2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 978-1-6654-0126-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/BIBM52615.2021.9669608

Integrative hierarchical ensemble clustering for improved disease subtype discovery

Bastian Pfeifer [1]*, Andrei Voicu-Spineanu[2], Michael G. Schimek[1] and Nikolaos Alachiotis[2]
[1]Institute for Medical Informatics, Statistics and Documentation. Medical University Graz, Austria
[2]Faculty of EEMCS, University of Twente, The Netherlands

***Abstract*—Multi-omics clustering methods are used for the stratification of patients into sub-groups of similar molecular characteristics. In recent years, a wide range of methods has been developed for this purpose. However, due to the high diversity of cancer-related data, a single method may not perform sufficiently well in all cases. Here, we propose a comprehensive framework for multi-omics hierarchical ensemble clustering. We provide a flexible environment that allows to build hierarchical clustering ensembles suitable for the available data and research goals. Survival analyses for data from The Cancer Genome Atlas (TCGA) indicate that our proposed ensembles provide more robust, and thus more reliable results than the state-of-the-art. We have implemented our architecture within the R-package HC-fused, which is freely available on Github.**

***Index Terms*—multi-omics, integrative clustering, ensemble clustering, disease subtyping, software optimization.**

I. Introduction

Integrative analyses of high-throughput molecular data contribute to a better understanding of disease-specific variations across patients. They allow for a comprehensive view on the disease, and enable their study on a system level.

In the last years, multi-omics clustering approaches were developed for the detection of disease subtypes. These methods group patients into sub-groups with similar biological characteristics. Newly diagnosed patients are classified into one of these sub-groups, therefore facilitating a more precise treatment. Multi-omics clustering approaches are typically utilized on gene expression (mRNA), DNA methylation, and micro-RNA data for the same set of patients. The corresponding data sets are frequently retrieved from The Cancer Genome Atlas (TCGA) database (https://cancergenome.nih.gov/), which represents one of the largest collections of multi-omics data sets. The ultimate goal of multi-omics analysis is to uncover the underlying biological processes causing and progressing the disease, allowing medical doctors to intervene timely, and to adopt medical treatment accordingly.

Individual clustering algorithms impose a particular structure onto the data, which can lead to different clustering results for the same data. Choosing a clustering algorithm for a given problem is not straightforward since clustering algorithms do not exhibit consistent behavior for different problems [1]. In this work we build upon and extend the R-package HC-fused [2] that can be used for integrative hierarchical clustering of multi-omics data sets. HC-fused long execution time, however, hinders the efficient analysis of a large number of patients. We present a substantially accelerated implementation of the HC-fused data fusion algorithm, which allows us to build complex hierarchical ensemble architectures for improved disease subtype discovery. Ensemble clustering, also called consensus clustering, combines multiple clustering models to produce a better result than that of individual clustering algorithms in terms of consistency, robustness, and quality [1].

The remaining of the paper is organized as follows. Section 2 provides background on integrative clustering. Section 3 discusses related work and sets our contribution in context. Section 4 introduces the proposed ensemble clustering approach. Section 5 reports on the results we obtained based on TCGA multi-omics cancer data sets. Section 6 discusses the presented work. Finally, we conclude with section 7.

II. Background

There are two types of integrative clustering approaches depending on the concerned data type, horizontal and vertical. Horizontal integration is the aggregation of same-type data, while vertical integration entails the analysis of heterogeneous omics data from the same group of patients [3]. A major problem that arises in vertical integration is that, when data sets are highly diverse with respect to their probabilistic distributions, simply concatenating them and applying single-omics methods is highly likely to bias the results. Another issue arises when the number of features differs across the data sets, resulting in more importance being assigned to a specific single-omics input over the others. Another classification of integrative clustering, based on when data integration takes places, distinguishes between early, intermediate, and late-integration approaches. Early-integration approaches first concatenate the data sets and then perform the data analysis. In intermediate integration methods, data of different types are first fused into a single view that is subsequently clustered to obtain the final cluster assignments. In late-integration methods, each omics data set is initially analyzed, and then the obtained information of interest is concatenated to a global view.

III. Related Work

A wide range of multi-omics clustering methods has been developed in the last years [4][5][6][2][7]. Most of them are based on standard clustering methods such as kmeans, hierarchical agglomerative clustering, and spectral clustering. However, they substantially differ in the way they perform

*Corresponding author: Bastian Pfeifer (bastian.pfeifer@medunigraz.at).

978-1-6654-0126-5/21/$31.00 ©2021 IEEE

data integration. One popular example of a fusion algorithm is called Similarity Network Fusion (SNF) [4]. For each omics, it models the similarity between patients as a network and then fuses these networks via an interchanging diffusion process. Spectral clustering is applied to the fused network to infer the final cluster assignments. A method that builds upon SNF is called Neighborhood-based Multi-Omics clustering (NEMO) and was recently introduced in [6]. The presented work provides solutions to partial data and implements a novel eigen-gap method [8] to obtain the optimal number of clusters. Another method is called PINSPlus [5]. Its authors suggest to systematically add noise to the data (via perturbation), and to infer the best number of clusters based on the stability against this noise. When the best k (number of clusters) is detected, binary matrices are formulated reflecting the cluster solutions for each single omic. A final agreement matrix is derived by counting the number of times two patients appear in the same cluster. This agreement matrix is then clustered by standard methods, such as kmeans or hierarchical clustering. An advanced approach to calculate this agreement matrix was recently developed within the R-package HC-fused [2].

In the present work, we provide a flexible and versatile framework for building ensembles based on hierarchical agglomerative clustering algorithms, implemented within the R-packgage HC-fused. We argue that different cancer types may consist of a high degree of pattern variability, and a single clustering method might not capture them all. Ensembles of clustering methods have the potential to provide more robust solutions. To the best of the authors' knowledge, this is the first work that explores the potential of ensemble clustering for the detection of disease subtypes. For the purpose of demonstration, four ensemble methods were used to analyze survival-related TCGA data sets. When the outcomes were compared with the state-of-the-art, the ensembles showed superior and more robust results.

IV. The new Approach

We propose a versatile and flexible framework for multi-omics integrative ensemble clustering. An overview of our framework is given in Figure 1. First, the multi-omics data sets are clustered using two different hierarchical clustering algorithms, lets say HC1 and HC2. This is done for each single-omics separately, resulting in omics-specific cluster solutions. These cluster solutions are represented as a network and their corresponding adjacency matrices are fused via the HC-fused algorithm (see Fig. 1). The obtained fusion similarity matrices are again clustered using the same hierarchical clustering methods (HC1 and HC2). The obtained networks are integrated to a single fusion similarity matrix. Finally, this similarity matrix is clustered by these two hierarchical algorithms (HC1 and HC2) and the consensus is calculated.

At this very last step, however, we do not expect the two clustering solutions to be much different, since the last fused similarity matrix already contains the consensus signals from both clustering algorithms. However, in our experiments we observed that slight differences can occur. Therefore we propose an algorithm that groups samples into one cluster in case the two algorithms (HC1 and HC2) agree on the cluster assignment. In case there is no consensus about the assignment, e.g only one of the clustering methods has grouped two samples into one cluster, the proposed algorithm will define a new cluster.

Here, we focus on the agglomerative hierarchical clustering algorithms as implemented within the R-package fastcluster [9]. The fastcluster package provides efficient implementations of eight different hierarchical clustering methods, namely single-linkage and complete-linkage clustering, an unweighted pair-group method using arithmetic averages (UPGMA) [10], a weighted pair-group method using arithmetic averages (WPGMA) [10] clustering, a weighted pair-group method using centroids (WPGMC) [11], an unweighted pair-group method using centroids (UPGMC) [10] clustering, and clustering based on Ward's minimum variance [12][13]. We have further developed the R-package HC-fused and now allow for building ensembles of any combination from the aforementioned clustering methods. We have studied four different hierarchical clustering ensembles; UPGMA-WPGMA, WPGMC-UPGMC, ward.D-ward.D2, and the combination of single linkage and complete linkage clustering. These are combinations of related clustering methods, but with different concepts to fuse two data points or clusters into a single one. There is no strategy to explore which method would work best on a specific data fusion problem. As a consequence different clustering ensembles should be taken into consideration. Here, our motivation was to consider well-established concepts.

Optimized hierarchical data fusion

To yield a practical implementation of our proposed ensemble framework, we initially improved runtime performance of the HC-fused fusion implementation. HC-fused is written in R, which is an interpreted language (the interpretation of R expressions takes place at runtime). This leads to longer execution times compared to compiled languages like C and C++. Therefore, to overcome the aforementioned disadvantage of the R language and to improve time efficiency, the HC-fused fusion algorithm was rewritten in C++. This conversion provides the benefit of having expressions converted by the C++ compiler directly into machine code prior to execution, thus reducing the overall execution time. To combine the capabilities of both the R language and C++, with R used for the front-end within the HC-fused package and C++ used for the backend (optimized C++ kernel), the interface between the two implementations was realized with the Rcpp library [14] that generates wrappers for the C++ functions and data structures to facilitate their use in R. Thereafter, a series of optimizations was applied to reduce execution times. These optimizations included (a) code reorganization, (b) removal of loop-invariant computations, (c) reduction of the function-call overhead in the inner-most for-loops, (d) optimization of data exchanges between the R and C++ implementations, (e) memory layout transformations for the better exploitation of spatial and temporal locality, and (f) dynamic pre-allocation of memory and improved internal memory management. During the development phase of the optimized version of HC-fused,

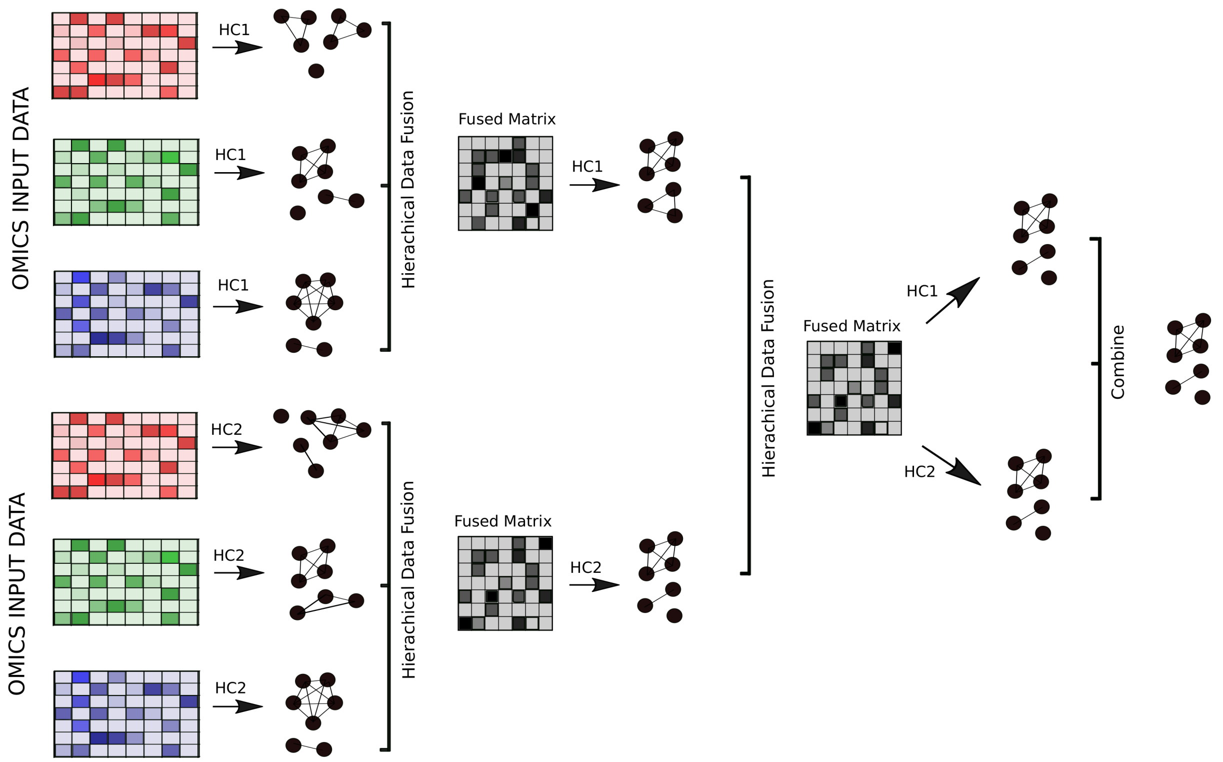

Fig. 1. Ensemble architecture for integrative hierarchical clustering. Shown is an illustrative example of an ensemble framework based on hierarchical clustering and data fusion algorithms. Initially, each single-omics is clustered by two different clustering methods, HC1 and HC2. HC1 and HC2 are two hierarchical clustering algorithms, which are implemented within the R-package fastcluster. The obtained cluster solutions are represented in networks and corresponding adjacency matrices. They are integrated via the HC-fused hierarchical network fusion algorithm, until a single fused matrix is obtained. The final fusion matrix is clustered by the HC1 and HC2 algorithms, and the cluster assignments are finally combined to one cluster solution.

two datasets were used for verifying functional correctness and assessment of performance. The first and smaller one consisted of 105 patients with mRNA and Methylation omics data, whereas the second and larger one comprised 849 patients and three types of omics: mRNA, Methylation, and miRNA. The final measurements showed an overall improvement of 784 times and 3384 times faster execution than the initial HC-fused R implementation, leading to better scalability of the optimized implementation for increased numbers of patients and of omics.

V. Results

We applied our proposed ensemble methods to four different cancer types, namely glioblastoma multiforme (GBM), kidney renal clear cell carcinoma (KIRC), liver hepatocellular carcinoma (LIHC), and sarcoma (SARC)). Our methodology was utilized on multi-omics data sets, including gene expression data (mRNA), DNA methylation, and micro-RNA for the same set of patients. The corresponding data were retrieved from http://acgt.cs.tau.ac.il/multi_omic_benchmark, a data repository which was recently proposed as a convenient benchmark for multi-omics clustering approaches [15]. Our data were preprocessed as follows: patients and features with more than 20% missing values were removed, and the remaining missing values were imputed with k-nearest neighbor imputation. In the methylation data, we selected those 5000 features with maximal variance in each data set. All features were then normalized to have zero mean and standard deviation one. The sizes of the data sets are shown in Table I. In a first investigation we clustered the multi-omics data sets with eight different agglomerative clustering methods, as implemented within the R-package fastcluster. The best number of clusters was determined by the silhouette coefficient (SIL), and the cluster solutions were integrated with the HC-fused fusion algorithm. In all cases, we set the number of HC-fused fusion iterations to 30. The maximal possible number of clusters was fixed at 10. We randomly sampled 30 times 100 patients from the data pool, performed survival analyses on these subsamples, and calculated the Cox log-rank tests for each of the 30 runs. The obtained p-values are displayed in Figure 2.

TABLE I
TCGA cancer data sets.

Cancer type	Patients	mRNA	Methylation	micro-RNA
GBM	538	12042	5000	534
KIRC	606	20531	5000	1046
LIHC	423	20531	5000	1046
SARC	265	20531	5000	1046

Displayed are the total numbers of features of each omics data set.

We observed that the performance varies across algorithms and cancer types. However, there is a strong indication that ensembles perform better than or equally good as each single member of the ensembles. The ward.D2 method, for instance, does not perform well on the LIHC data set, whereas the ward.D alternative produces significant results. Notably, the ensemble, combining both methods, is not negatively affected by the rather low performing ward.D algorithm. The same observation can be reported for the GBM cancer type. The performance of single linkage clustering is low, while the complete linkage alternative provides substantially better results. The combination of both, however, is not negatively affected by the former. These first insights suggest that ensembles stabilize the outcomes. Interestingly, hierarchical clustering based on ward.D seems to be specifically suited for LIHC cancer (see Fig. 2). All other hierarchical approaches do not produce significant p-values.

Furthermore, we compared our ensemble architectures with the state-of-the-art methods SNF (Similarity Network Fusion) [4], PINSplus [5], and NEMO [6]. For SNF, we set the number of neighbors for the k-nearest neighbor network to 20. We specified the α hyperparameter with 0.5. The number of diffusion iterations was fixed at 10. These values are within the range the authors of these R-packages suggest, and had performed best for the analyzed data set. To determine the number of clusters, we utilized the method based on rotation cost. In case of NEMO, we used the same number of neighbors for network generation. The number of clusters was determined by the eigen-gap method. For the PINSplus methodology, we set the maximal possible clusters to 10 and applied the perturbation method to infer the best number of clusters. The number of iterations was set to a minimum of 20 (default value). We utilized hierarchical clustering to obtain the final cluster solutions.

Our proposed ensemble framework provides the best performing method in three out of four cases, when the median Cox log-rank p-values are the criterion (see Table 2 and Figure 3). The combination of single and complete linkage clustering works best for the KIRC and the SARC data set, while the ensemble approach based on the ward algorithms is best for LIHC. In all cases, a significant p-value (for $\alpha = 0.05$ significance level) can be reported. A closer look at Table 2, however, suggests that the ward.D ensemble is the most stable one across all studied ensembles. Overall, at least one cancer type causes the other ensembles to produce highly non-significant results. The SNF method outperforms our proposed ensembles on the GBM data set. The mcquitty-average ensemble also provides a significant p-value in that case, but the p-value is slightly larger in comparison with SNF. Notably, the ensemble methods are the only ones which even provide significant results when a 0.01 significance level is assumed (see Table II).

We also compared the methods under consideration with respect to their computational efficiency (Table 3). The results are based on the KIRC data set for which we sampled 30 times 100 patients from the data pool. NEMO is the fastest algorithm. The second fastest algorithm is SNF. SNF and NEMO employ similar clustering techniques and its similar execution times are as expected. Our accelerated version of HC-fused outperforms the PINSplus approach. Even our proposed ensembles, which require far more fusion iterations and clustering, are faster than the PINSplus algorithm. The UPGMC-WPGMC and the ward.D-ward.D2 ensembles are the fastest ensemble clustering methods. For purpose of comparison, the original R-implementation of HC-fused takes about 30 minutes to terminate.

VI. Discussion

In this work, we proposed integrative hierarchical ensemble clustering for improved disease subtype discovery. We have further developed the R-package HC-fused, which now allows for building ensembles of arbitrary complexity. We compared the performance of four different ensemble architectures with the state-of-the-art methods SNF, NEMO, and PINSplus. We evaluated these methods based on four different cancer types. There is clear indication that our approach improves on the stability of obtained results. Furthermore, the best clustering solution was produced by one of our ensembles in three out of four cases. From the obtained results one can conclude that there might not be a single multi-omics clustering algorithm appropriate for all cancer types. Here we are providing a framework that allows a researcher to engineer combinations of hierarchical clustering methods to meet cancer-specific data demands. Overall, we believe that a general class such as hierarchical clustering can providing more flexibility with respect to ensembles, thus contributing to a better understanding of the genomic diversity across cancer types.

We observed that the SNF and NEMO algorithms are computational slightly faster than HC-fused. It should be noted, however, that the HC-fused fusion algorithm requires multiple runs on the exact same input matrices, and thus this process is specifically suited for multi-core execution. In contrast, the diffusion process implemented within SNF and NEMO is rather challenging to parallelize because employed similarity estimates depend on previous iterations. Moreover, the calculation of the best number of clusters within HC-fused is currently also executed sequentially, yet another option for optimization in a multi-core environment.

VII. Conclusion

We have developed an efficient ensemble framework for integrative clustering. An updated version of the R-package HC-fused is available from GitHub (https://github.com/pievos101/HC-fused). It includes a substantially improved code via C++ implementations and provides an easy-to-use function for ensemble building We plan to further develop the HC-fused package in various directions. Currently, the best number of clusters is inferred by the silhouette coefficient. Future work will evaluate a wide range of alternatives to determine an optimal number of clusters. For instance, we will interface the HC-fused program with the R-package Nbclust [16]. It provides 30 indices for the optimal number of clusters. Future work will also include the application of our approach to other data such as LinkedOmics (http://linkedomics.org [17]). Eventually, we will enable our algorithm to work with mixed input data, so that relevant clinical observations in the form of categorical data can be incorporated.

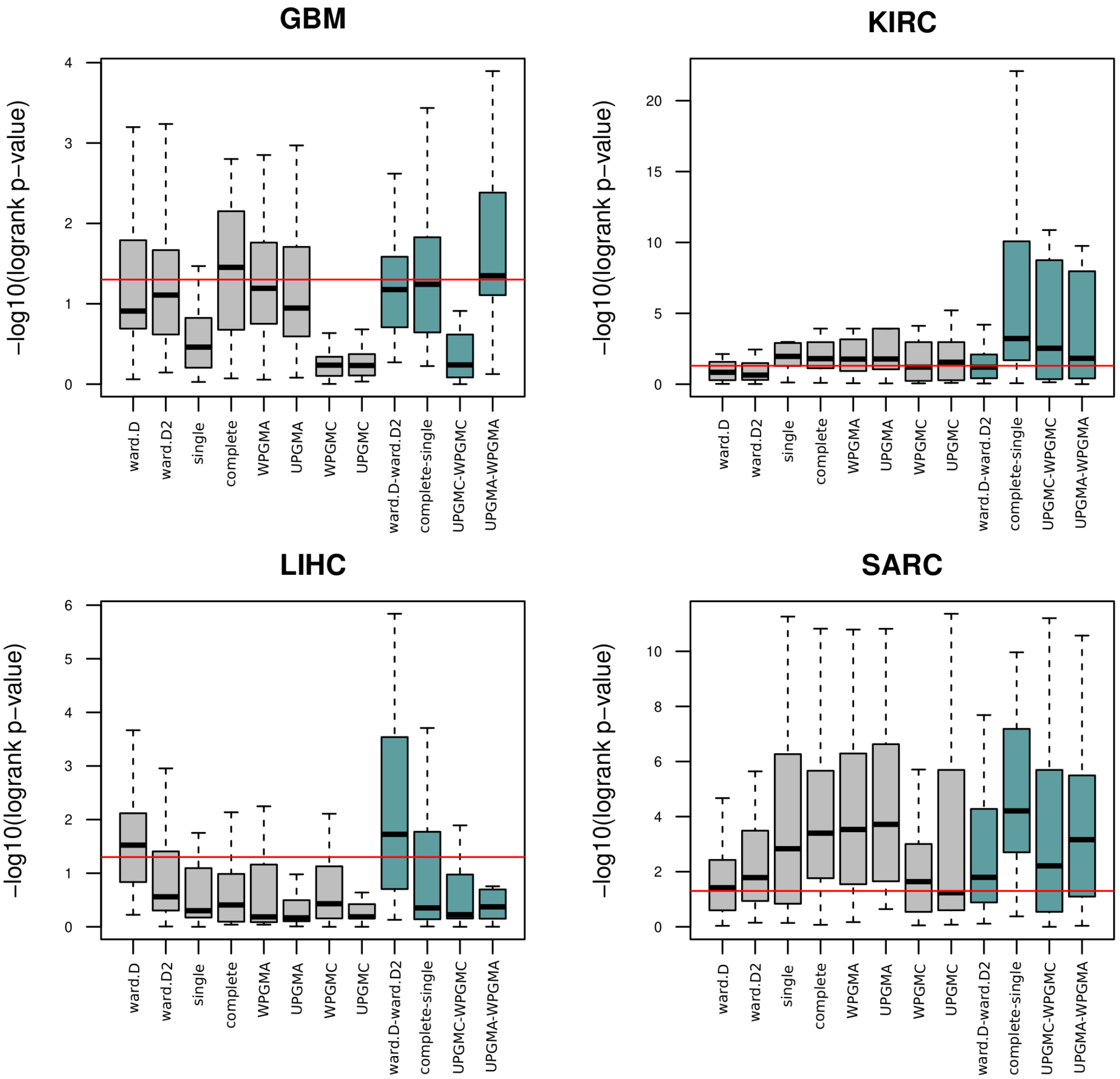

Fig. 2. TCGA results obtained from integrative hierarchical ensemble clustering. Boxplots of the p-values (on a negative log10 scale) are displayed for the four different cancer types (GBM, KIRC, LIHC, and SARC). The results of the ensemble approaches are highlighted in blue. The red line refers to $\alpha = 0.05$ significance level.

TABLE II
Survival analyses of TCGA cancer data.

Cancer type	Patients	SNF	PINSplus	NEMO	ward.D1-ward.D2	single-complete	UPGMC-WPGMC	UPGMA-WPGMA
GBM	538	**0.0113**	0.0614	0.0159	0.0668	0.0601	0.5767	0.0448
KIRC	606	0.0434	0.3098	0.3337	0.0629	**0.0006**	0.0031	0.0150
LIHC	423	0.5471	0.3785	0.1845	**0.0190**	0.4446	0.5945	0.4252
SARC	265	0.0161	0.0511	0.0856	0.0160	**0.0001**	0.0062	0.0007

Display of the median Cox log-rank p-values. The best performing method for each cancer type is highlighted in bold.

TABLE III
Computational demand.

Cancer type	Patients	SNF	PINSplus	NEMO	HC-fused	ward.D1-ward.D2	single-complete	UPGMC-WPGMC	UPGMA-WPGMA
KIRC	606	49.35	162.60	45.84	86.23	131.20	137.70	129	143

The elapsed time of the applied methods is shown in seconds.

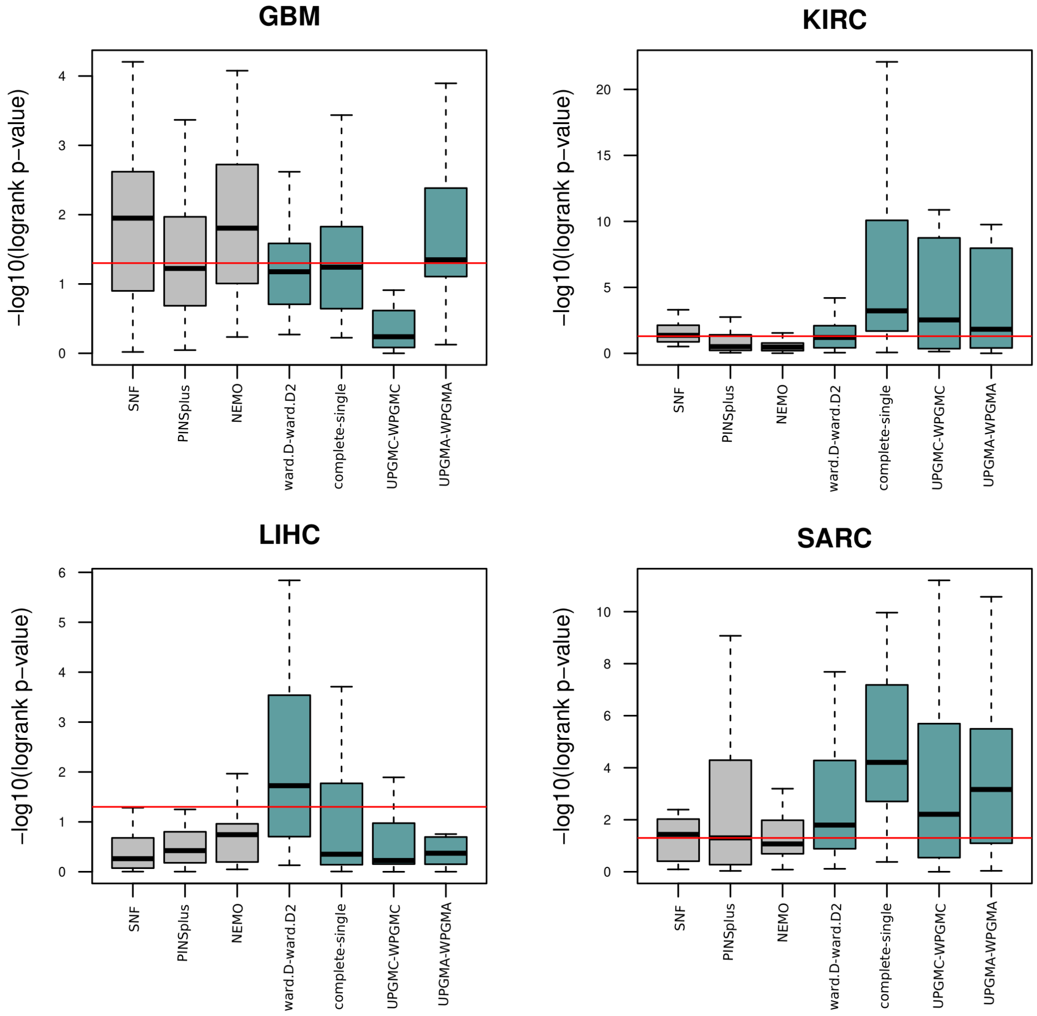

Fig. 3. TCGA results obtained from integrative hierarchical ensemble clustering. Boxplots of the p-values (on a negative log10 scale) are displayed for the four different cancer types (GBM, KIRC, LIHC, and SARC). Results for the ensemble approaches are highlighted in blue and compared with the state-of-the-art methods SNF, PINSplus, and NEMO. The red line refers to the $\alpha = 0.05$ significance level.

REFERENCES

[1] T. Alqurashi and W. Wang, "Clustering ensemble method," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 6, pp. 1227–1246, 2019.

[2] B. Pfeifer and M. G. Schimek, "A hierarchical clustering and data fusion approach for disease subtype discovery," *Journal of Biomedical Informatics*, vol. 113, p. 103636, 2021.

[3] S. Richardson, G. C. Tseng, and W. Sun, "Statistical methods in integrative genomics," *Annual Review of Statistics and its Application*, vol. 3, pp. 181–209, 2016.

[4] B. Wang, A. M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains, and A. Goldenberg, "Similarity network fusion for aggregating data types on a genomic scale," *Nature Methods*, vol. 11, no. 3, pp. 333–337, 2014.

[5] H. Nguyen, S. Shrestha, S. Draghici, and T. Nguyen, "Pinsplus: a tool for tumor subtype discovery in integrated genomic data," *Bioinformatics*, vol. 35, no. 16, pp. 2843–2846, 2019.

[6] N. Rappoport and R. Shamir, "Nemo: cancer subtyping by integration of partial multi-omic data," *Bioinformatics*, vol. 35, no. 18, pp. 3348–3356, 2019.

[7] G. Brière, É. Darbo, P. Thébault, and R. Uricaru, "Consensus clustering applied to multi-omics disease subtyping," *BMC Bioinformatics*, vol. 22, no. 1, pp. 1–29, 2021.

[8] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[9] D. Müllner, "fastcluster: Fast hierarchical, agglomerative clustering routines for r and python," *Journal of Statistical Software*, vol. 53, no. 1, pp. 1–18, 2013.

[10] R. R. Sokal, "A statistical method for evaluating systematic relationships." *Univ. Kansas, Sci. Bull.*, vol. 38, pp. 1409–1438, 1958.

[11] J. C. Gower, "A comparison of some methods of cluster analysis," *Biometrics*, pp. 623–637, 1967.

[12] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, 1963.

[13] F. Murtagh and P. Legendre, "Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion?" *Journal of Classification*, vol. 31, no. 3, pp. 274–295, 2014.

[14] D. Eddelbuettel, R. François, J. Allaire, K. Ushey, Q. Kou, N. Russel, J. Chambers, and D. Bates, "Rcpp: Seamless r and c++ integration," *Journal of Statistical Software*, vol. 40, no. 8, pp. 1–18, 2011.

[15] N. Rappoport and R. Shamir, "Multi-omic and multi-view clustering algorithms: review and cancer benchmark," *Nucleic Acids Research*, vol. 46, no. 20, pp. 10 546–10 562, 2018.

[16] M. Charrad, N. Ghazzali, V. Boiteau, and A. Niknafs, "NbClust: An R package for determining the relevant number of clusters in a data set," *Journal of Statistical Software*, vol. 61, no. 6, pp. 1–36, 2014. [Online]. Available: http://www.jstatsoft.org/v61/i06/

[17] S. V. Vasaikar, P. Straub, J. Wang, and B. Zhang, "Linkedomics: analyzing multi-omics data within and across 32 cancer types," *Nucleic Acids Research*, vol. 46, no. D1, pp. D956–D963, 2018.