REPORT PROJECT:

3D TRANSCRIPTION MAP

Adam BELLAICHE M2 bioinformatics University of Paris 7 adam.bellaïche@gmail.com	Costas BOUYIOUKOS Assistant professor University of Paris 7 costas.bouyioukos@univ-paris-diderot.fr

Years: 2018-2019

INTRODUCTION

The aims of the project was to develop a python application able to analyse the spatial correlations and associations between 3D genome organisation and gene transcription. Provide and interactive 3D visualisation of the genome. A R application was developed by Costas BOUYIOUKOS but he considered his application to slow and decide to migrate in python.

"Genome conformation capturing techniques (from 3C to Hi-C) have lighted the way genomes are organised in the three-dimensional space (3D). At the same time, advances in next generation sequencing (NGS) and its application to transcriptomics (RNA-seq) allow to access the activity of whole transcriptomes with elevated accuracy.
Genome architecture both affects and is affected by transcriptional activity, long distance regulatory elements (enhancers) are reported to regulate, in a fine tuning manner, the transcription of a large amount of genes in eukaryotic genomes.
On that epigenomic level of gene regulation the proposed concept of transcription factories, that is the local concentration of transcriptional activity in 3D hotspots, is playing a key role.
We have introduced a method, relatively easy to its use and interpretation, that allows the genome wide study of the associations between genome architecture, transcriptional activity and genome regulation and with this project I want to develop a proper python application." .Costas Bouyioukos.

Here, a python application was implemented. We will see how this python application is implemented and show and analyse her results.

MATERIALS AND METHODS

Material

This programs was developed in python 3. So, it need python 3 and not python 2. Then it is constituted by 3 files:

- main.py: which doing main tasks helped by three other files,
- file_manager.py: which permits to manage and catch error when users indicate path file to the main,
- map3d.py: which contains a class to create and manage the 3D map. It means, like for a graph class, this class contains a data structure for 3D map. But it also contains functions to display the map and manage this display,
- correlation.R: which contains a little code R to do a correlation matrix and give it to main.py. We will see later why this R code is used.

So there are two languages used in this program: python 3 and R.
Moreover, in python, there are four required libraries:

- numpy
- matplotlib [2]
- pandas
- scipy.

Then to test the application, there are two dataset given by Costas BOUYIOUKOS:

- file called: SCHIZONTS.genes_pos, which contains 3D genome positions table of Plasmodium falciparum [1].

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	

Left to right columns: gene name, chromosome, x, y, z. Lines start with the gene name.

- file called: profiles_Otto2010, which contains gene expression table of Plasmodium falciparum [1].

	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	

Lines start with the gene name. Left to right columns: transcription activity at Hour H.

Methods

Building of the data structure for 3D map

Read datas

To get the data from the files program need to be launch with argument in his command line. In fact, the two first arguments is respectively: path to the file of 3D position, path to the file of genome expression. Then the program check if the files exist and use pandas library, because it is a good library to manage data frame, to store the files.

Calculate distance matrix and correlation matrix

Then, from the two data frames, program calculates the distance matrix and the correlation matrix:

	Distance matrix	Correlation matrix
Description	matrix which contains for each gene his distances with each other genes	matrix which contains for each gene his transcription correlation with each other genes. (Spearman)
Calcul	compute by using squareform and pdist from scipy library.	compute by using R cor function. Here we don't use pandas correlation function because it's take 30 minutes against 2 minutes in R.

Build the data structure of the 3D map

The 3D map class to be init needs: the correlation matrix, the genome position matrix and the distance matrix.

From the distance matrix, for one selected gene we store his N closer genes (parameters given by users). Then, in the correlation matrix we will take the correlation score between the selected gene and his N closer genes. By doing the sum of all this correlation scores we get the transcription correlation (called Tc) score for the selected gene which represents the associations between his 3D genome organisation and his transcription.

At the end, we get the 3D coordinates of the selected gene from the matrix of genome position.

With this four scalars: coordinates and Tc, the data structure of the 3D map is built, it is called a plot dictionary. In fact, this data structure is a python dictionary which is organized like this: {“gene name”: [x, y, z, correlation score]}. So keys are gene name and each value is a list which contains the 3D positions (x, y, z) and the Tc of the gene.

Visualization of the 3D map

At the end, the programs iterates on this plot dictionary to display a 3D scatter plot using matplotlib.
In fact, the plot dictionary contains all data required to do the plot:

- x,y,z: to place the point in the scatter plot.
- Tc: to color the gene in terms of his normalized Tc, calculated previously.
- gene name: to labelize the point.

It is true that’s each point (which represents a gene) is colored in terms of his Tc. But this color is attributed by a color map. A color map is an array of colors used to map pixel data (represented as indexes into the color table) to the actual color values. This color map permits to discern the information encoded by the data [4]. Of course, the graduated color map is plotted also to have a critic view on points’ colors.

To finish with the visualization, there is a little managing of events on the 3D plot. In fact, by clicking on a point, it display the gene name associated or by pressing “ctrl+alt+s” it save the current image in the directory results in pdf. It is possible to modify the 3D plot: rotate it, zoom on area...

Optimization of the time processing

To not lose time, each list is initialized and sized. There no dynamic variables except dictionaries. In fact, by initializing and sized lists, the time processing of the program is significantly reduced.

Moreover, the program has just three loops to read date, create the data structure of the 3D map and to plot points. The program try to not iterates more than one time on the same array_like, dictionary, it permits to not lose time by iterating on the same variable several times. And the program called library like pandas, numpy or matplotlib which are well-known by the community for their quality and permits to use functions which permit low cost in time processing.

We chose a dictionary for the data structure of the 3D map because it is a constant time access to value of the dictionary. And it permits to use key like the gene name.

RESULTS

time processing:

Like Costas BOUYIOUKOS said, the R application was to slow. In fact, it runs in 50 minutes with the datas [1]. In python, with the same datas, it runs in 2 minutes.

Visualization of the 3D map:

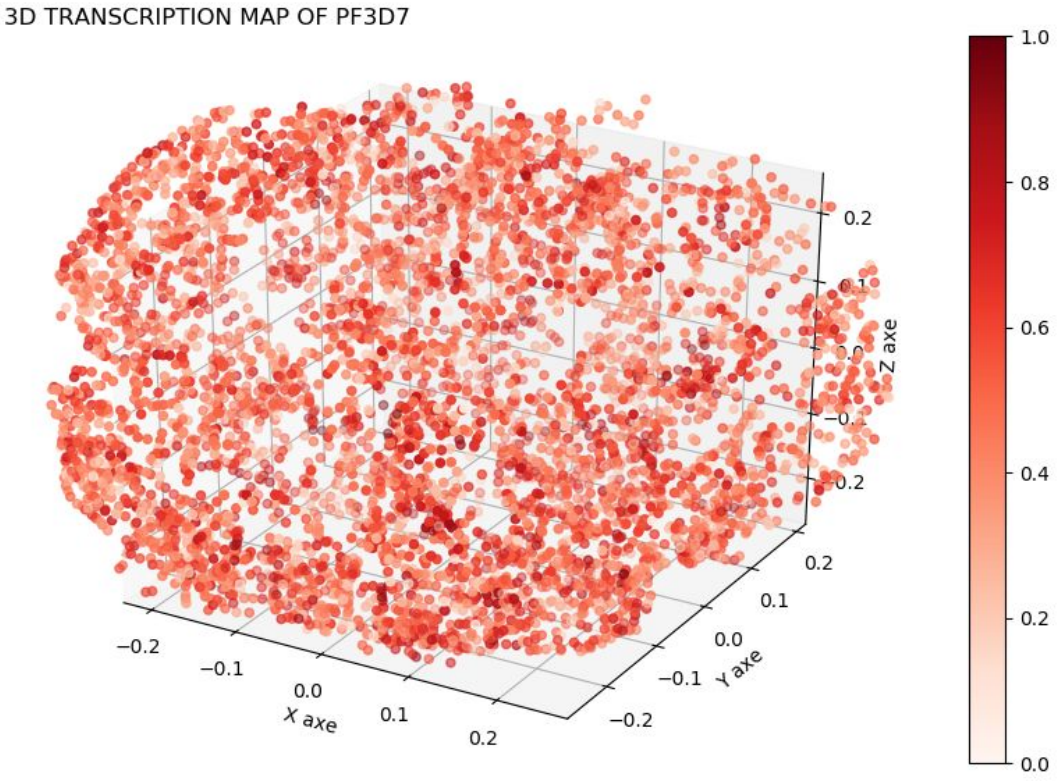

Image 1: 3D scatter plot got by launch the program with the data [1] mentioned in the material. More a point is red, more is transcription activity is linked the his 3D position in the genome.

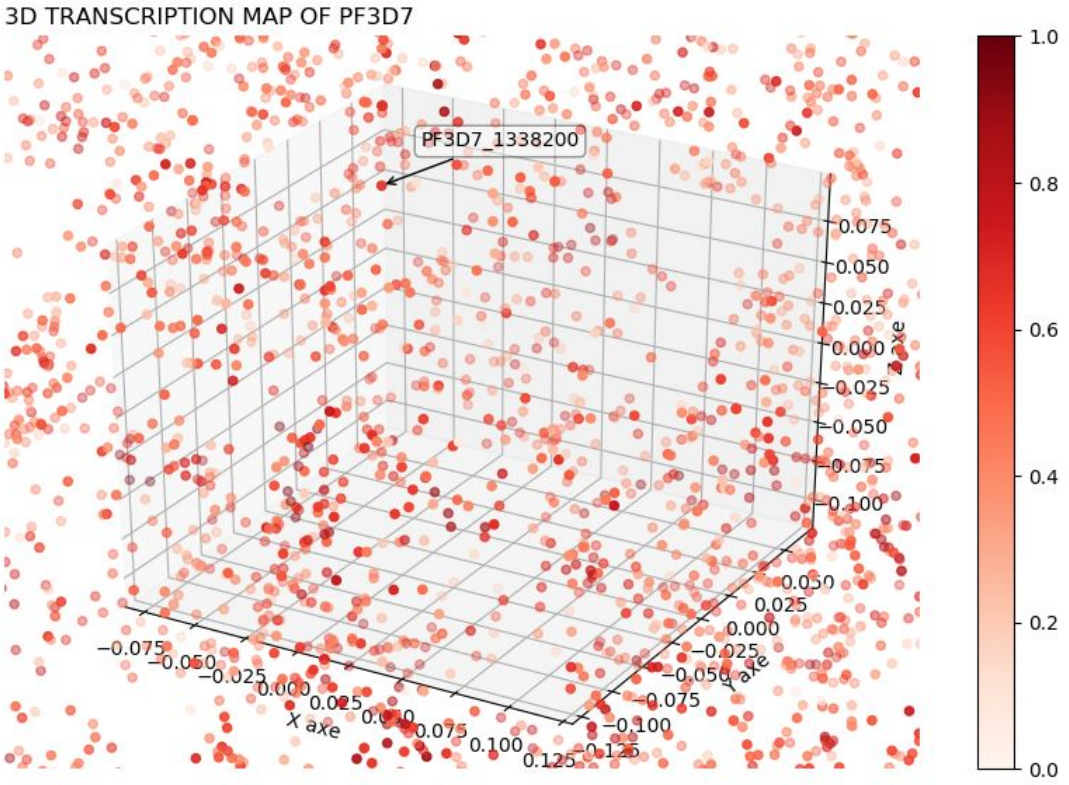

Image 2: example of a click on a point to get the gene name associated.

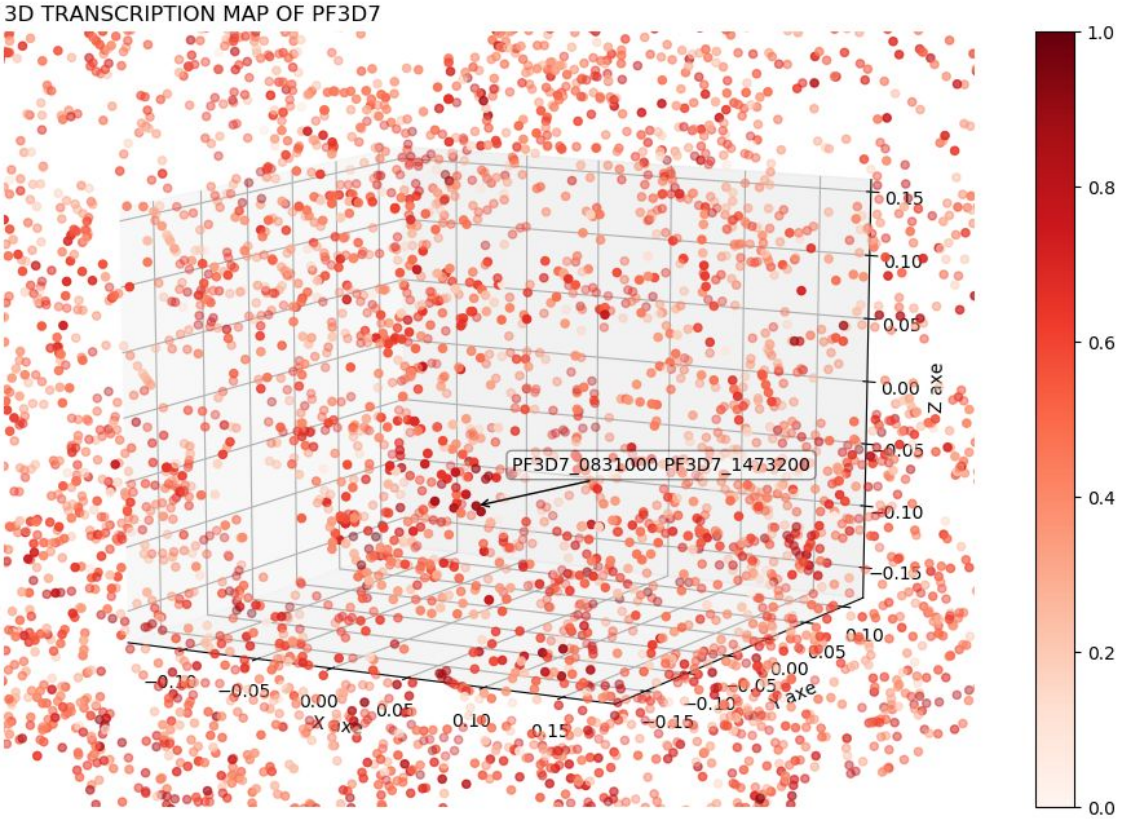

Image 3: example of area very red which means transcription activity is linked the his 3D position in the genome.

DISCUSSION

time processing:

The time processing of the python application is good. It is more faster than the R application, what is one goal of the project.

But it is possible to ameliore the the time processing by:

- find a better way than use R cor function, because it needs to save in the python code a data frame, to read it in R, to use the R cor function on it, to save the results and read results in python to have the final correlation matrix. As we said the pandas corr function take too much time (10 minutes). An example of better way is to use the library rpy2 or pyreadr to save or read faster the files of data frame. Or we can implement a cor function in python (here we missed times to do it).
- migrate the application in Cython.

Visualization of the 3D map:

Analyse:

In the Image 1, we can see the all 3D map of the genome of Plasmodium falciparum. This plot is made with the color map called “Reds” which is a color map of nuanced reds. More the transcriptional activity is correlated with the gene 3D position more the color approach dark red.
Here, we can have a good view of the link between the transcriptional activity and the 3D position of the genes. In fact, it is possible to see some area near to dark red. It means that the transcription activity of the Plasmodium falciparum genome is very correlated with the 3D genes positions. May be these genes are just promotor and target.

In the Image 2, we can see how the gene name appears when you click on a point. You can search the gene with this ID: PF3D7_1338200 on GenBank [5] for example. It returns that the gene is gene coding for a ribosomal protein. Moreover, this ID means this a gene of Plasmodium falciparum (PF) at the chromosome 13.

In the Image 3, we can see two gene names: PF3D7_0831200, PF3D7_1473200 which have color near dark red and two points closer. On GenBank, there are no links between this gene. But they come from different chromosome: 8 and 14 but they are closer. Their colors are dark red, and it means that may be, the transcription of theses genes are linked. So we ask us if this 3D map brings new interactions not reported on GenBank.

We try to find two closer dark red points on the plot and which have interactions reported on GenBank but it is not easier and the plot needs improvements.

Perspectives:

One perspective of the visualization is to improve the 3D plot. In fact, it should be nice to have a better distinction of the transcription area by: build a better colormap by follows Bang Wong or Shneiderman Ben rules for data visualization [4] or [3] , change the points' size in terms of the Tc. Or by ameliore the interactions with the plot for example by implement a research gene name bar which zoom on the point associated with the gene name in the bar.

Another perspectives is to see the results with another dataset, to be critic on the visualization. Add lot of event by pressed key on the keyboard like change the colormap or the dataset …

The most important perspective is to integrate an biologic enrichment with the gene selected by click. For example, if you click on one point, it brings out a windows with the informations of the gene.

Conclusion

Here we developed a python application able to display a 3D Transcription map from two files which contains 3D position of genome and gene expression like it has been shown in the material.

The goal of the project is respected. But there are lot of improvement to do with the visualization.

The program is available at https://github.com/toontun/3DTranscriptionMap.git or https://github.com/toontun/3DTranscriptionMap with all the data to test it. If you want to use it, sent me an email at: adam.bellaiche@gmail.com because the datas are private and so the repos git is private also. I will let you access to the repos. Follow the README to use it properly.

REFERENCES

[1]: Ay, F.; Bunnik, E. M.; Varoquaux, N.; Bol, S. M.; Prudhomme, J.; Vert, J.-P.; Noble, W. S. & Le Roch, K. G. Three-dimensional modeling of the P. falciparum genome during the erythrocytic cycle reveals a strong connection between genome architecture and gene expression. Genome Research, 2014, 24, 974-988.

[2]: Hunter, J. D. Matplotlib: A 2D graphics environment Computing In Science & Engineering, IEEE COMPUTER SOC, 2007, 9, 90-95.

[3]: Shneiderman, Ben.”The Eyes Have It: A Task by Data Type Taxonomy for Information visualizations”. Visual Languages, 1996. Proceedings., IEEE Symposium on. IEEE, 1996.

[4]: Bang Wong.”Point of view: Color Coding”. Nature Methods, 2010, VOL.7 NO.8, p573.

[5]: Benson DA, Cavanaugh M, Clark K, Karsch-Mizrachi I, Lipman DJ, Ostell J, Sayers EW. GenBank. Nucleic Acids Res. 2013; 45(Database issue):30–5.