Report 4: Raw Hi-C data processing

Cyril Matthey-Doret

22 octobre 2016

Introduction

All previous definitions of TAD-boundaries relied on arbitrary thresholds and their sizes were proportional to the length of their respective TAD. This has not much biological meaning and It would be better to define boundaries separately for each TAD. A straight-forward approach is to use the contact matrices obtained from Hi-C experiments[1]. Here I follow the procedures described by **Rao et al (2014)** to process the matrices.

1. Normalization

Two normalization vectors were provided with the dataset; VC (Vanilla Coverage) and KR (Knight et Ruiz). I used the KR vector as this is the same used in the article where the data comes from. I worked only with the matrices at 5kb (highest) resolution and MAPQ ≥ 30. Matrices are in sparse matrix format. where the row/column indexes correspond to the left edge of the bin.

The normalization procedure consists in dividing each entry in the contact matrix by a corresponding value in the vector:

$M^*_{i,j} = \frac{M_{i,j}}{V_{KR}[\frac{i}{res}] * V_{KR}[\frac{j}{res}]}$

Where $M_{i,j}$ is an entry from the raw matrix and $M^*_{i,j}$ corresponding normalized entry.

The normalization algorithm did not converge for all entries, so there are missing values in the vector. The values that could not be normalized are considered as 0 in the matrix (i.e. not listed in sparse format).

2. Summing interactions:

I computed interactions in a similar manner that Rao and al (2014) computed the insulation score. A diamond of width w is slid over the diagonal of matrix M of n dimensions. Interactions are summed inside the diamond at each position d along the diagonal. I used a 100kb diamond in the algorithm, as recommanded by Rao and al.

$\begin{cases} 1 \leq w \leq \frac{n}{2} + 1 \\ \forall d \in \{w, ..., n - (w-1)\} \end{cases} V_d = \sum_{i=d-(w-1)}^{d} \sum_{j=d}^{d+(w-1)} M_{i,j}$

2. TAD boundaries

Since I already have a set of TADs, there is no need to compute the insulation score to find the TAD borders at local minima. The procedure I use to define TAD boundaries is:

1. Slide a diamond of 100kb*100kb over the matrix diagonal with a stepsize of 5kb and store the total number of interactions inside the diamond.

[1] Matrices used are from the GSE63525 GM12878 primary intrachromosomal contact matrices dataset

   - Note: the diamond can only slide from rows nr [diamond size] to i-[diamond size] otherwise it will get out of the matrix. The positions where the diamond did not slide are = 0

2. Compute the maximum interactions value for each TAD.
3. slide from the TAD borders until the number of interaction (recorded by the diamond) reaches a portion of this value (e.g. 10%)
4. define the areas between these points as boundaries.