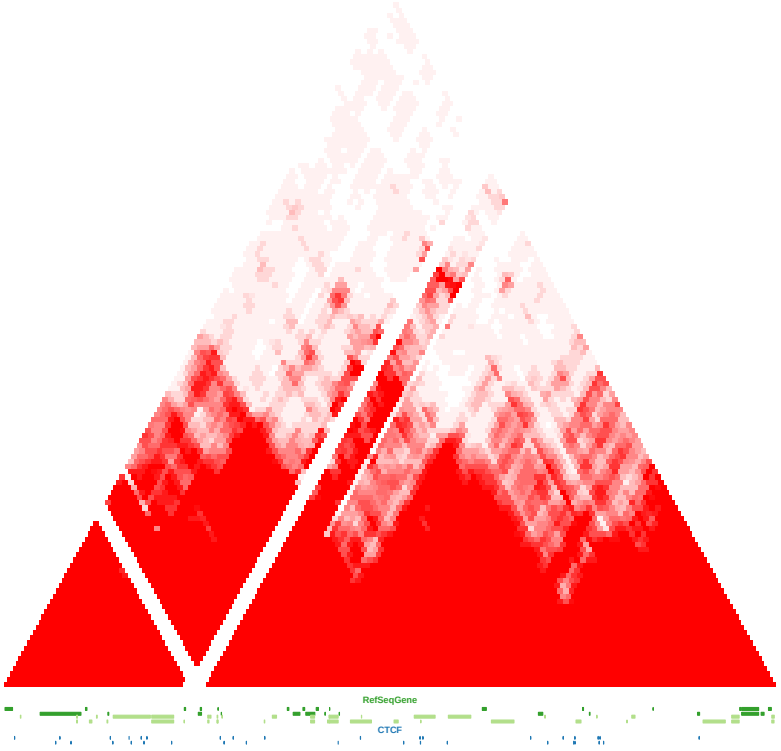

Figure 6: **Visualization of interaction map and genomic annotations.** CTCF enriched regions and RefSeq genes over the interaction map of the E14 sample.

```
> E14.binned <- binningC(E14$chrXchrX, binsize=100000, method="median", step=3)
> require(rtracklayer)
> gene <- import(file.path(exDir,"refseq_mm9_chrX_98831149_103425150.bed"),
+                format="bed")
> ctcf <- import(file.path(exDir,"CTCF_chrX_98892125_102969775.bed"),
+                format="bed")
> mapC(E14.binned,
+      tracks=list(RefSeqGene=gene, CTCF=ctcf),
+      maxrange=10)
```

5.4 Comparison of HTCexp objects

The *HiTC* package provides methods to perform simple operations on *HTCexp*, such as dividing, substracting two objects or extracting a genomic region.
It also proposes a graphical view to compare two 'C' experiments. In the following example, the MEF sample is compared to the E14 sample (see Figure **??**).

```
> MEF.binned <- binningC(MEF$chrXchrX, binsize=100000, method="median", step=3)
> mapC(E14.binned, MEF.binned,
+      tracks=list(RefSeqGene=gene, CTCF=ctcf),
+      maxrange=10)
```

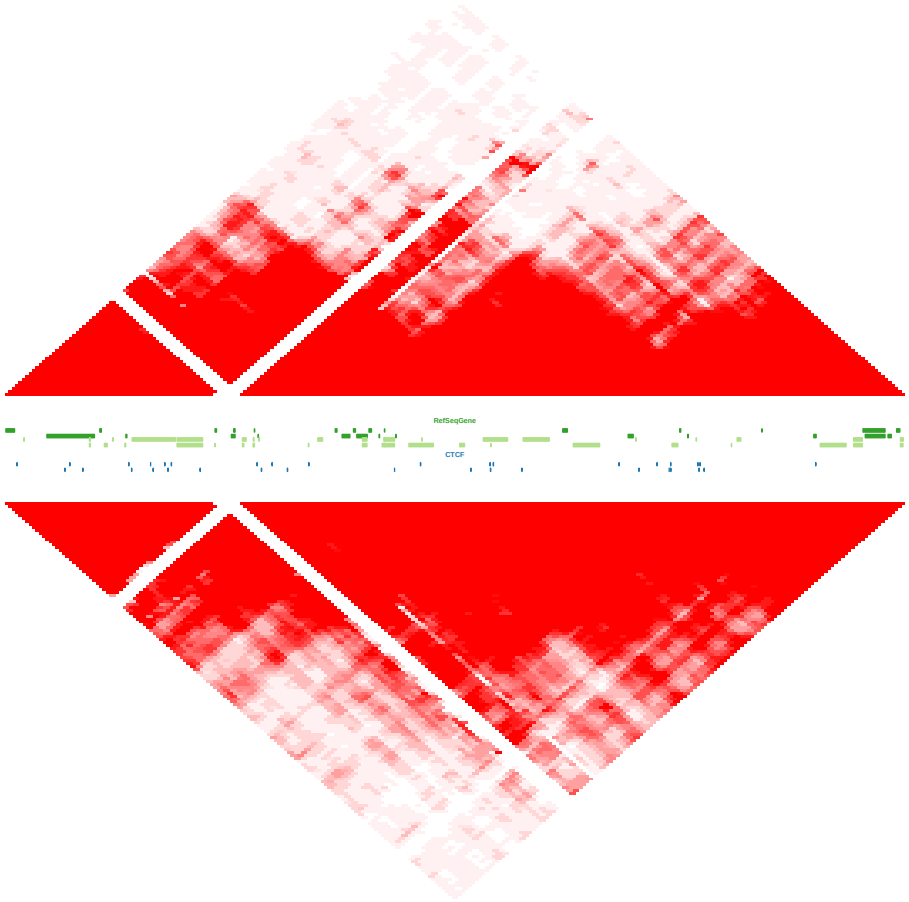

Figure 7: **Comparison of interaction maps.** Comparison of two binned interaction maps, and visualization with genomic annotations.

6 HTClist : Multiple 'C' experiments

Basically, 5C and Hi-C data can be described in the same way. Thus, most of the functions and methods described for the 5C data can be applied to the Hi-C data.

6.1 Visualization of Interaction Maps

The visualization of the *HTClist* is designed such as several interaction maps from the same experiment can be displayed together.
Therefore these data are typically displayed using two dimensional heatmaps of all cis/trans maps.

```
> mapC(hiC, maxrange=100)
```

6.2 Hi-C analysis

In this section, we present how, using a few command lines, we can reproduce some analyses of the **?** paper (see Figures **??**-**??**) on the chromosome 14, from visualization of maps to Principal Component Analysis (PCA).

```
> ## Extract region of interest and plot the interaction map
> hiC14 <- extractRegion(hiC$chr14chr14,
```

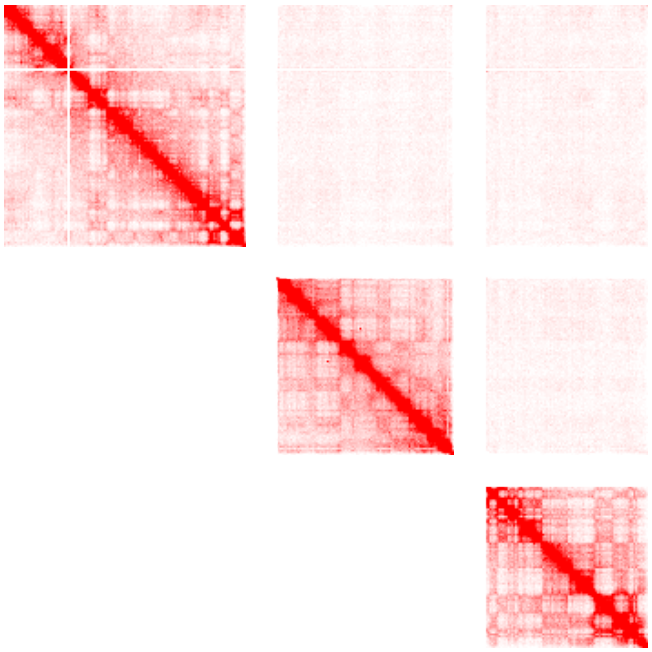

Figure 8: **Visualization of a Hi-C dataset.** Two dimensional heatmaps of the cis/trans maps from the Liberman-Aiden et al. dataset.

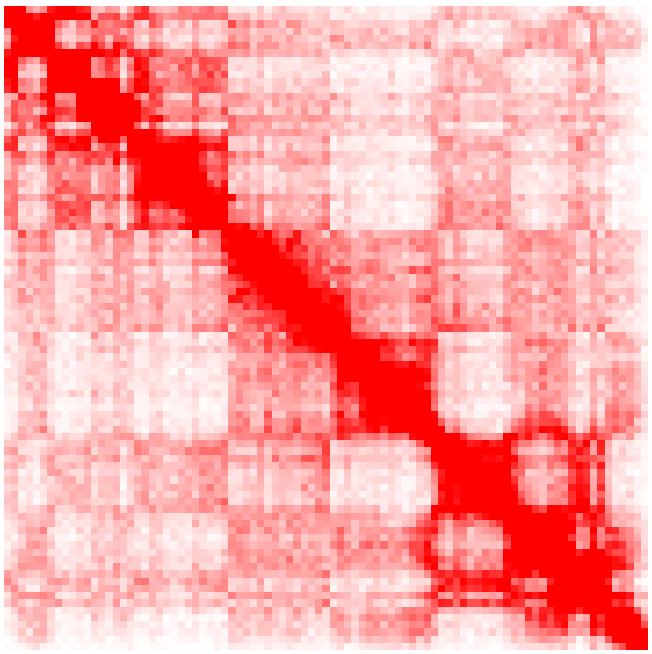

Figure 9: **Hi-C interaction map of chromosome 14.**

```
+                       chr="chr14", from=1.8e+07, to=106368584)
> mapC(HTClist(hiC14), maxrange=100)

> ## Data Normalization by Expected number of Counts
> hiC14norm <- normPerExpected(hiC14, method="loess")
> mapC(HTClist(hiC14norm), log.data=TRUE)

> ## Correlation Map of Chromosome 14
> #intdata(hiC14norm) <- as(cor(as.matrix(intdata(hiC14norm))),"Matrix")
> intdata(hiC14norm) <- HiTC:::sparseCor(intdata(hiC14norm))
```

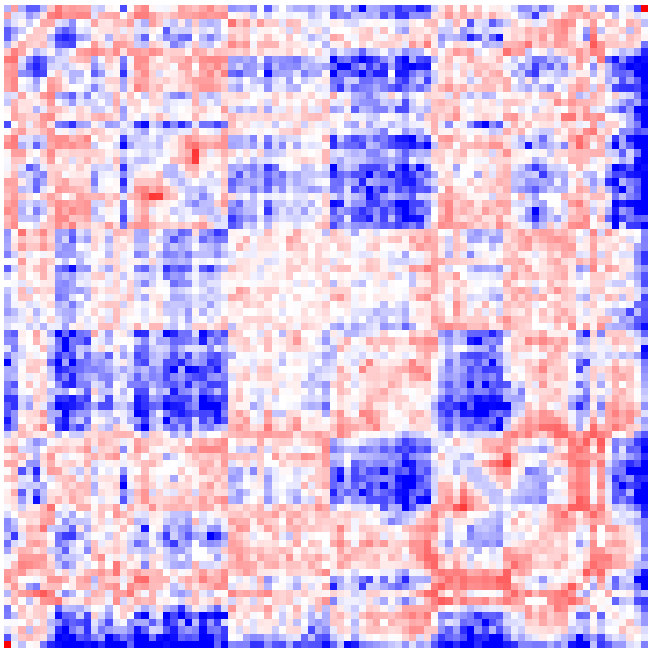

Figure 10: **Interaction map of chromosome 14 normalized by the expected interaction counts.**

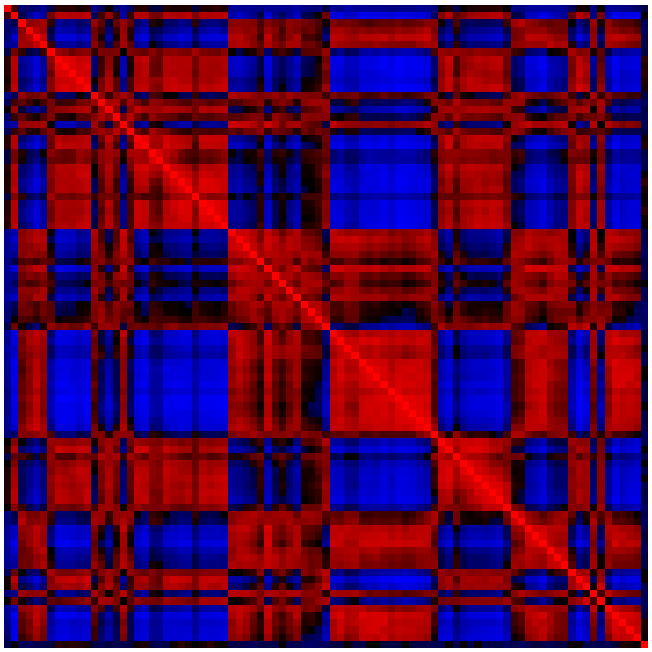

Figure 11: **Correlation map of chromosome 14**

```
> mapC(HTClist(hiC14norm), maxrange=1, minrange=-1,
+      col.pos=c("black", "red"), col.neg=c("blue","black"))
> ## Principal Component Analysis
> pc <- pca.hic(hiC14, normPerExpected=TRUE, method="loess", npc=1)
> plot(start(pc$PC1), score(pc$PC1), type="h",
+      xlab="chr14", ylab="PC1vec", frame=FALSE)
>
```

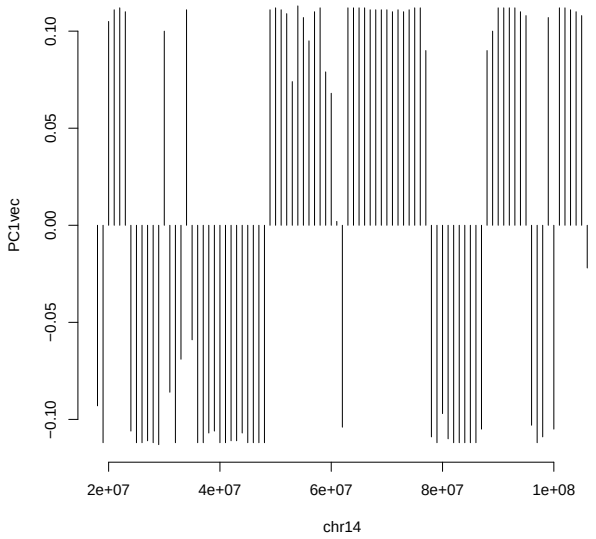

Figure 12: **PCA analysis (Lieberman-Aiden et al.).** Results of the PCA (eigenvector), which reflect the compartmentalization inherent in the heatmap.

7 A word about speed and memory usage

In order to improve the run time on machines with multiple processors, some of the functions in the *HiTC* package have been implemented to make use of the functionality of the *parallel* package. If the options `mc.cores` is initialised before calling these functions, they will make use of `mclapply` instead of the normal `lapply`.
Since the version 1.5.2 of the package, the interaction maps are now stored as *Matrix* object. In case of very high resolution data, such as the 20kb interaction maps published by **?**, a sparse matrix representation is much more efficient in terms of memory usage. The memory requires by the HiTC package for high resolution Hi-C data is represented in the figure **??**. However, in many cases, using *Matrix* objects instead of *matrix* objects is much more slower. Thus, for some functions such as `binningC`, the user can now set the `optimize.by` argument to "speed" or "memory". If set to "speed", the *Matrix* object is convert into a standard *matrix* class, thus taking much more memory during the execution of the function but being much faster. One can notice that for now, all the vizualition functions are based on *matrix* object.

Package versions

This vignette was generated using the following package versions:

- R version 3.3.0 RC (2016-04-26 r70550), `x86_64-apple-darwin13.4.0`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: BSgenome 1.40.0, BSgenome.Hsapiens.UCSC.hg18 1.3.1000, BiocGenerics 0.18.0, Biostrings 2.40.0, GenomeInfoDb 1.8.0, GenomicRanges 1.24.0, HiCDataHumanIMR90 0.105.0, HiTC 1.16.0, IRanges 2.6.0, Matrix 1.2-6, S4Vectors 0.10.0, XVector 0.12.0, rtracklayer 1.32.0
- Loaded via a namespace (and not attached): Biobase 2.32.0, BiocParallel 1.6.0, BiocStyle 2.0.0, GenomicAlignments 1.8.0, RColorBrewer 1.1-2, RCurl 1.95-4.8, Rsamtools 1.24.0, SummarizedExperiment 1.2.0, XML 3.98-1.4, bitops 1.0-6, grid 3.3.0, lattice 0.20-33, tools 3.3.0, zlibbioc 1.18.0

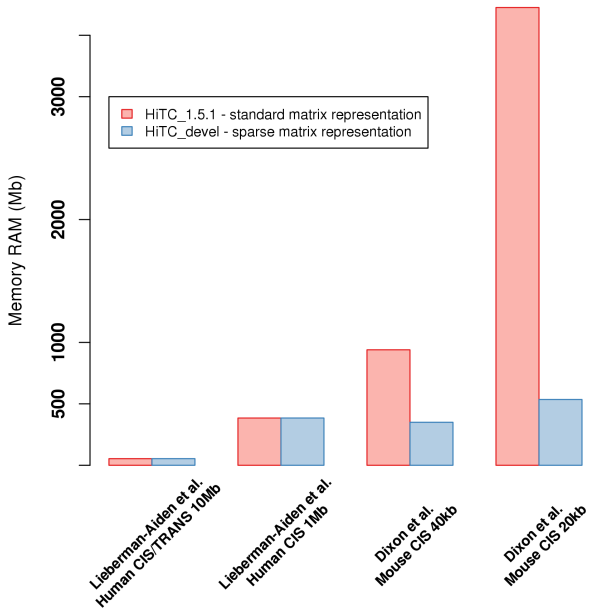

Figure 13: **HiTC memory usage.** Improvement of the memory usage of the HiTC package through the storage of sparse matrix (*Matrix*).

Acknowledgements

Many thanks to Pierre Gestraud for useful discussion and help in developping this R package. A special thanks to the *HiTC* users, and especially to Sameet Mehta for useful discussions and idea to improve it.