TADs are 3D structural units of higher-order chromosome organization in Drosophila

Quentin Szabo, Daniel Jost, Jia-Ming Chang, Diego I. Cattoni, Giorgio L. Papadopoulos, Boyan Bonev, Tom Sexton, Julian Gurgo, Caroline Jacquier, Marcelo Nollmann, Frédéric Bantignies, Giacomo Cavalli.
Sci. Adv. 2018;4:eaar8082

何子安 Zi-Onn｜管漢程 Han-Cheng｜謝皓雲 Hao-Yun

Bioinformatics 112

2024/01/04

Outline

- Overview
- Objectives
- Data Description
- Tools
- Challenges and Solutions
- Result
- Demo
- Cooperate

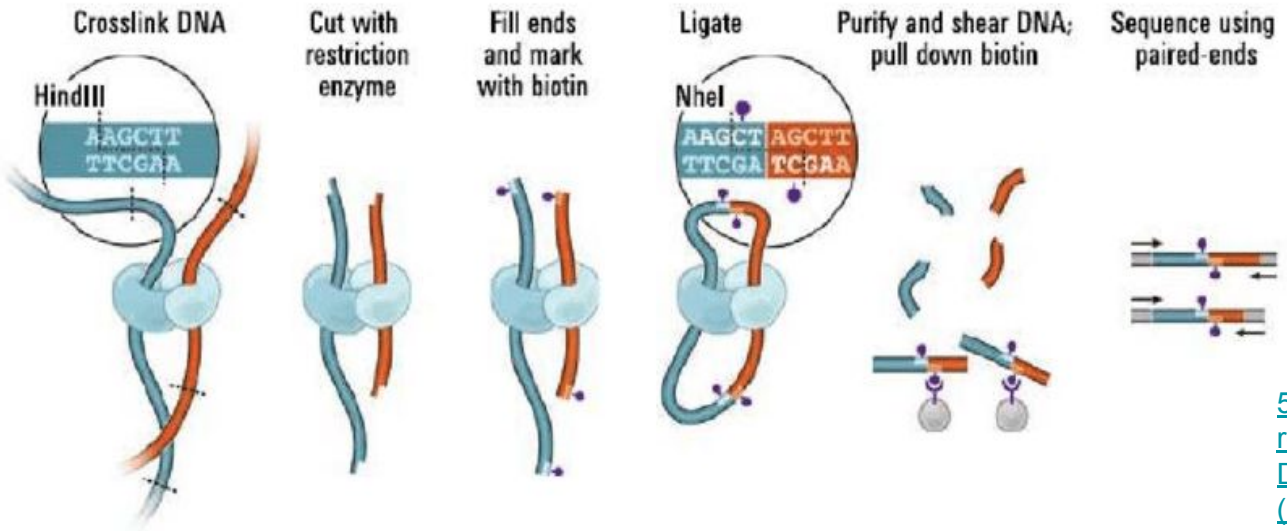
Overview

Overview

- Hi-C is a technique to predict the **3D structure of genome.**
- Hi-C studies have revealed that genome is partitioned into **TADs**
- **However, whether TADs are true physical units in each cell nucleus or whether they reflect statistical frequencies of measured interactions within cell populations is unclear**
- **The result is “TADs are fundamental 3D genome units”, not just a math concept.**

Hi-C

- Goal: Deciphering the rules of genome folding in the cell nucleus
- Hi-C is a novel technique that combines **Chromosome Conformation Capture (3C)** and **next-generation sequencing** (**NGS**)
- Explain the **interaction between enhancer and promoter**



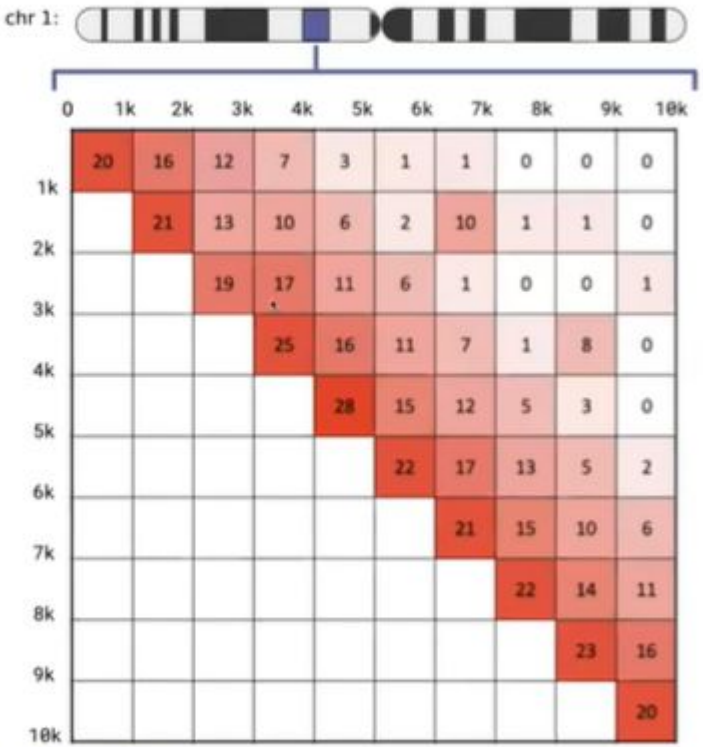
[5 -Principe du Hi-C. Les cellules sont réticulées avec du formaldéhyde... | Download Scientific Diagram (researchgate.net)]

HiC heat map

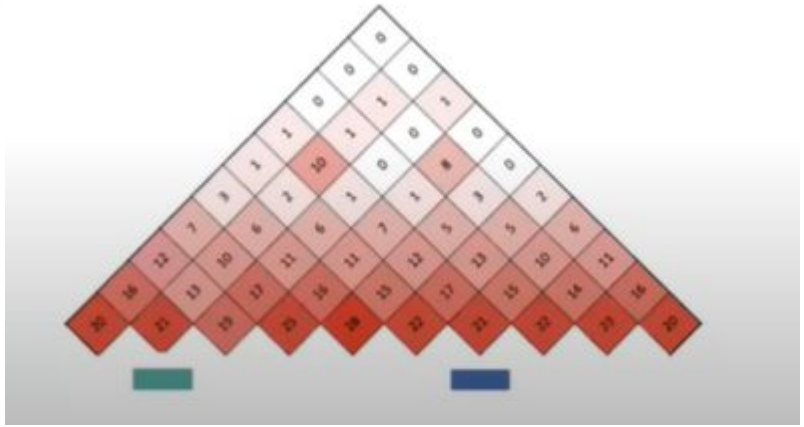
- **Hi-C heat map** is a matrix-like plot, and it must be symmetric
- X,Y axis represent position of genome, kb is the unit of resolution, that is the bin size.

- Color is the number of reads that supports a 3D linkage
- We can change resolution to determine which character to see. Compartment: 100kb, TADs: 50kb, loop: 5kb

Visualization: Hi-C Heat Map



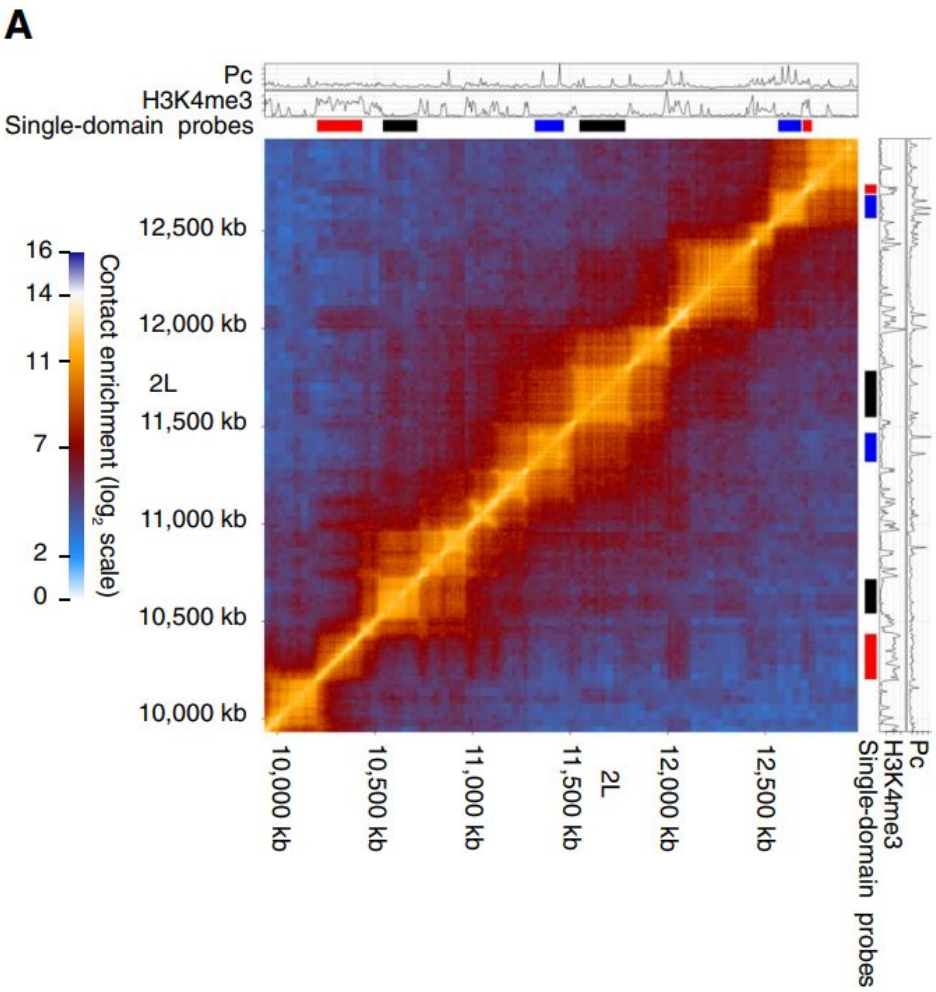
More color = More reads = More likelihood of contacts



ARIMA GENOMICS

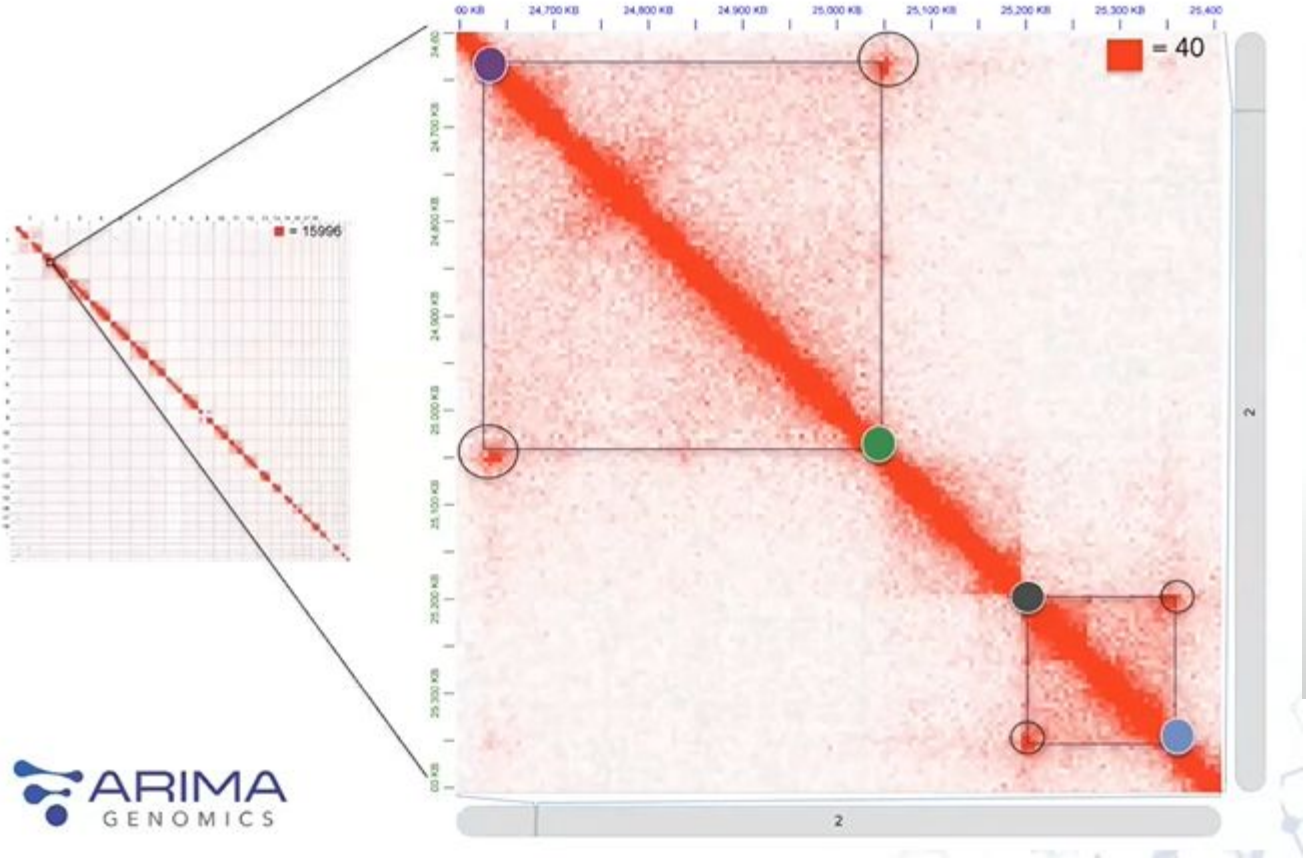
Confidential

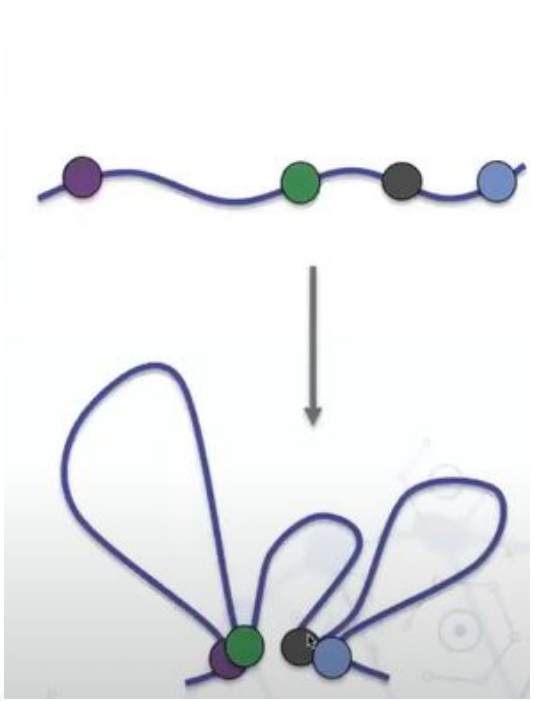
Credit by ARIMA Genomics


A
Pc
H3K4me3
Single-domain probes
12,500 kb
12,000 kb
2L
11,500 kb
11,000 kb
10,500 kb
10,000 kb
16
14
11
7
2
0
Contact enrichment (log2 scale)
10,000 kb
10,500 kb
11,000 kb
11,500 kb
2L
12,000 kb
12,500 kb
Pc
H3K4me3
Single-domain probes

What is TADs

- Domains of **highly interacting** chromatin.
- demarcate functional epigenetic domains defined by combinations of specific chromatin marks





Credit by ARIMA Genomics

Objectives

Objectives

- Reproduce HiC map by using original fastq files, to spot TADs from it.

Data Description

Data Description: `.fastq` files.

- From 6 samples:
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837380
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837381
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837378
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837379
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837376
  - https://www.ebi.ac.uk/ena/browser/view/SRX2837377

```
TADs_data_zipped
├── SRX2837376-fastq_ftp-20231214-0515
│   ├── SRR5579160.fastq.gz
│   ├── SRR5579160_1.fastq.gz
│   ├── SRR5579160_2.fastq.gz
│   ├── SRR5579161.fastq.gz
│   ├── SRR5579162.fastq.gz
│   ├── SRR5579162_1.fastq.gz
│   ├── SRR5579162_2.fastq.gz
│   ├── SRR5579163.fastq.gz
│   ├── SRR5579163_1.fastq.gz
│   ├── SRR5579163_2.fastq.gz
│   ├── SRR5579164.fastq.gz
│   ├── SRR5579164_1.fastq.gz
│   ├── SRR5579164_2.fastq.gz
│   ├── SRR5579165.fastq.gz
│   ├── SRR5579165_1.fastq.gz
│   ├── SRR5579165_2.fastq.gz
│   ├── SRR5579166.fastq.gz
│   ├── SRR5579166_1.fastq.gz
│   └── SRR5579166_2.fastq.gz
├── SRX2837377-fastq_ftp-20231214-0516
│   ├── SRR5579167_1.fastq.gz
│   ├── SRR5579167_2.fastq.gz
│   ├── SRR5579168_1.fastq.gz
│   ├── SRR5579168_2.fastq.gz
│   ├── SRR5579169_1.fastq.gz
│   └── SRR5579169_2.fastq.gz
├── SRX2837378-fastq_ftp-20231214-0514
│   ├── SRR5579170_1.fastq.gz
│   ├── SRR5579170_2.fastq.gz
│   ├── SRR5579171_1.fastq.gz
│   ├── SRR5579171_2.fastq.gz
│   ├── SRR5579172_1.fastq.gz
│   ├── SRR5579172_2.fastq.gz
│   ├── SRR5579173_1.fastq.gz
│   └── SRR5579173_2.fastq.gz
├── SRX2837379-fastq_ftp-20231214-0514
│   ├── SRR5579174_1.fastq.gz
│   ├── SRR5579174_2.fastq.gz
│   ├── SRR5579175_1.fastq.gz
│   ├── SRR5579175_2.fastq.gz
│   ├── SRR5579176_1.fastq.gz
│   └── SRR5579176_2.fastq.gz
├── SRX2837380-fastq_ftp-20231214-0425
│   ├── SRR5579177_1.fastq.gz
│   └── SRR5579177_2.fastq.gz
└── SRX2837381-fastq_ftp-20231214-0513
    ├── SRR5579178_1.fastq.gz
    └── SRR5579178_2.fastq.gz
```

Data Description: `.hic` files.

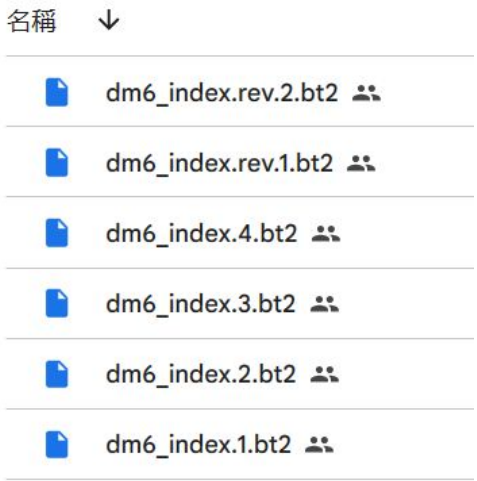
- Hic format is an indexed binary format designed to permit fast random access to contact matrix heatmaps.
- The format is used for displaying chromatin conformation data in the browser.
- useful for displaying interactions at a scale and depth that exceeds what can be easily visualized with the interact and bigInteract formats.
- After running a chromatin conformation experiment such as in situ Hi-C, we can pass our results through the various pipelines to produce a hic file.

Data Description: Reference Genome & Index Genome

The paper used “dm3” (Apr. 2006 (BDGP R5/dm3)) as reference genome, but in our own experiments, we used “dm6” (Aug. 2014 (BDGP Release 6 + ISO1 MT/dm6)), which is the newer version of fruit fly genome published in 2014, as our reference genome.

After obtaining the reference genome, we then use “bowtie 2” package to generate Index Genome from dm6 data.

```
$ bowtie2-build dm6.fa.gz dm6_index
```



Tools

Tools: FAN-C

FAN-C
0.9.1

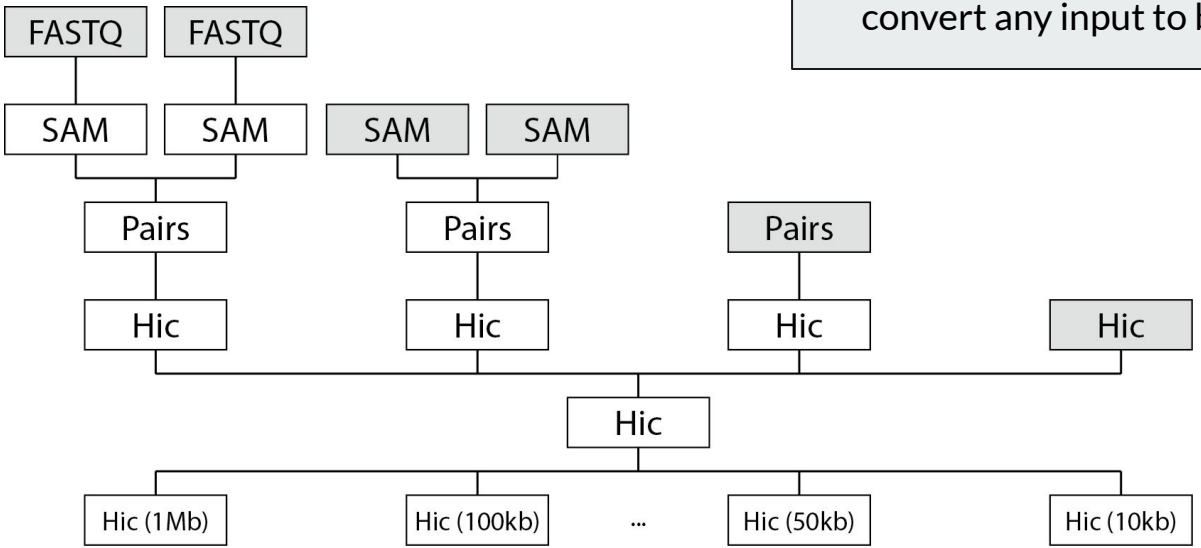
We used FAN-C to convert `.fastq` file to `.hic` file using the given data and the index genome we previously generated from dm6 via bowtie 2.

FAN-C is a Python (3.6+) toolkit for the analysis and visualization of Hi-C data.

The "`fanc auto`" command can convert fastq to hic. And we can use the "`fancplot`" command to draw Hi-C heatmap using the `.hic` data we get from the fanc auto command.

Underneath FAN-C

convert any input to binned Hi-C matrices



Underneath FAN-C

1. map reads in FASTQ files to a reference genome
2. generating SAM/BAM files
3. SAM/BAM files with paired-end reads will be automatically sorted and mate pairs will be matched to generate Pairs files
4. Pairs files will be converted into fragment-level Hic objects
5. Multiple fragment-level Hic objects will be merged into a single Hi-C object
6. Finally, the fragment-level Hic object will be binned at various bin sizes

Challenges & Solution

Challenges & Solution: Large Files Sizes (1)

The very first issue we encountered is the file size issue. As the genome used in the paper are some very large genome. The files we got were too large to be computed locally by our laptops.

Therefore, we uploaded some of the `.fastq` files to Google Drive, and built up the environment of FAN-C using Google Colab.

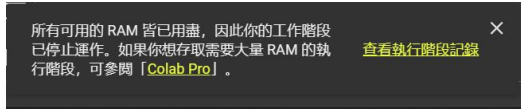
SRR5579163.fastq.gz	474.7 MB
SRR5579163_2.fastq.gz	1.94 GB
SRR5579163_1.fastq.gz	1.94 GB
SRR5579160.fastq.gz	519.2 MB
SRR5579160_2.fastq.gz	2.25 GB
SRR5579160_1.fastq.gz	2.25 GB

Challenges & Solution: Large Files Sizes (2)

However, even so, the .fastq files were still too large to be computed by FAN-C, we would always encounter "`RunTime Error`" after the "`fanc auto`" command got executed for more than 4 hours.

Or in the even worse case, we would run out of the RAM Google provides users on Colab.

So my solution to this issue was to use "`fastqsplitter`" to divide the original fastq file into 5 smaller `.fastq` files. Each fraction has the size around 0.5 GB.




```
!fastqsplitter -i SRR5579163_1.fastq.gz -o SRR5579163_1A.fq.gz -o SRR5579163_1B.fq.gz
-o SRR5579163_1C.fq.gz -o SRR5579163_1D.fq.gz -o SRR5579163_1E.fq.gz
```

Challenges & Solution: Unable to normalise.

After splitting data in smaller fractions, we chose 2 smaller fractions to be the input for "`fanc auto`" each time. However, again, we would still encounter Runtime Error after a few hours. But with fraction, we were able to get a hic output before the Runtime Error interrupts the whole process.

The problem is that such hic output hasn't been normalised yet, and every time we tried to normalise it by `hic.normalise(method="KR")`, the whole environment would crash. So in the end we had to manually adjust the `-vmax` parameter in `fancplot` command to assign better colour output to our hic map.

```
!fancplot chr2L:1mb-1.5mb -o theplot.png -p triangular -vmax 2.5 output/hic/SRR5579163_B_hic.hic
```

😩

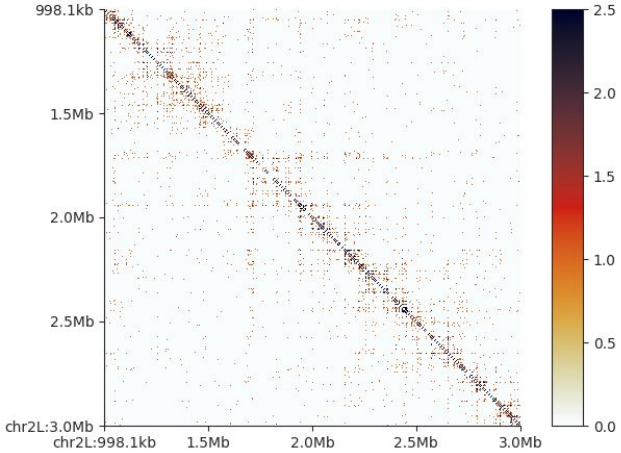
- Different pipeline with different tools

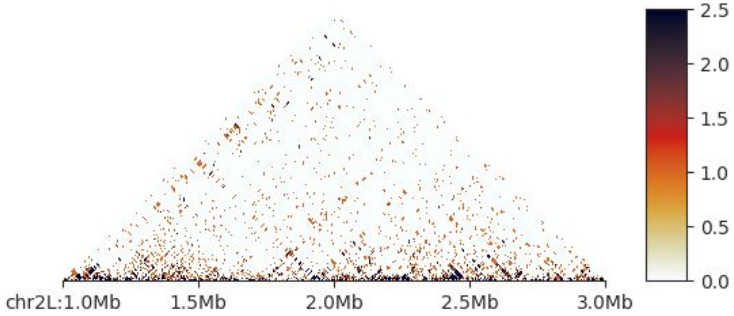
  would have different version/format of `.bed`, `.hic`.
- Some of the results can’t be use in other tools.

Results

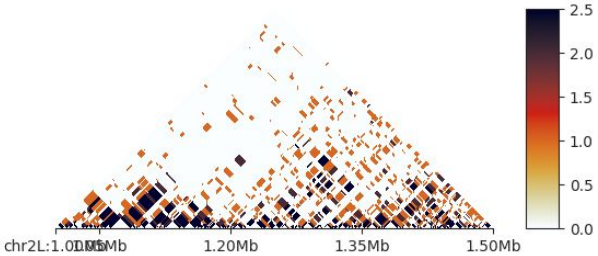
Results (view from large scale)

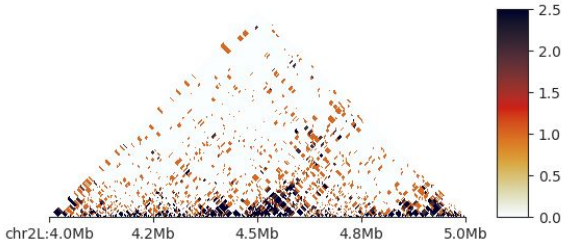
Finally, we managed to create hic heatmap with 2 of SRR5579163's fractions.





Results (view from smaller scale)



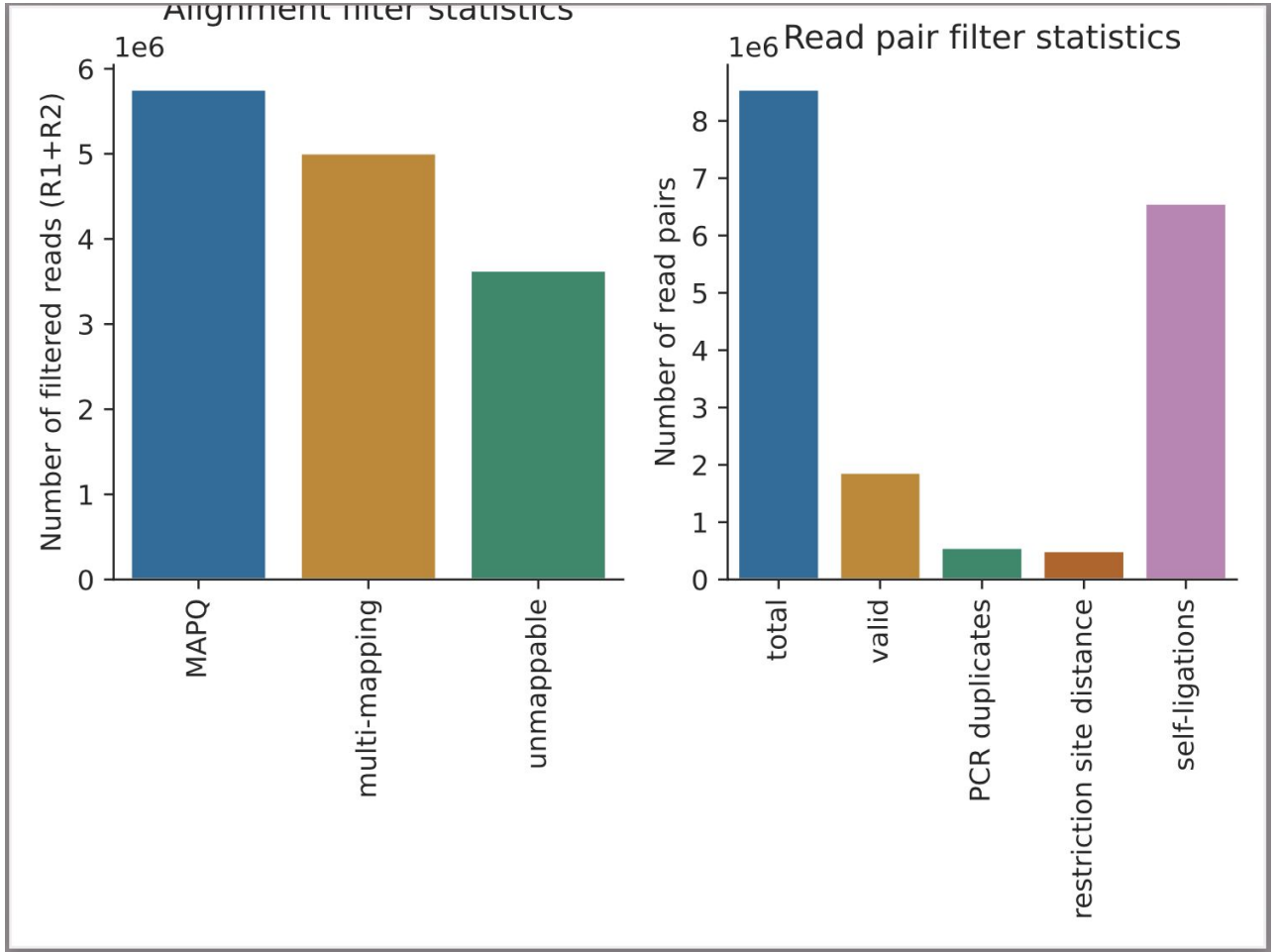


It's easier to spot TADs from our Hi-C map when viewing it from smaller scale. The section of inspecting can be adjusted in the previous `fancplot` command.

```
<fanc.hic.Hic object at 0x7f25c6509ae0>
<class 'fanc.hic.Hic'>
                    Chromosome  Start    End
0                        chr2L      1   7428
1                        chr2L   7429   9360
2                        chr2L   9361  22467
3                        chr2L  22468  23673
4                        chr2L  23674  26566
...                        ...    ...    ...
43343  chrY_CP007108v1_random  43940  48089
43344  chrY_CP007108v1_random  48090  48279
43345  chrY_CP007108v1_random  48280  63339
43346  chrY_CP007108v1_random  63340  63529
43347  chrY_CP007108v1_random  63530  66731

[43348 rows x 3 columns]
```

Results


Alignment filter statistics
1e6
Number of filtered reads (R1+R2)
0
1
2
3
4
5
6
MAPQ
multi-mapping
unmappable
Read pair filter statistics
1e6
Number of read pairs
0
1
2
3
4
5
6
7
8
total
valid
PCR duplicates
restriction site distance
self-ligations

Demo (Github & Tooling)

Github

- Reproducibility
  - How to **document** our project?
  - How to maintain our **code**?
  - How to **reproduce** our result?
  - How to **coordinate** team work?

Tooling

The fun part: Tooling itself

The struggle part: There are just so many tools

Questions?

Cooperate

Zi-Onn: documentation, fanc, data, experiments.

Hao-Yun: problem solving, experiments, hic map production.

Han-Cheng: paper reading, experiments.