Mining a massive RNA-seq dataset with biclustering: are evolutionary approaches ready for big data?

Patryk Orzechowski*†
University of Pennsylvania
Philadelphia, PA 19104, USA
patryk.orzechowski@gmail.com

Jason H. Moore
University of Pennsylvania
Philadelphia, PA 19104
jhmoore@upenn.edu

ABSTRACT

Finding meaningful structures in big data is challenging, especially within big and noisy data. In this short paper, we present the results of the application of 6 different biclustering methods to a massive human RNA-seq dataset with over 35k genes from over 125k samples. We assess which biclustering methods can handle that large data and compare the results to the mini-batch k-means, a popular clustering approach. Finally, we assess the importance of evolutionary-based approaches in biclustering 'big data'.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; • **Computing methodologies** → **Cluster analysis**; *Search methodologies*; Bio-inspired approaches; • **Theory of computation** → *Massively parallel algorithms*;

KEYWORDS

biclustering, data mining, evolutionary computation, big data

ACM Reference Format:
Patryk Orzechowski and Jason H. Moore. 2019. Mining a massive RNA-seq dataset with biclustering: are evolutionary approaches ready for big data?. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3319619.3321916

1 INTRODUCTION

Biclustering is an increasingly popular data mining technique, which seeks for sup-groups of rows and sub-groups of columns. The challenge is to detect multiple different patterns, for example ones that have the same value in a couple of row, columns, or are monotonously increasing. The main application of biclustering approaches is genetics. Biclustering is considered NP-hard.

The major aim of this study is to verify how selected method can handle a very large dataset. For this task a massive human genomic dataset called ARCHS4 was used [3]. The input data contains over 35k rows and 125k columns. The paper compares the running times and asks a question on the current standing and perspectives of evolutionary algorithms (EA) in biclustering.

2 METHODS

The following methods were applied to the genomic dataset ARCHS4:

Mini-batch K-means. A variant of a popular k-means clustering method was used as a baseline. The method for calculating distances between the rows uses batches, what allows to speed up computations and minimize memory utilization [13].

EBIC. Evolutionary search-based BIClustering [9, 11, 12] is a recently published hybrid biclustering algorithm [6–8] that utilizes multiple GPUs. The method, which is based on multiple evolutionary strategies, has previously presented high precision in multiple patterns detection on benchmark datasets.

iBBiG. Iterative binary bi-clustering of gene sets (IBBiG) uses genetic algorithm and iteratively identifies patterns in binarized data [1]. Each bicluster is assigned score based on its homogeneity score, which is determined using Shannon's Entropy.

FABIA. Factor analysis for bicluster acquisition (FABIA) [2] uses a multiplicative model for locating biclusters. Each bicluster is modeled as a product of two sparse vectors Λ and Z plus additional noise ϵ. The input matrix is modeled as a sum of p biclusters (1):

$$A = \sum_{i=1}^{p} \lambda_i z_i^T + \epsilon = \Lambda Z + \epsilon \tag{1}$$

QUBIC. The original approach of Li et al. [5] was refactored, speeded up and wrapped within Bioconductor package for R called QUBIC [15]. The method uses qualitative representation of the data and searches for heavy subgraphs in a graph with nodes representing genes and edges similarity between genes.

runibic. This recent method [10] integrates with Bioconductor and speeds up UniBic, one of the leading biclustering methods. It is based on detection of the longest common subsequences between the indexes of multiple pairs of sorted rows [14].

Plaid model. Plaid model [4] assumes that the values of the input matrix are sums of layers, according to (2)

$$a_{ij} = \sum_{k=1}^{K} \theta_{ijk} \rho_{ik} \kappa_{jk} \tag{2}$$

*corresponding author
†Patryk Orzechowski is also affiliated with Department of Automatics and Robotics, AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Krakow, Poland

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6748-6/19/07…$15.00
https://doi.org/10.1145/3319619.3321916

where K is the number of biclusters (called layers), and two other binary parameters (ρ_{ik} and κ_{jk}) specify if a row i and a column j are included in the bicluster. The lasts parameter θ_{ijk} determines the contribution of the bicluster.

3 RESULTS

The original file with data from the experiment described in [3] is provided in hierarchical data format (h5). After downloading the data file from the repository, genetic data was decompressed, normalized and log-scaled. The total size of the input dataset in comma separated format was over 33 GB. Our working environment included multiple Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.60GHz and 8 GeForce GTX 1080 Ti GPU devices (each GPU has 11 GB memory). All considered methods were expected to load a dataset, perform analysis and return (bi)clusters that included gene identifiers. The methods were run with their default parameters, apart from EBIC, for which two population sizes were used (5k and 10k), and mini-batch k-means, for which multiple number of clusters was provided.

The running time and memory consumption of the analyzed methods is presented in Fig. 1.

Table 1: Averaged running time and memory consumption of mini-batch k-means and biclustering approaches on ARCHS4 human dataset. (*) - notice that for EBIC only CPU memory is included, but it also uses GPU memory.

Method	Avg. time	Avg. mem.	Max. mem.
MiniBatchKMeans	467.1 mins	146637.3 MB	217737.9 MB
EBIC (5k/4 GPUs)	512.2 mins	(*) 39252.0 MB	(*) 53674.8 MB
EBIC (10k/8 GPUs)	833.1 mins	(*) 47915.2 MB	(*) 60710.1 MB
FABIA	3255.2 mins	172703.1 MB	187930.0 MB
QUBIC2	3568.7 mins	164555.59 MB	164555.59 MB
Plaid	Reached memory limit of 250000 MB		
runibic	Didn't converge in 120 hours (5 days)		
iBBiG	ERROR (long vectors unsupported)		

Compared to the baseline clustering, only EBIC managed to provide results with the similar time frame. Other biclustering implementations were either at least 4 times slower, or crashed because of the size of the data. EBIC was also memory efficient - it utilized much less memory compared to other biclustering methods and a baseline. This also held if total memory available on all GPU devices was summed (the actual utilization was lower).

4 CONCLUSIONS

The main advantage of using biclustering in comparison with clustering is smaller dimensionality of the resulting structures, as each bicluster covers only a subspace of columns. This allows for a much better interpretability and explainability of the findings compared to clustering, especially for the datasets with multiple columns.

The major aim of this study was to verify if and how existing biclustering implementations can handle large datasets. In this paper we presented the results of analyzing a massive genomic dataset, ARCHS4, with 6 different biclustering approaches with the context of a clustering method. To our best knowledge, this study involves the largest dataset that has ever been analyzed with any biclustering method. We have observed that only half of 6 biclustering approaches considered provided any result within the reasonable time frame.

Among the methods that successfully completed the task, an evolutionary-based approach EBIC presented similar running time and performance as the baseline clustering algorithm. The second genetic algorithm included in this study, iBBiG, unfortunately crashed prematurely.

To conclude, powerful methods based on GA have already been proposed, but there is still a great potential of using evolutionary algorithms in mining big data .

ACKNOWLEDGMENTS

This research was supported in part by PLGrid Infrastructure and by NIH grants LM010098, LM012601 and AI116794.

REFERENCES

[1] Daniel Gusenleitner, Eleanor A. Howe, John Quackenbush, Stefan Bentink, and Aedín C. Culhane. 2012. iBBiG: iterative binary bi-clustering of gene sets. *Bioinformatics* 28, 19 (07 2012), 2484–2492. https://doi.org/10.1093/bioinformatics/bts438 arXiv:http://oup.prod.sis.lan/bioinformatics/article-pdf/28/19/2484/13833634/bts438.pdf

[2] S. Hochreiter, U. Bodenhofer, M. Heusel, A. Mayr, A. Mitterecker, A. Kasim, T. Khamiakova, S. Van Sanden, D. Lin, W. Talloen, et al. 2010. FABIA: factor analysis for bicluster acquisition. *Bioinformatics* 26, 12 (2010), 1520–1527.

[3] Alexander Lachmann, Denis Torre, Alexandra B Keenan, Kathleen M Jagodnik, Hoyjin J Lee, Lily Wang, Moshe C Silverstein, and Avi Maáyan. 2018. Massive mining of publicly available RNA-seq data from human and mouse. *Nature communications* 9, 1 (2018), 1366.

[4] Laura Lazzeroni and Art Owen. 2002. PLAID MODELS FOR GENE EXPRESSION DATA. *Statistica Sinica* 12, 1 (2002), 61–86. http://www.jstor.org/stable/24307036

[5] G. Li, Q. Ma, H. Tang, A. H. Paterson, and Y. Xu. 2009. QUBIC: a qualitative biclustering algorithm for analyses of gene expression data. *Nucleic acids research* 37, 15 (2009), e101–e101.

[6] Patryk Orzechowski and Krzysztof Boryczko. 2016. Hybrid Biclustering Algorithms for Data Mining. In *Applications of Evolutionary Computation*, Giovanni Squillero and Paolo Burelli (Eds.). Springer International Publishing, Cham, 156–168.

[7] Patryk Orzechowski and Krzysztof Boryczko. 2016. Propagation-Based Biclustering Algorithm for extracting inclusion-maximal motifs. *Computing & Informatics* 35, 2 (2016).

[8] Patryk Orzechowski and Krzysztof Boryczko. 2016. Text Mining with Hybrid Biclustering Algorithms. In *Artificial Intelligence and Soft Computing*, Leszek Rutkowski, Marcin Korytkowski, Rafał Scherer, Ryszard Tadeusiewicz, Lotfi A. Zadeh, and Jacek M. Zurada (Eds.). Springer International Publishing, Cham, 102–113.

[9] Patryk Orzechowski and Jason H. Moore. 2019. EBIC: an open source software for high-dimensional and big data analyses. *Bioinformatics* (2019), btz027. https://doi.org/10.1093/bioinformatics/btz027

[10] Patryk Orzechowski, Artur Pańszczyk, Xiuzhen Huang, and Jason H Moore. 2018. runibic: a Bioconductor package for parallel row-based biclustering of gene expression data. *Bioinformatics* 34, 24 (2018), 4302–4304. https://doi.org/10.1093/bioinformatics/bty512

[11] Patryk Orzechowski, Moshe Sipper, and Xiuzhen Huang. 2018. EBIC: an evolutionary-based parallel biclustering algorithm for pattern discovery. *Bioinformatics* 34, 21 (05 2018), 3719–3726. https://doi.org/10.1093/bioinformatics/bty401

[12] Patryk Orzechowski, Moshe Sipper, Xiuzhen Huang, and Jason H Moore. 2018. EBIC: a next-generation evolutionary-based parallel biclustering method. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. ACM, 59–60.

[13] David Sculley. 2010. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*. ACM, 1177–1178.

[14] Zhenjia Wang, Guojun Li, Robert W Robinson, and Xiuzhen Huang. 2016. UniBic: Sequential row-based biclustering algorithm for analysis of gene expression data. *Scientific reports* 6 (2016).

[15] Yu Zhang, Juan Xie, Jinyu Yang, Anne Fennell, Chi Zhang, and Qin Ma. 2017. QUBIC: a bioconductor package for qualitative biclustering analysis of gene co-expression data. *Bioinformatics* 33, 3 (2017), 450–452. https://doi.org/10.1093/bioinformatics/btw635