NeoHiC: A web application for the analysis of Hi-C data

D. D'Agostino, P. Liò, M. Aldinucci, and I. Merelli. NeoHiC: A web application for the analysis of Hi-C data. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 12313, pages 98–107, 2020. DOI: 10.1007/978-3-030-63061-4_10

NeoHiC: a Web Application for the Analysis of Hi-C Data

Daniele D'Agostino[1], Pietro Liò[2], Marco Aldinucci[3], and Ivan Merelli[4]

[1] Institute for Applied Mathematics and Information Technologies "E. Magenes", National Research Council of Italy, Genoa, Italy dagostino@ge.imati.cnr.it
[2] Computer Laboratory, University of Cambridge, UK, Pietro.Lio@cl.cam.ac.uk
[3] Computer Science Department, University of Torino, Italy marco.aldinucci@unito.it
[4] Institute for Biomedical Technologies, National Research Council of Italy, Segrate (MI), Italy ivan.merelli@itb.cnr.it

Abstract. High-throughput sequencing Chromosome Conformation Capture (Hi-C) allows the study of chromatin interactions and 3D chromosome folding on a larger scale. A graph-based multi-level representation of Hi-C data is essential for proper visualisation of the spatial pattern they represent, in particular for comparing different experiments or for re-mapping omics-data in a space-aware context. The size of the Hi-C data hampers the straightforward use of currently available graph visualisation tools and libraries. In this paper, we present the first version of NeoHiC, a user-friendly web application for the progressive graph visualisation of Hi-C data based on the use of the Neo4j graph database. The user could select the richness of the environment of the query gene by choosing among a large number of proximity and distance metrics.

Keywords: Hi-C, graph database, web application, graph visualisation

1 Introduction

Modern bioinformatics aims at integrating different omics data to shed light into the mechanisms of gene expression and regulation that give rise to different phenotypes, in order to understand the underlying molecular processes that sustain life and to intervene into these processes by developing new drugs [1, 2] when pathological changes occur [3, 4]. In this context, the exploration of the 3D organization of chromosomes in the nucleus of cells is of paramount importance for many cellular processes related to gene expression regulation, including DNA accessibility, epigenetic patterns, and chromosome translocations [5, 6].

In particular, High-throughput sequencing Chromosome Conformation Capture (Hi-C) allows the study of chromatin interactions and 3D chromosome folding on a larger scale [7, 8]. The graph-based representation of Hi-C data produced, for example, by NuChart [9, 10] or CytoHic [11], which are software for representing the spatial position of genes in the nucleus, will be essential for creating maps where further omics data can be mapped, in order to characterize

different spatially associated domains. This visualisation is an effective complement of the traditional matrix-based representations, for example, produced by Juicer [5] [12] or TADbit [6] [13].

Contact matrices, or better their probabilistic models, allow creating representations that only involve two chromosomes, while graphs can describe the interactions of all the chromosomes using a graph-based approach. This representation highlights the physical proximity of genes in the nucleus in comparison to coordinate-based representations. The very same problem impairs representations based on Circos [7], which can characterize the whole genome in one shot, but fail to describe the physical proximity of genes. In previous works [14, 15] we showed some exciting results relying on the possibility of creating metrics for defining how far two genes are one from the other, with possible applications to cytogenetic profiling, to the analysis of the DNA conformation in the proximity of the nucleolus, and for describing the social behavior of genes.

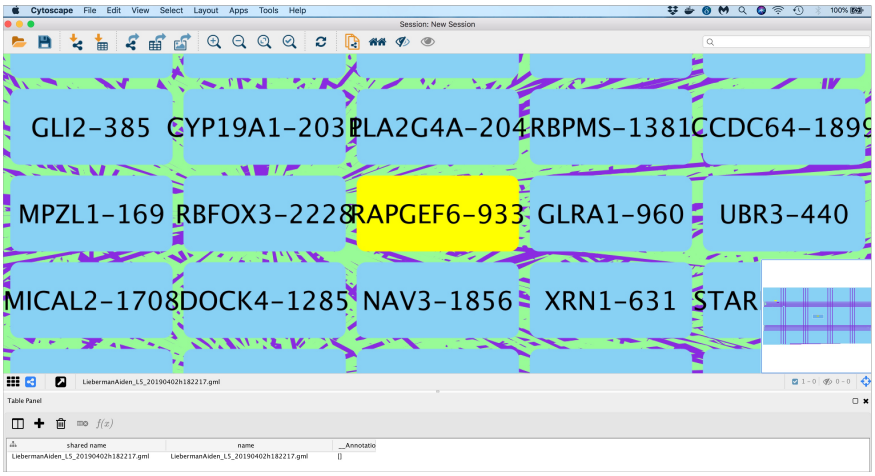

Fig. 1. The visualisation of a Hi-C network using Cytoscape.

However, the typical size of a graph achieved through a Hi-C analysis is in the order of thousands of nodes and hundreds of thousands of edges, which makes its exploration extremely complex, at least using the tools available. We tested both esyN [16], a tool for the construction and analysis of networks for biological research, and the well-known Cytoscape [8] [17] platform, with a network composed by about 2,400 nodes and 175,000 edges and we found many difficulties

[5] https://github.com/aidenlab/juicer
[6] https://github.com/3DGenomes/TADbit
[7] https://circos.ca/
[8] https://cytoscape.org/

in visualizing and analyzing such a massive network with these tools. For example Fig. 1 has been obtained using Cytoscape: it shows all the nodes of the network, but it is not easy to show only a subset of them, i.e. the neighborhood of a selected gene, as discussed later for example in Fig. 2, or also to analyze the edges provided by different experiments.

Such large networks represent an issue also for the effective storage of databases because in Hi-C data, most of the information is represented by the edges connecting genes. Therefore a proper way for their effective management is represented by Graph databases like Neo4j. But this solution has been considered only in these last years, see, for example [18]. At the same time, the most important repositories as STRING [19] or InterMine [20] are still based on relational databases.

For these reasons, we present the first version of NeoHiC, a web application specifically designed to manage and analyze graphs produced by investigating Hi-C data. In this version, we considered Neo4j as graph database management system and NuChart as the tool to compute Hi-C data.

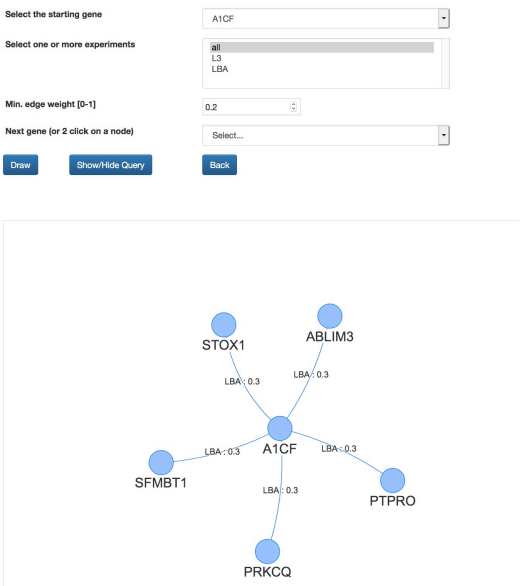

Fig. 2. The basic visualisation of NeoHiC.

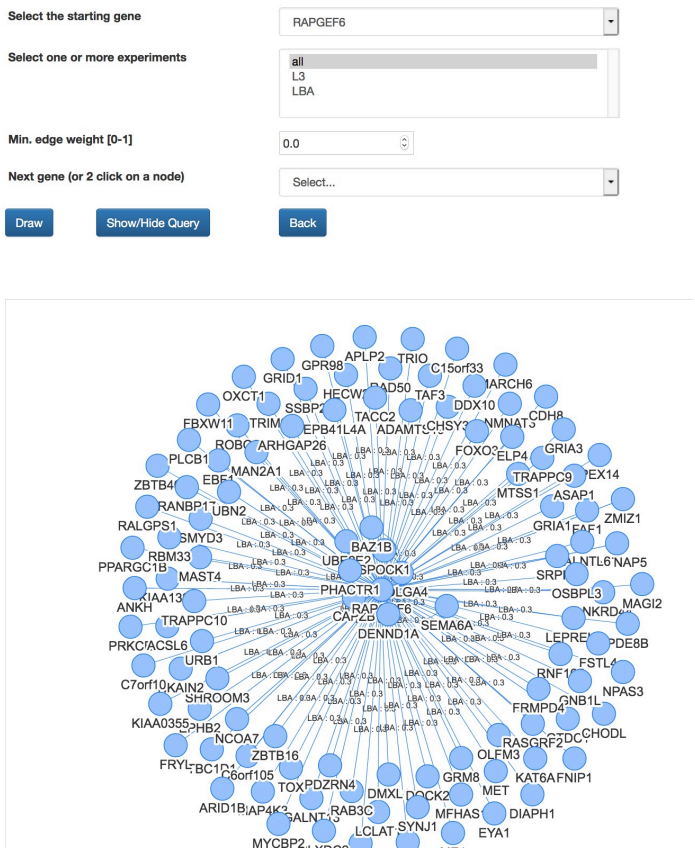

Fig. 3. The initial visualisation of the same network of Fig. 1 after selecting RAPGEF6 as a starting gene.

2 Materials and Methods

NeoHiC relies on the Neo4j graph database and modern web technologies, such as the Node.js JavaScript framework, used in many scientific applications and frameworks [21].

Graph databases

Graph databases are part of the NoSQL database family created to address the limitations of the existing relational databases. While the graph model explicitly lays out the dependencies between nodes of data, the relational and other NoSQL database models link the data through implicit connections.

In particular, in relational databases, references to other rows and tables are indicated by referring to primary key attributes via foreign key columns. Joins

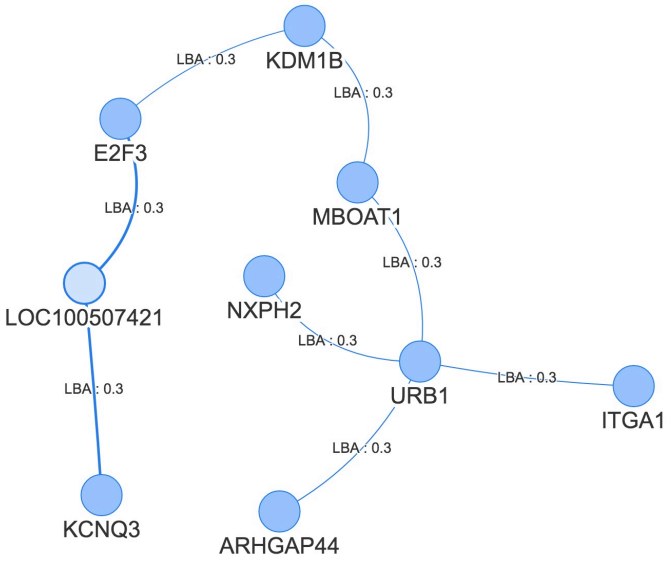

Fig. 4. The visualisation of a path in the network after five steps.

are computed at query time by matching primary and foreign keys of all rows in the connected tables. These operations are compute-heavy and memory-intensive and have an exponential cost. Moreover, when many-to-many relationships occur in the model, there is the need to introduce a JOIN table (or associative entity table) that holds foreign keys of both the participating tables, further increasing storage space, and the execution time of join operations.

On the contrary, in the data model of graph databases, the relationships have the same importance as the nodes. Database designers are not required to infer connections among entities using special properties such as foreign keys. For this, graph databases, by design, allow fast and straightforward retrieval of complex hierarchical structures that are difficult to model in relational systems.

Therefore the first step for creating the Web application has been the development of a tool for converting the graph-based representation of Hi-C data in a format that can be directly ingested by a graph database. The tool is a command-line Node.js script responsible for converting the output of NuChart, i.e. a file representing the graph of the chromatin conformation of the analyzed cells, in three files based on the comma-separated value format (CSV). The first file represents just a node describing the experiment, with associated the number of genes and links it produced. The other two files contain the two sets of nodes and edges. It is to note that an experiment usually adds new links between genes already existing in the database.

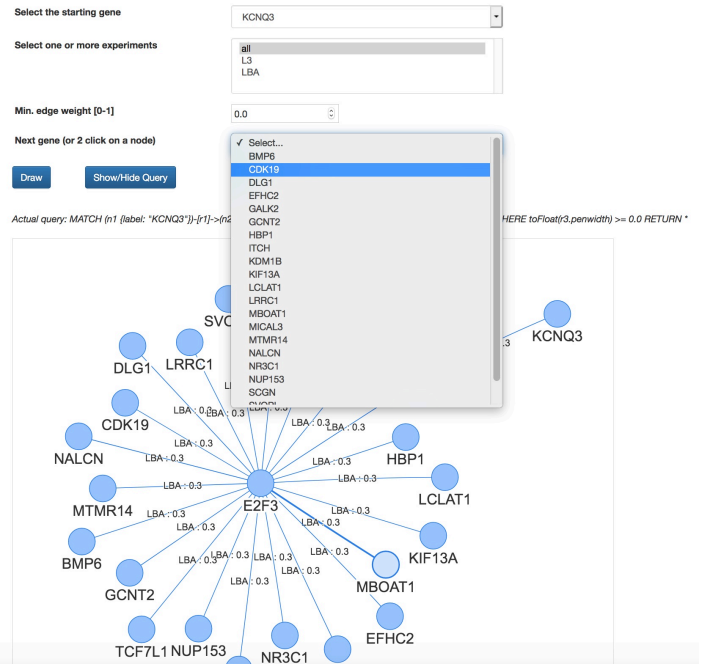

Fig. 5. The selection of the step after E2F3, useful in large networks.

Web Application

The Web application represents the second step. It represents an extension of the Neovis.js[9] visualisation library, which provides general-purpose graph visualisation powered by vis.js with data from Neo4j [22].

NeoHiC interacts with the database through the Javascript driver by performing queries like

```
MATCH (n1 {label: 'A1CF'})-[r1]-(n2) RETURN *
```

whose result is shown in Figure 2. These queries follow the Neo4j's graph query language Cypher[10]. The above query selects the node of the database labeled with 'A1CF' and retrieves the nodes representing the genes linked with it. This gene is linked with only five other genes that correspond to the nodes matched with $n2$, while the five edges correspond to $r1$. Edges are labeled with the experiments that created them, i.e., the Lieberman-Aiden et al. Hi-C data [8].

[9] https://github.com/neo4j-contrib/neovis.js/
[10] https://neo4j.com/developer/cypher-query-language/

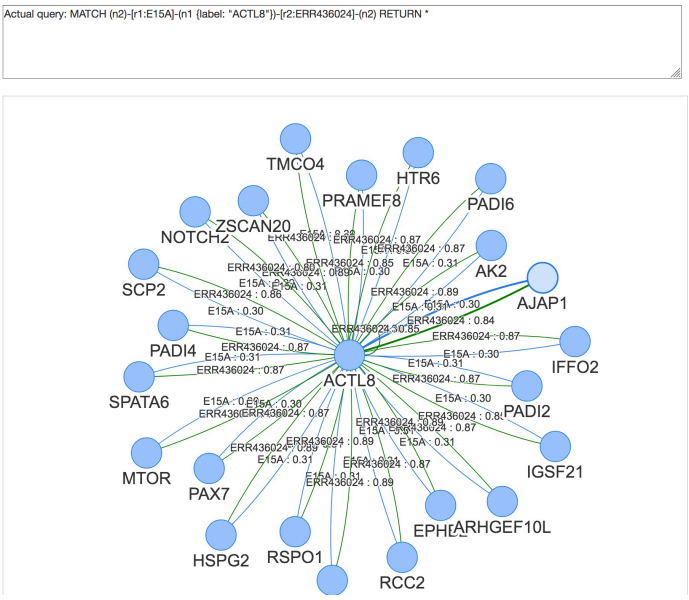

Fig. 6. The selection of a subset of the neighboring nodes of ACTL8.

3 Results

NeoHiC is based on the same approach adopted by STRING, where a protein-protein interaction network is expanded one step at a time by clicking on one of the visible nodes. Examples of visualisation are provided in Fig. 3 and Fig. 4.

The present version of the Web application allows users to select a starting gene and inspecting, step by step, the network described by the Hi-C data. In particular, the result is shown in Fig. 4 corresponds to the query

```
MATCH (n1 {label: "KCNQ3"})-[r1]-
(n2 {label: "LOC100507421"})-[r2]-
(n3 {label: "E2F3"})-[r3]-
(n4 {label: "KDM1B"})-[r4]-
(n5 {label: "MBOAT1"})-[r5]-
(n6 {label: "URB1"})-[r6]-(n7)
WHERE toFloat(r6.penwidth) >= 0.0RETURN *
```

where each of the five gene selection corresponds to add a $n_x - [r_x] - n_{x+1}$ pattern.

In most cases, the selection of a specific gene as the next step is complex, as in Fig 3, even if it is possible to zoom and move the network. We, therefore, added the possibility to select the next gene as a drop-down menu, as illustrated in Fig. 5.

It is also possible to go back of one step with the *Back* button if the users selected a wrong gene or they want to explore another path. It is also possible to go back to multiple steps by clicking on one edge, for example, on the edge between 'KDM1B' and 'MBOAT1' in Fig. 4.

Furthermore, it is possible to filter the neighboring genes based on the weight associated with the edges and limit the edges to those provided by a subset of the experiments included in the database. Last, the application shows the query corresponding to the shown network configuration, that can be exploited by expert users to interact directly with Neo4j via its command line or Web client to perform specific analysis tasks.

Figure 6 shows the result of a manually-inserted query that filters among the 262 neighboring genes of 'ACTL8', only those having a link for each of the two selected experiments.

4 Conclusion and Future Development

The NeoHiC web application, including the tool for converting the NuChart results, is available via GitHub[11].

It represents a first step in the development of a Web portal [23] for the sharing and analysis of Hi-C data. Currently, we offer a docker container for the execution of NuChart [24] and NeoHiC as two separate tools. Our first future development will be to deploy a cloud service on the HPC4AI platform [25] providing scientists with a portal to explore and publish novel experiments [26].

Besides this, we are adding more filters and options to improve the analysis of the data. At first, we will allow the comparison of the shortest paths linking two genes in two or more experiments, to statistically highlight differences in the chromatin conformation of different cells. Then we will integrate further analyses, such as the significance of the vertex clustering attitude (triangle), as described in [14].

The final goal is represented by the integration of 1D and 2D information on the Hi-C graphs to correlate the 3D conformation of the genome with regulatory and expression patterns and to adopt artificial intelligence to speed up the extraction of relevant results from the data.

Acknowledgments

This work has been funded by the Short-term 2018 Mobility Program (STM) of the National Research Council of Italy (CNR).

References

1. Chiappori, F., Merelli, I., Milanesi, L., Marabotti, A. (2013). Static and dynamic interactions between GALK enzyme and known inhibitors: guidelines to design

[11] https://github.com/dddagostino/neohic

new drugs for galactosemic patients. European journal of medicinal chemistry, 63, 423-434.
2. Merelli, I., Cozzi, P., D'Agostino, D., Clematis, A., Milanesi, L. (2010). Image-based surface matching algorithm oriented to structural biology. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 8(4), 1004-1016.
3. Viti, F., Merelli, I., Caprera, A., Lazzari, B., Stella, A., Milanesi L. (2008) Ontology-based, Tissue MicroArray oriented, image centered tissue bank, BMC bioinformatics 9(4), S4.
4. Banegas-Luna, A. J., Imbernon, B., Llanes Castro, A., Pérez-Garrido, A., Ceron-Carrasco, J. P., Gesing, S., ... & Pérez-Sánchez, H. (2019). Advances in distributed computing with modern drug discovery. Expert opinion on drug discovery, 14(1), 9-22.
5. Ling JQ, Hoffman AR (2007) Epigenetics of Long-Range Chromatin Interactions. Pediatr Res 61: 11R–16R.
6. Phillips-Cremins JE, Corces VG (2013) Chromatin Insulators: Linking Genome Organization to Cellular Function. Mol Cell 50(4): 461-474.
7. Duan Z, Andronescu M, Schutz K, Lee C, Shendure J et al. (2012) A genome-wide 3C-method for characterizing the three-dimensional architectures of genomes. Methods 58(3): 277-288.
8. Lieberman-Aiden, E., van Berkum, N.L., Williams, L., Imakaev, M., Ragoczy, T., ... & Dekker, J. (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. Science 326: 289-293. doi:https://doi.org/10.1126/science.1181369. PubMed: 19815776.
9. Merelli, I., Lio', P. & Milanesi, L. (2013). NuChart: an R package to study gene spatial neighbourhoods with multi-omics annotations. PLoS One, 8(9), e75146.
10. Tordini, F., Drocco, M., Misale, C., Milanesi, L., Lio', P., Merelli, I., Torquati, M. & Aldinucci, M. (2017). NuChart-II: the road to a fast and scalable tool for Hi-C data analysis," International Journal of High Performance Computing Applications", vol. 31, iss. 3, pp. 196-211.
11. Shavit, Y. & Lio', P. (2013). CytoHiC: a cytoscape plugin for visual comparison of Hi-C networks. Bioinformatics, 29(9), 1206-1207.
12. Durand, N. C., Shamim, M. S., Machol, I., Rao, S. S., Huntley, M. H., Lander, E. S., & Aiden, E. L. (2016). Juicer provides a one-click system for analyzing loop-resolution Hi-C experiments. Cell systems, 3(1), 95-98.
13. Serra, F., Bau, D., Goodstadt, M., Castillo, D. Filion, G., & Marti-Renom, M.A. (2017). Automatic analysis and 3D-modelling of Hi-C data using TADbit reveals structural features of the fly chromatin colors. PLOS Comp Bio 13(7) e1005665
14. Merelli, I., Tordini, F., Drocco, M., Aldinucci, M., Lio', P. & Milanesi, L. (2015). Integrating multi-omic features exploiting Chromosome Conformation Capture data, Front. Genet., 6:40.
15. Tordini, F., Aldinucci, M., Milanesi, L, Lio', P. & Merelli, I. (2016). The genome conformation as an integrator of multi-omic data: the example of damage spreading in cancer, Front. Genet., 7:194.
16. Bean, D.M., Heimbach, J., Ficorella, L., Micklem, G., Oliver, S.G. & Favrin, G. (2014) esyN: Network Building, Sharing and Publishing. PLoS ONE 9(9): e106035.
17. Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., ... & Ideker, T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome research, 13(11), 2498-2504.
18. Have, C. T. & Jensen, L. J. (2013). Are graph databases ready for bioinformatics? Bioinformatics, 29(24), 3107.

19. Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., ... & Kuhn, M. (2014). STRING v10: protein–protein interaction networks, integrated over the tree of life. Nucleic acids research, 43(D1), D447-D452.
20. Smith, R. N., Aleksic, J., Butano, D., Carr, A., Contrino, S., Hu, F., ... & Stepan, R. (2012). InterMine: a flexible data warehouse system for the integration and analysis of heterogeneous biological data. Bioinformatics, 28(23), 3163-3165.
21. Galizia, A., Roverelli, L., Zereik, G., Danovaro, E., Clematis, A., & D'Agostino, D. (2019). Using Apache Airavata and EasyGateway for the creation of complex science gateway front-end. Future Generation Computer Systems, 94, 910-919.
22. Lyon, W. (2018) Graph Visualization With Neo4j Using Neovis.js. Online, https://bit.ly/2vOmPkj
23. D'Agostino, D., Roverelli, L., Zereik, G., La Rocca, G., De Luca, A., Salvaterra, R., ... & Tiengo, A. (2019). A science gateway for Exploring the X-ray Transient and variable sky using EGI Federated Cloud. Future Generation Computer Systems, 94, 868-878.
24. Merelli, I., Fornari, F., Tordini, F., D'Agostino, D., Aldinucci, M., & Cesini, D. (2019). Exploiting Docker containers over Grid computing for a comprehensive study of chromatin conformation in different cell types. Journal of Parallel and Distributed Computing, 134, 116-127.
25. Aldinucci, M., Rabellino, S., Pironti, ... & F. Galeazzi (2018). HPC4AI, an AI-on-demand federated platform endeavour." in ACM Computing Frontiers, Ischia, Italy, 2018. doi:https://doi.org/10.1145/3203217.3205340
26. Aldinucci, M., Torquati, M., Spampinato, C., Drocco, M., Misale, C., Calcagno, C., & Coppo, M. (2014). Parallel stochastic systems biology in the cloud. Briefings in Bioinformatics, 15(5), 798-813.