3D transmap a package to analyse associations between transcriptomics and 3D genomics

Hocine Meraouna
M2BI
Université Paris Diderot
Projet Long

Abstract

The field of 3D genomics grew at increasing rates in the last decade, and the volume and complexity of the produced 3D data urged the development of new technologies for distributing genome sequences that provides new opportunities for the development of innovative approaches. The chromosomes folding their localization in the three-dimensional nucleus has also been shown to have a significant impact on chromatin activity. It can thus be interesting to have a tool that takes advantage of the emergency of this 3D genomic data and gives a visual indication of the relationship between them and the genome expression.

I - Introduction

The genome is organized within well-defined regions called chromosome territories [1, 2], and studies have shown that the three-dimensional genome organization plays an important role in the regulation of some specific biological processes [3, 4] it especially has a tide relationship with the transcription [5], this importance of spatial co-locatization of the functional DNA elements has been recognized early [6], and it's relation with the biological processes was typically studied with microscopy-based methodologies to catch a glimpse of chromatin behavior [7], but it is the development of molecular biology techniques based on the chromosome conformation capture (3C) methods [8] such as 3C-Carbon Copy (5C) and Hi-C that can capture folding of up to the entire genomes, that have lighted the way genomes are organised in the three-dimensional space and expanded the size and scope of possible investigations, in parallel, the advances in next generation sequencing and its application to

transcriptomics enabled impressive scientific achievements like the access to the activity of a whole transcriptome with high accuracy and unprecedented speed [9]. The 3D genome data extracted from the 3C technology is complex and processes multiple levels of structural organization [10], creating the need for new tools that will improve data access, facilitate interpretation and take full advantage of all the information this data can provide, in that way some customized visualization tools and techniques are today available like Genome3D [11], GMOL [12] or 3DGB [13], but these tools don't provide informations for the relationship between the genome 3D architecture and the transcriptional activity.
In this project we implement an interactive python visualization tool to analyse the spatial correlations and associations between 3D genome organisation and gene transcription to allow the genome wide study of the associations between genome 3D architecture, transcriptional activity and genome regulation based on transcription factories concept.

II - Material and methods

II.1 - Material

Python :

The application was written in Python3 and tested under a Linux environment, Python is an interpreted, high-level object-oriented and interactive programming language, it design emphasizes code readability and it combines very clear syntax with remarkable power, it has modules, classes, exceptions and dynamic typing. An incredible amount of useful non standard libraries are also available for Python. All this points making Python a perfect language for the purpose of this project.

Python non-standard libraries :

pandas :
pandas is an open source, BSD-licensed library providing high-performance, easy to use data structures and data analysis tools for Python programming language, some of the most useful elements provided by pandas for this project are its set of labeled array data structures, mainly DataFrames and Series, input/output tools for the loading of tabular data from flat files, moving window statistics and many others.

NumPy :
NumPy is a BSD-licensed library enabling reuse with few restrictions, it is the fundamental package for scientific computing with Python, it contains a powerful multidimensional array object, sophisticated functions, useful linear algebra, Fourier transform, and random number capabilities and many other useful elements.

SciPy :
SciPy library is one of the core packages that make up the SciPy stack. It provides many user-friendly and efficient numerical routines such as routines for numerical integration and optimization. SciPy is also an ecosystem containing a collection of open source softwares for scientific computing in Python mainly NumPy, pandas and matplotlib.

Matplotlib :
Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hard copy formats and interactive environments across platforms. The main usefulness of matplotlib in this project is its easy to use toolkits providing the ability to make 3D plots.

Data files :

The application require two input data files :

1- The genes positions file, it is a file that contains the 3D positions of the genes on a genome, the structure of the file must be the following as we can see on fig.1 :
a first line that represents the columns names, the only required columns are X, Y and Z and must have these specific names (note that the first column with no name represents the row names)
the other lines must start with the gene name.

	chr	X	Y	Z		
PF3D7_0100100	chr01	0.358894	0.120871	0.107271		
PF3D7_0100200	chr01	0.356576	0.083654	0.093874		
PF3D7_0100300	chr01	0.366940	0.097076	0.079452		
PF3D7_0100400	chr01	0.374553	0.117058	0.065022		
PF3D7_0100500	chr01	0.360451	0.115275	0.071775		
PF3D7_0100600	chr01	0.353052	0.114340	0.075319		
PF3D7_0100700	chr01	0.337844	0.112417	0.082601		
PF3D7_0100800	chr01	0.318406	0.111964	0.093615		
PF3D7_0100900	chr01	0.329376	0.129016	0.101694		

fig.1 : example of 10 first lines of a gene position file

2- The genes expression file, it is the file that contains the values of the expression of the genes, in fig.2 we have an example of such a file were the expression was measured at different hours.
the first column of each line from the second must contain the gene name.

	Hour0	Hour8	Hour16	Hour24	Hour32	Hour40	Hour48		
PF3D7_0100100	5.8692	10.0409	8.5659	6.1746	7.5189	11.3458	11.8587		
PF3D7_0100200	5.3815	3.6215	1.6919	4.6601	1.2313	3.9779	3.8809		
PF3D7_0100300	3.1295	5.0767	2.2089	1.1385	1.1082	1.5369	2.3447		
PF3D7_0100400	3.7487	6.5621	6.6931	5.3734	7.0145	8.0469	6.9087		
PF3D7_0100500	2.0807	0	0	6.593	0	0	7.1902		
PF3D7_0100600	0.3236	0.2385	0.26	1.6604	0.2271	0.3335	2.9826		
PF3D7_0100700	0	0	0	0	0	0.8827	0		
PF3D7_0100800	1.0132	0.6223	0.2713	1.4269	1.1849	4.35	2.9566		
PF3D7_0100900	0.8852	0.4893	1.7781	3.0053	0.9317	2.9074	4.4356		

fig.2 : example of 10 first lines of a gene expression file

II.2 - Methods

The first step is to read the genes positions and expression files and load their data into pandas data frames, we then only keep the genes that are common to both files.

Distance matrix :
In order to be able to get the closest genes to each gene, we compute the euclidean distance matrix out of the genes positions data frame, an Euclidean distance matrix is a square n*n matrix representing the spacing of a set of n points in euclidean space, by containing the distances taken pairwise between these points.
The scipy.spatial cdist function is used to get the distance matrix, computes the distance between m points using Euclidean distance as distance metric between the points. The points are arranged as m n-dimensional row vectors in the matrix.

Correlation matrix :
We then proceed the compute the Spearman correlation matrix out of the transpose of the genes expression data frame, a correlation matrix is used to evaluate the dependence between many variables, the result is a table containing the correlation coefficient between each variable, there is many correlation methods, the Spearman's rank correlation coefficient is a nonparametric measure of rank correlation.
For a sample of size *n*, the *n* raw scores Xi , Yi are converted to ranks rg Xi , rgYi, and rs is computed from:

$$r_s = \rho_{\mathrm{rg}_X,\mathrm{rg}_Y} = \frac{\mathrm{cov}(\mathrm{rg}_X, \mathrm{rg}_Y)}{\sigma_{\mathrm{rg}_X}\sigma_{\mathrm{rg}_Y}}$$

The pandas corr function is used to obtain the correlation matrix, it computes pairwise correlation of columns, excluding NaN/Null values.

We then proceed to sum the correlation coefficient of the n (n being chosen by the user) closest genes of each gene.

Visualization :
Once the previous steps done, we end up with a new data frame containing the genes x, y z positions and the sum of correlation coefficient of each gene n closest genes. We so plot in 3D this data frame's genes on x, y, z axis, and those are coloured from blue to red following the associated correlation coefficient.
The 3D plot is obtained using matplotlib.pyplot scatter function.

The Python program can be found in this github repository :
https://github.com/hocinebib/3D_transmap_Meraouna
and can be run with the following command line :
$python3 src/main.py data/SCHIZONTS.genes_pos.txt data/profiles_Otto2010_copy.min 10

where data/SCHIZONTS.genes_pos.txt is the genes positions file, data/profiles_Otto2010_copy.min is the genes expression file and 10 is the n number of close genes.

Data storing :
genes expression and genes positions data are mainly stored in pandas data frames which are two dimensional size-mutable, potentially heterogeneous tabular data structures with labeled rows and columns and on which arithmetic operations align on both row and coluln labels. a dictionary was used to store the sum of the correlation coefficients of the closest genes for each gene. And finally, numpy arrays were used to store the x, y, z position values, gene names, and correlation sum for the 3D scatter plotting.

Results

The following results were obtained using *plasmodium falciparum* genes expressions data (of the three intraerythropoetic stages of P. falsiparum) obtained with chromosome conformation capture coupled with next-generation sequencing (Hi-C) [14], and genes position data (3D coordinates x,y,z of plasmodium genes in the schizozoite stage).

Running the program on this files gives the result shown on fig.3, as it may be seen we have very few red to yellow spots as most of the spots are blue, we may also notice that the red spots are usually surrounded by yellow and orange spots meaning that the genes on those areas are more or less co-expressed, we also notice that red to yellow spots aren't found only on specific areas of the 3D architecture but that there is some of them are distributed more or less all over the genome 3D organization, but of course we notice areas that are way more rich of these red to yellow spots and thus can be further investigated to eventually get an idea of the nature of the relationship between the expression of these genes and their position on the 3D architecture, we can also spot from this figure an issue of the visualization result of the program as we may see once we have a lot of genes it is less easy the spot some genes due to the big amount of points that cover them, but we can zoom in and rotate to get a better view of these genes.

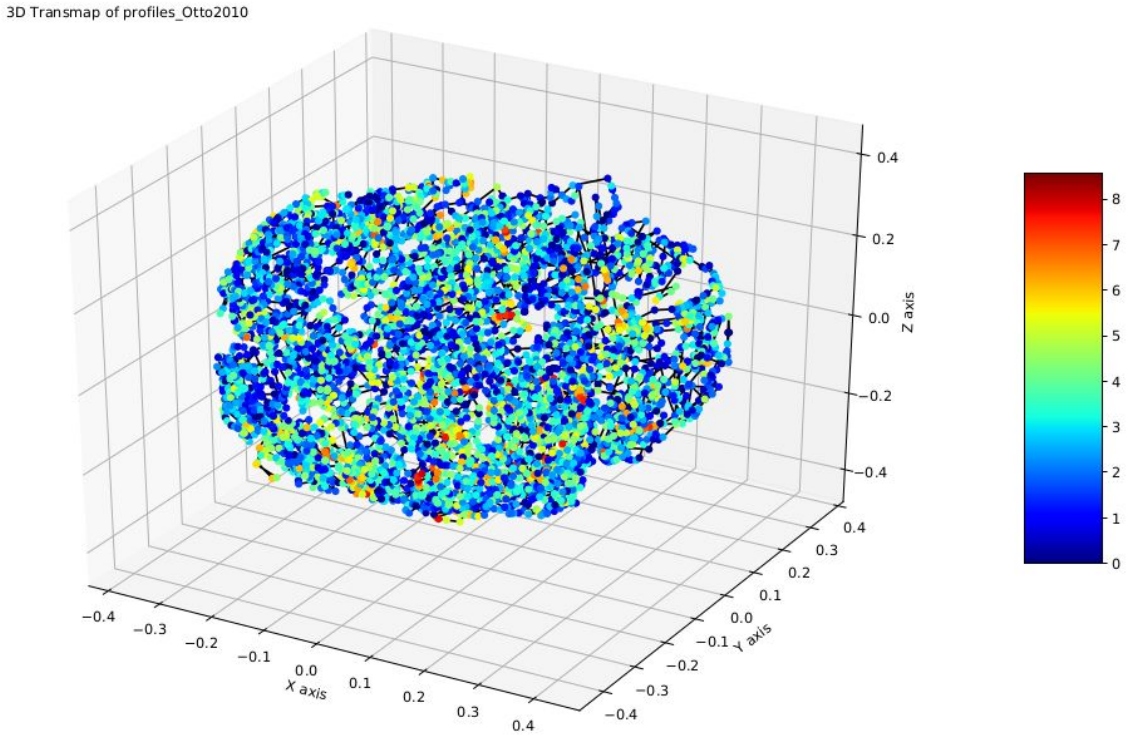

fig.3 : 3d transmap of *plasmodium falciparum* genes given by the program

Discussion

The python visualization tool for spatial correlations and associations between 3D genome organisation and gene transcription analysis has been implemented and gives a result, but in its current state the tool presents some issues, first, it is not fast enough, indeed, the computation of the correlation matrix takes from 10 to 12 minutes in my computer for the *plasmodium falciparum* data, we also notice that when we have a huge amount of genes to display it is hard to spot well all the genes, additionally like we can see it in fig.4 when the genes are close to each other or are on the same plane and we hover over them we get a list of gene names but we can't really not which one we are looking for, furthermore, the tool is good for early visualization of the relationship between the genome spatial organisation and the transcription, but not much can be induced from it and could use some improvement in providing results allowing deeper analysis, so the tool has still rooms to improve.

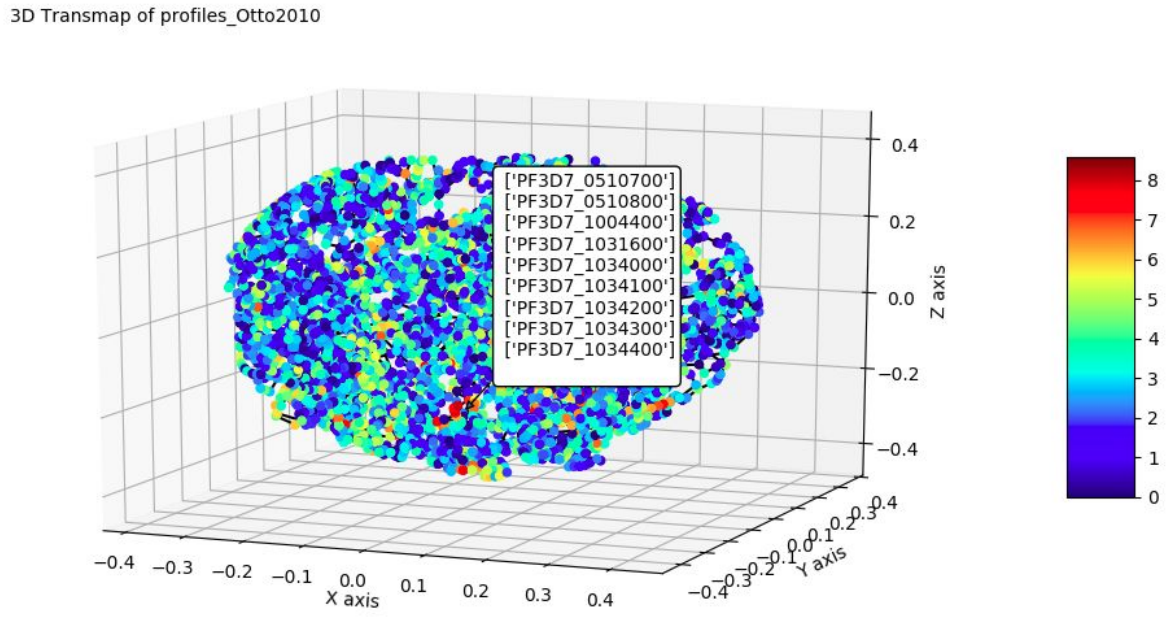

fig.4 : 3d transmap of *plasmodium falciparum* genes given by the program with mouse hovering on some genes

Possible solutions for some of the issues :

Here I'll present some potential solutions to the above mentioned issues, as said on the results section a solution for the unreadability of the result in the case of huge amount of genes is to zoom in to increase the spacing between the points, for the list of gene names on the hovering we can solve the problem by coloring the names text with the same color as the referred gene, for the duration it seems that the pandas corr function isn't fast so it would be possible to make my own correlation matrix computation function that would try to be on a lesser order of complexity or try to find another than pandas's correlation calculation function, my friend Adam for exemple used R's correlation function for a way faster way to get the matrix.

Conclusion

It has been shown that the relationship between the three-dimensional genome organization and the transcription provides an important biological interest, this Python tool gives us a three dimensional transcription map that can be very useful for an early visualization and analysis of the genes expression in a three dimensional architecture of the genome and thus allow the genome wide study of the associations and correlations between genome organization, the transcriptional activity and the genome regulation. But the application still needs improvement to be more efficient in particular on its computation duration that seems

quite long for what it does, one may also want a more advanced display of the result to make the analysis easier as it seems right now pretty hard to get accurate inductions from the current result display.

Acknowledgment

Thanks to **ImportanceOfBeingErnest** from Stackoverflow for providing a useful way to display annotations upon the hovering over a scatter point on the 3D plot, also thanks to Adam Bellaïche and Hélène Kabbech.

References

1- T. Cremer, C. Cremer, *Chromosome Territories, Nuclear Architecture and Gene Regulation in Mammalian Cells*, Nature Reviews Genetics. 2 (2001) 292–301.

2- G. Bascom, T. Schlick, *Linking Chromatin Fibers to Gene Folding by Hierarchical Looping*, Biophysical Journal. 112 (2017) 434–445.

3- M.R. Hübner, M.A. Eckersley-Maslin, D.L. *Spector, Chromatin organization and transcriptional regulation*, Current Opinion in Genetics and Development. 23 (2013) 89–95.

4- A.L. Sanborn, S.S.P. Rao, S.-C. Huang, N.C. Durand, M.H. Huntley, A.I. Jewett, I.D. Bochkov, D. Chinnappan, A. Cutkosky, J. Li, K.P. Geeting, A. Gnirke, A. Melnikov, D. McKenna, E.K. Stamenova, E.S. Lander, E.L. *Aiden, Chromatin extrusion explains key features of loop and domain formation in wild-type and engineered genomes*, Proceedings of the National Academy of Sciences.

5- S. Fanucchi, Y. Shibayama, S. Burd, M.S. Weinberg, M.M. *Mhlanga, XChromosomal contact permits transcription between coregulated genes*, Cell. 155 (2013) 606–620.

6- B. Tolhuis, et al., *Looping and interaction between hypersensitive sites in the active beta-globin locus*, Mol. Cell. 10 (6) (2002) 1453–1465.

7- J. Fraser, I. Williamson, W.A. Bickmore, J. Dostie, *An overview of genome organization and how we got there: from FISH to Hi-C*, Microbiol. Mol. Biol. Rev. 79 (3) (2015) 347–372.

8- J. Dekker, et al., *Capturing chromosome conformation*, Science 295 (5558) (2002) 1306–1311.

9- ML, M. (2019). *Sequencing technologies - the next generation. - PubMed - NCBI*. [online] Ncbi.nlm.nih.gov. Available at: https://www.ncbi.nlm.nih.gov/pubmed/19997069 [Accessed 6 Jan. 2019].

10- Waldispühl, J., Zhang, E., Butyaev, A., Nazarova, E. and Cyr, Y. (2019). *Storage, visualization, and navigation of 3D genomics data*.

11- Asbury TM, Mitman M, Tang J, Zheng WJ. *Genome3D: a viewer-model framework for integrating and visualizing multi-scale epigenomic information within a three-dimensional genome*. BMC Bioinformatics. 2010 Sep 2

12- Nowotny, J., Wells, A., Oluwadare, O., Xu, L., Cao, R., Trieu, T., He, C. and Cheng, J. (2019). *GMOL: An Interactive Tool for 3D Genome Structure Visualization*.

13- Butyaev,A., Mavlyutov,R., Blanchette,M., Cudré-Mauroux,P., Waldispühl,J. (2015) A low-latency, big database system and browser for storage, querying and visualization of 3D genomeic data. *Nucleic Acid Research*.

14- Ferhat Ay, Evelien M Bunnik, Nelle Varoquaux, Sebastiaan M Bol, Jacques Prudhomme, Jean-Philippe Vert, William Stafford Noble, and Karine G Le Roch. *Three-dimensional modeling of the P. falciparum genome during the erythrocytic cycle reveals a strong connection between genome architecture and gene expression*. Genome research, 24(6):974–88, 6 2014.

15- Sciencedirect.com. (2019). *Methods | 3D genome mapping and analysis methods | ScienceDirect.com*. [online] Available at: https://www.sciencedirect.com/journal/methods/vol/142/suppl/C [Accessed 6 Jan. 2019].

16- Jowhar, Z., Gudla, P., Shachar, S., Wangsa, D., Russ, J., Pegoraro, G., Ried, T., Raznahan, A. and Misteli, T. (2019). *HiCTMap: Detection and analysis of chromosome territory structure and position by high-throughput imaging*.

17- Kocanova, S., Goiffon, I. and Bystricky, K. (2019). *3D FISH to analyse gene domain-specific chromatin re-modeling in human cancer cell lines*.

18- Oshidari, R. and Mekhail, K. (2019). *Catch the live show: Visualizing damaged DNA in vivo*.