PNAS RESEARCH ARTICLE | BIOPHYSICS AND COMPUTATIONAL BIOLOGY OPEN ACCESS

MoSBi: Automated signature mining for molecular stratification and subtyping

Tim Daniel Rose[a], Thibault Bechtler[a], Octavia-Andreea Ciora[a], Kim Anh Lilian Le[a], Florian Molnar[a], Nikolai Köhler[a], Jan Baumbach[b], Richard Röttger[c], and Josch Konstantin Pauling[a,1]

Edited by David Donoho, Stanford University, Stanford, CA; received October 4, 2021; accepted February 28, 2022

The improving access to increasing amounts of biomedical data provides completely new chances for advanced patient stratification and disease subtyping strategies. This requires computational tools that produce uniformly robust results across highly heterogeneous molecular data. Unsupervised machine learning methodologies are able to discover de novo patterns in such data. Biclustering is especially suited by simultaneously identifying sample groups and corresponding feature sets across heterogeneous omics data. The performance of available biclustering algorithms heavily depends on individual parameterization and varies with their application. Here, we developed MoSBi (molecular signature identification using biclustering), an automated multialgorithm ensemble approach that integrates results utilizing an error model-supported similarity network. We systematically evaluated the performance of 11 available and established biclustering algorithms together with MoSBi. For this, we used transcriptomics, proteomics, and metabolomics data, as well as synthetic datasets covering various data properties. Profiting from multialgorithm integration, MoSBi identified robust group and disease-specific signatures across all scenarios, overcoming single algorithm specificities. Furthermore, we developed a scalable network-based visualization of bicluster communities that supports biological hypothesis generation. MoSBi is available as an R package and web service to make automated biclustering analysis accessible for application in molecular sample stratification.

stratification | biclustering | subtyping | multiomics | pathomechanism

Significance

Molecular patient stratification and disease subtyping are ongoing and high-impact problems that rely on the identification of characteristic molecular signatures. Current computational methods show high sensitivity to custom parameterization, which leads to inconsistent performance on different molecular data. Our new method, MoSBi (molecular signature identification using biclustering), 1) enables so far unmatched high performance for stratification and subtyping across datasets of various different biomolecules, 2) provides a scalable solution for visualizing the results and their correspondence to clinical factors, and 3) has immediate practical relevance through its automatic workflow where individual selection, parameterization, screening, and visualization of biclustering algorithms is not required. MoSBi is a major step forward with a high impact for clinical and wet-lab researchers.

Optimizing treatments and improving patients ' health is the goal of precision medicine. In contrast to canonical medicine, where treatments are prescribed empirically (1), precision medicine aims to identify individually adapted treatments. Nowadays, diseases are commonly diagnosed based on the International Classification of Diseases. This assumes that diseases show similar symptoms in every individual; hence treatments are meant to act on the majority of symptoms. Patient stratification for precision medicine builds on the idea that a cohort of patients with varying or similar symptoms might have different molecular causes. They can then be stratified on the molecular level and divided into subgroups (2). Therefore, precision medicine wants to move away from classical disease definitions to characteristic signatures of molecular alterations which enable individualized treatments.

Achieving this requires an understanding of molecular disease mechanisms. Unsupervised machine learning methods are best suited since they uncover the inherent structure of the given data and do not require labeled data, which might be biased toward classical disease understandings (3). Unsupervised clustering methods seek to identify distinct subgroups over the entire features set, but it is unrealistic to assume that diseases manifest in all features. Instead, they are limited to a subtype-specific subset. Biclustering algorithms can meet this requirement.

Molecular data is usually available in data matrices with patient samples as columns and biomolecular features as rows. Biclustering algorithms cluster samples and biomolecules of a data matrix simultaneously. This results in sample groups with a molecular subset that characterizes the group. Numerous algorithms have been published, which try to tackle the problem from different angles. An overview of important concepts was published by Madeira and Oliveira (4).

Similar to clustering (5), evaluations of biclustering algorithms have shown differences in performance under various real-life and synthetic conditions (6, 7). A common way to improve the results of machine learning techniques is ensemble approaches, for example, for biomarker discovery (8). The goal is to improve robustness, consistency, novelty, and stability over what single algorithms could achieve (9). Also, for biclustering problems, ensemble algorithms have been proposed (10–16). Most of these ideas are adaptations of approaches for ensemble clustering. Some of the proposed methods have not been

Author contributions: T.D.R., J.B., R.R., and J.K.P. designed research; T.D.R., T.B., O.-A.C., K.A.L.L., F.M., N.K., and J.K.P. performed research; T.D.R. and J.K.P. contributed new reagents/analytic tools; T.D.R., T.B., O.-A.C., K.A.L.L., F.M., N.K., and J.K.P. analyzed data; and T.D.R., N.K., J.B., R.R., and J.K.P. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: josch.pauling@tum.de.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2118210119/-/DCSupplemental.

Published April 11, 2022.

implemented (10, 13) and therefore not easily accessible, while others are single-algorithm ensemble approaches that cannot overcome the limitations of one algorithm.

The analysis and interpretation of biclustering results can profit from visualizations, which show the content or relations between biclusters. Many approaches have been developed (17–23), which are often bound to specific algorithms or do not scale well for many biclusters (18).

Here we propose a multialgorithm biclustering ensemble approach for the stratification of molecular samples. In the manuscript, we 1) introduce the methodology and network visualization; 2) evaluate the performance on multiple experimental metabolomics, proteomics, and transcriptomics datasets; 3) with a framework for synthetic data generation, evaluate the approach on synthetic data; 4) apply our approach in a multiomics context; and 5) present open-source software to make biclustering more accessible for research.

Results

A Multialgorithm Ensemble Biclustering Approach. The steps of our ensemble approach (MoSBi—molecular signature identification using biclustering) are described in Fig. 1*A*; for full details, please refer to *Materials and Methods*. At first, we selected a set of established or recently developed biclustering algorithms (Table 1), which are executed independently. Next, similarities between all biclusters are calculated. The similarity is described by the degree of overlap, meaning the more samples and features shared between biclusters, the higher their similarity. Highly similar biclusters point toward the same pattern in the data. Similarities are filtered for random overlaps, and a bicluster network is generated with biclusters as nodes and connections between them if they exceed a higher than random similarity (for details, see *Materials and Methods*). This removes overlaps of biclusters that are likely to occur randomly and do not carry meaningful overlaps. The same network without the filtered random overlaps is shown in *SI Appendix*, Fig. S1. While biclusters with similar disease subtypes are still close together, the overall connectivity in the network is significantly higher. The example network shown in Fig. 1*A* reveals several highly connected communities in the network, which are not as strongly connected with each other. By using the Louvain modularity, such communities can be extracted and converted into ensemble biclusters. Two thresholds control the size of the resulting ensemble biclusters. We previously successfully utilized the principle of MoSBi to identify de novo subtypes of nonalcoholic liver disease based on clinical lipidomics data (34).

Before evaluating the performance of MoSBi on multiple omics datasets, we selected a public thymic epithelial tumor dataset (35) to show the application and potential of our approach. Ku et al. (35) measured the proteome of 134 tumor, tumor-adjacent, and normal thymus samples and revealed significant differences in the proteome signatures of thymoma subtypes. In Fig. 1*A*, the similarity network of predicted biclusters colored by sample groups can be seen. Node sizes were scaled according to the number of samples. It provides an overview of the match of predicted biclusters with known information about samples, in this case, cancer subtypes/tissues. While being a central part of the workflow, to compute ensemble biclusters, networks also serve as a visualization of biclustering predictions.

It is immediately obvious that clusters of nodes can be found in the network, indicating biclusters with a similar set of samples and features. This can be observed by similar color distributions of biclusters clustering together. The clusters show high intraconnectivity, but also connections to other clusters. This means that some signatures are shared between network communities. After applying the Louvain modularity, these clusters result in network communities.

Some communities, in particular, communities 2, 4, and 8, predominantly consist of type A, B, and AB thymoma. We performed Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway enrichment of protein sets from the ensemble biclusters (Fig. 1*B*). All selected communities showed significant repair mechanism pathways, which is well known for tumors to influence those pathways. Additionally, community 2, which includes samples of all thymoma subtypes, indicating a common signature on the proteomic level, showed two cancer-related terms.

We investigated the occurrence of proteins and samples in biclusters belonging to one community (Fig. 1*C*). A difference in the distributions of samples and features can be observed. The distribution of features is strongly positively skewed with a very low mode. In contrast, the sample distribution, for example, for communities 5 and 8, has a mode close to one. This shows that biclusters inside the same community (after filtering edges for random overlaps) can carry very different features and samples. By setting thresholds, ensemble biclusters can be restricted to point to consistent patterns in the data or to allow for variability.

In Fig. 1*D*, we visualize the affiliation of biclusters to algorithms, which predicted them. This reveals that biclustering algorithms tend to identify overlapping regions in the data, resulting in highly connected communities consisting only of one algorithm. This shows the necessity of taking the results of multiple biclustering algorithms into account and relying on not one but many different algorithms to capture patterns in the data beyond the specificities of a single algorithm. While we observe a good overlap of some network communities with the tumor subtypes, some individual (unconnected) biclusters also show high overlaps, for instance, with the type B thymoma. The strength of the MoSBi algorithm lies in the aggregation of biclusters. The visualization also helps to identify and analyze these individual biclusters, if they exhibit a high consensus with relevant information such as biological factors.

The results above demonstrate the power and utility of the workflow to establish a sophisticated biclustering analysis, to generate biological hypotheses.

Individual Biclustering Algorithms vs. MoSBi. Next, we compared the individual performances of available biclustering algorithms and contrasted them with the performance of MoSBi. For that, we selected six published and publicly available datasets from the metabolomics, transcriptomics, and proteomics disciplines (details in *SI Appendix*, Table S1). All datasets were analyzing cancer tissues or investigated cancer subtypes. As a gold standard, we used the condition match score to quantify the overlap between predicted biclusters and sample labels (see *Materials and Methods*), where the relevance describes how well predicted biclusters correspond to known labels, and recovery describes how well the labels were recovered by predictions. Additionally, Gene Ontology (GO) and KEGG pathway enrichment was performed to evaluate the gene sets in predicted biclusters.

The match between predictions and sample groups can be seen in Fig. 2*A*. It reveals a heterogeneous performance of the individual biclustering algorithms. Spectral only predicted biclusters in two out of the six scenarios. The iterative signature algorithm (Isa) has the highest recovery on the Tang et al. (36) metabolomics and Ku et al. (35) proteomics data and both transcriptomics datasets but has a poor performance on Yang et al. (37) metabolomics and Wiśniewski et al. (38) proteomics data. While having a good

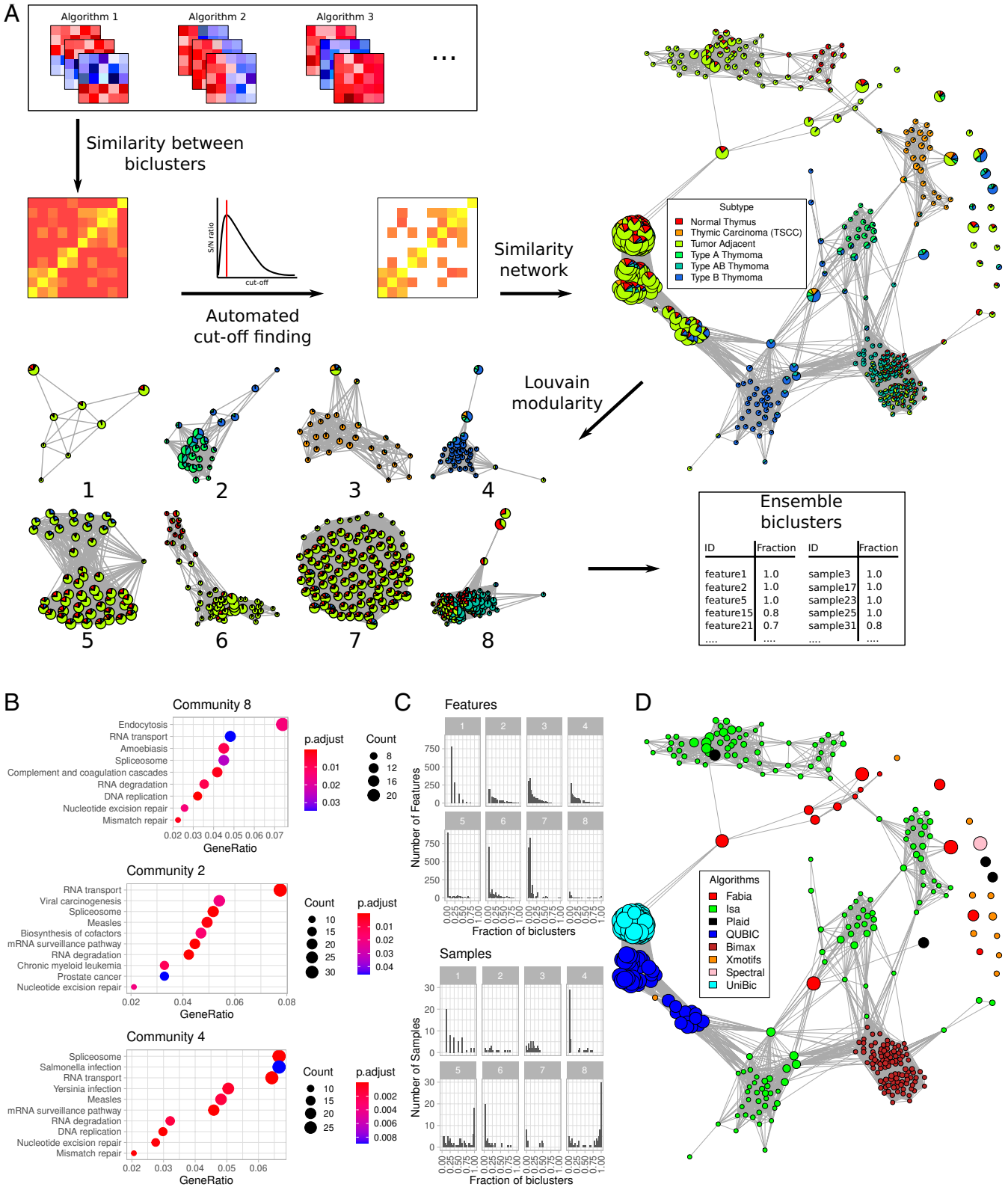

Fig. 1. Workflow of MoSBi with exemplary network visualizations. (*A*) Steps of the MoSBi approach. First, biclusters are predicted by multiple algorithms, and a similarity matrix is computed, which is then filtered for larger than random overlaps, using an error model. The matrix is then converted to a network that can be visualized with metainformation about samples or features. Louvain communities are then extracted and converted into ensemble biclusters. As an example, the bicluster network of proteomics data from Ku et al. (35) is shown. Nodes represent biclusters, with edges between them if their overlap exceeds the error threshold. (*B*) KEGG pathway enrichment for features of selected communities 2, 4, and 8. (*C*) Frequency of features (*Upper*) and samples (*Lower*) in biclusters that belong to one community. (*D*) Bicluster network of proteomics data from Ku et al. (35). Node colors represent algorithms, by which they were predicted.

Table 1. List of evaluated biclustering algorithms in alphabetical order

Algorithm	Publication
BicARE	Gestraud et al. (24)
Bimax	Prelić et al. (25)
CC	Cheng and Church (26)
Fabia	Hochreiter et al. (27)
Isa	Bergmann et al. (28)
Plaid	Lazzeroni and Owen (29)
QUBIC	Zhang et al. (30)
Quest	Murali and Kasif (31)
Spectral	Kluger et al. (32)
UniBic	Wang et al. (33)
Xmotifs	Murali and Kasif (31)

The results of algorithms can be imported and accessed with our MoSBi R package or executed using the webtool.

recovery, Isa never scores best on relevance. Similar behavior can be observed for Plaid, which, on average, performs very well for relevance, but shows low recoveries. It can also be observed that Plai is the only algorithm that reached a relevance and recovery higher than 0.5, and achieved this in one proteomics dataset. We then applied our ensemble approach to the predictions of all algorithms per dataset (Fig. 2*A*, black marker). The ensemble approach is one of the two best performing tools in either recovery or relevance in all other datasets, except for the Tang et al. (36) metabolomics data, where we could observe high overlaps with other clinical confounders (*SI Appendix*, Fig. S2). On metabolomics data, with fewer features compared to sequencing data, the communities can additionally be visualized as cooccurrence networks (*SI Appendix*, Fig. S3). Over all six datasets, MoSBi performed second best, on average, by relevance and second best by recovery after Plaid and Isa, which both have poorer performances on the other scale.

To investigate the performance of the algorithms on the gene level, we performed KEGG pathway and GO biological process enrichment for the proteomics and transcriptomics datasets. In KEGG enrichment (Fig. 2*B*), The Biclustering Analysis and Results Exploration (BicARE) algorithm predicted the most biclusters with at least one significantly enriched term in three datasets. Interestingly, it did not stand out when investigating sample group labels. The ensemble method again showed a better performance than the average of biclustering algorithms. The same holds for the enrichment of biological processes with GO terms (Fig. 2*C*). Since all investigated proteomic and transcriptomic datasets were cancer related, we searched specifically for enriched KEGG pathways including the word "cancer," "carcinoma," or "tumor" (Fig. 2*D*). On the Wiśniewski et al. (38) proteomics data, only Isa and BicARE found significant terms for biclusters, but only at very low frequencies. In the Ku et al. (35) proteomics data, MoSBi found the most significant terms and, on the two transcriptomics datasets, the second most after Fabia (factor analysis for bicluster acquisition) and BicARE.

This reveals that individual biclustering algorithms peak in one or another measure or dataset, but in an unpredictable manner. However, the MoSBi ensemble approach is more consistent and therefore more reliable for biclustering analysis.

Performance on Synthetic Data. Evaluation on experimental data is preferable since it accurately resembles the real-life application of biclustering and stratification. Unfortunately, two-dimensional (2D) gold standards are usually not available, since many factors are influencing the molecular state of samples. Synthetic data can overcome this problem. This is frequently done to evaluate biclustering algorithms (6, 25, 39).

Based on the synthetic data generation of Prelić et al. (25), we developed a workflow to create synthetic scenarios, where one or multiple properties can be investigated (*SI Appendix*, *Synthetic Evaluation Scenarios*). We repeated previous scenarios from Prelić et al. (25) and added scenarios, covering sparsity, overlaps, and mixed sizes (*SI Appendix*, Table S2), and evaluated them on biclustering algorithms (*Materials and Methods* and *SI Appendix*, Figs. S6–S10). Since molecular omics data can include missing values, we investigated the effect of sparsity on the performance of biclustering algorithms (Fig. 3*A*). While the overall performance of all algorithms decreases with increased sparsity, Fabia and Isa showed a higher resilience until a sparsity of 20% (percentage of missing values in the matrix; *SI Appendix*), after which the results deteriorated. The relevance was more robust against sparsity and did not decrease as strongly as the recovery.

So far, synthetic evaluation has focused on the assessment of individual characteristics of the data (e.g., noise or size). Using our workflow and knowledge from previous synthetic scenarios, we defined a complex scenario, incorporating all previously mentioned manipulations to the data (Fig. 3*B*). We evaluated all approaches in this scenario and added a negative binomial background to simulate unique molecular identifier RNA sequencing (RNAseq) data (Fig. 3 *B*, *Left*). Performance analysis separated the tools into two groups: clearly higher performing tools consisting of Fabia, Isa, and MoSBi, and the rest performing significantly inferiorly. Fabia shows the best recovery, and the ensemble approach shows the best relevance, but only marginally above Fabia and Isa. Even with the poor performance of many algorithms, MoSBi can still achieve high recovery and relevance. Algorithm selection has an influence on every ensemble approach; therefore, excluding the worst-performing algorithms from the ensemble approach yields a high increase of the relevance of the ensemble approach, while the recovery remains similar (*SI Appendix*, Fig. S11*A*).

Being an average, the relevance does not characterize every distribution correctly, but is widely used in biclustering evaluation studies. We investigated the relevance distribution of all algorithms independently (Fig. 3*C*) and combined (Fig. 3*D*). Some distributions are skewed. The combined distribution is positively skewed, showing that the majority of biclusters have a very low overlap with the gold standard. Predictions by the ensemble approach show a different distribution (Fig. 3*E*), where the majority of biclusters have a score above 0.5. Since an ensemble approach is sensitive to the performance of the underlying biclustering algorithms, we selected the best-performing algorithms and repeated the analysis (*SI Appendix*, Fig. S11 *C* and *D*). As can be seen, the performance of MoSBi is even more evident, showing the importance of the utilized algorithms. On the other hand, it shows that the approach can achieve a good performance, even with some poorly performing algorithms included. By combining highly overlapping biclusters, MoSBi can reduce the number of mismatched biclusters. This also shows that the relevance distribution can give more detailed insights into algorithm performance. MoSBi additionally reduces the number of biclusters drastically, making an investigation of all predictions more manageable. Analysis of the MoSBi parameters (*SI Appendix*, Fig. S12) showed that the row and column thresholds should be in the range of 0.02 and 0.2. The relevance increases with higher minimum community size thresholds, whereas the recovery decreases. An application-specific trade-off has to be decided by users. The number of randomizations for the similarity cutoff estimation does not affect the performance of MoSBi.

To investigate the performance of all algorithms under the best conditions, we optimized their parameters to achieve the best possible performance (*SI Appendix*, Fig. S13). This showed that algorithms can produce markedly better results given correct

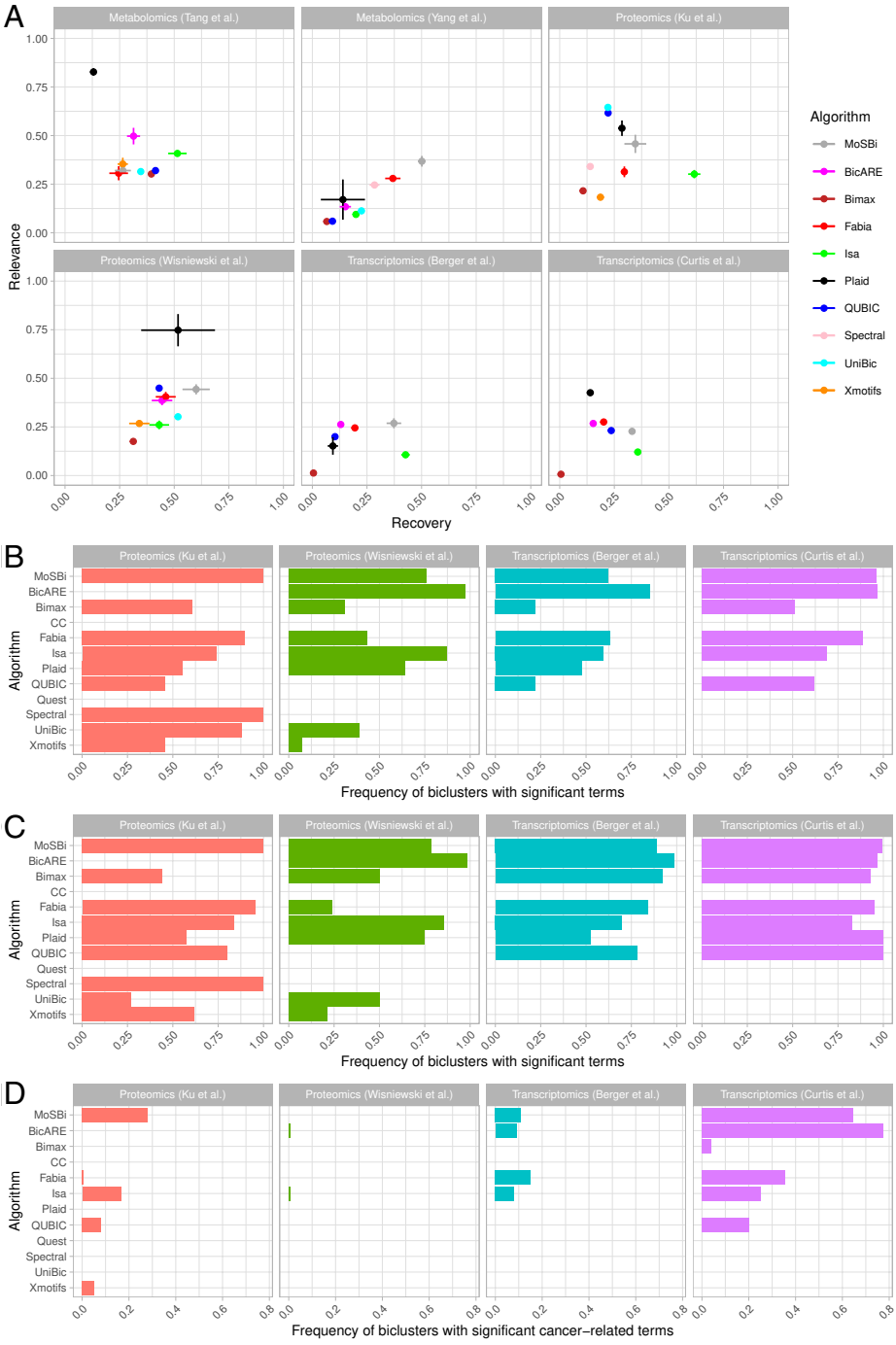

Fig. 2. Performance of MoSBi and individual biclustering algorithms on cancer-related omics data. Data was used from Ku et al. (35), Wisniewski et al. (38), Berger et al. (41), Curtis et al. (44), Tang et al. (36), and Yang et al. (37). For further information about the data, see *SI Appendix*, Table S1. (*A*) Recovery and relevance for the condition match score of biclustering tools based on samples for cancer (subtypes). (*B*) Frequency of predicted biclusters per algorithm, with one or more significant KEGG terms (adjusted *P* value cutoff < 0.05). (*C*) Frequency of biclusters with one or more significant GO terms from the "biological process" category. (*D*) Frequency of predicted biclusters per algorithm, with one or more cancer-related KEGG terms.

parameters compared to their standard parameters, in Fig. 3*B*. However, this is time consuming and only possible for data with an existing gold standard. The differences between the two complex synthetic scenarios showed that parameters and performances vary widely between datasets. Therefore, an ensemble method offers an easier method to achieve good performance independently of parameter optimization.

Biclustering in a Multiomics Context. Since biclustering requires a data matrix as input, it can naturally be applied to multiomics data, when merged into one data matrix. To investigate the performance of MoSBi in a multiomics context, we used the TCGA breast cancer cohort from the Xena Platform (40), which provides omics data for multiple breast cancer subtypes. RNAseq, microRNA (miRNA), and protein data were run independently and combined for all biclustering algorithms. All resulting bicluster networks (Fig. 4*A*) appear similar, with big basal communities and multiple communities consisting mainly of the LumA or LumB subtype, often highly interconnected. The protein data network shows a less distinct basal community, whereas the miRNA data network shows Her2 samples mixed with LumB samples.

In the next step, we evaluated the performance of the biclustering algorithms on the different data types. A consistent perfor-

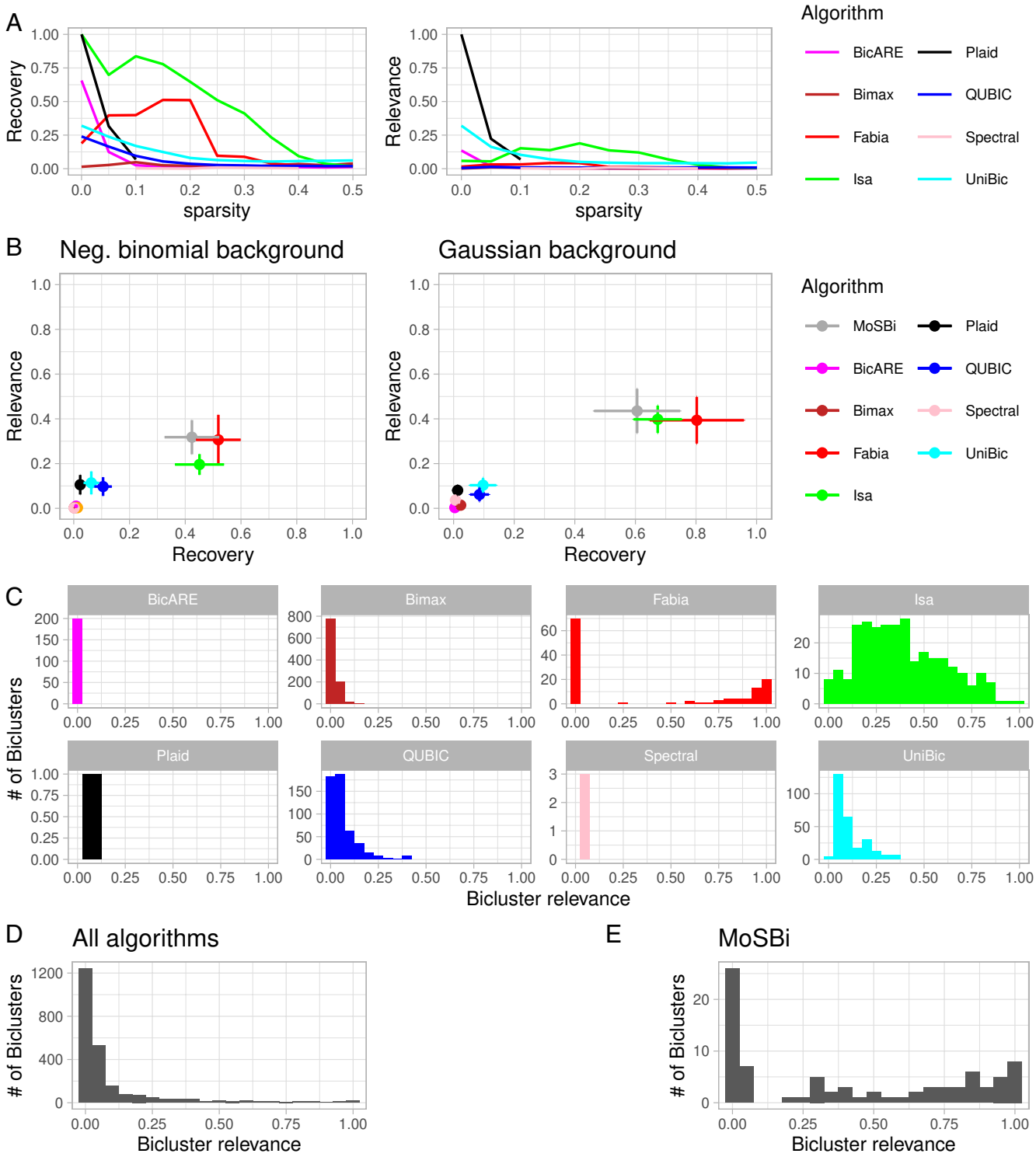

Fig. 3. Evaluation of biclustering algorithms on synthetic data. (*A*) Recovery and relevance of biclustering algorithms with increasing sparsity, for one hidden shift bicluster. (*B*) Performance of biclustering algorithms and ensemble approach on a synthetic scenario including different bicluster types, sizes, sparsity, and noise with a negative binomial distributed background (*Left*) and normally distributed background (*Right*). (*C*) Relevance distribution of biclustering algorithms for the scenario shown in *B*, *Right*. (*D*) Relevance distribution of all algorithms summed up from *C*. (*E*) Relevance distribution of the predictions of the ensemble approach using the biclusters from *D*.

mance of most algorithms can be observed (Fig. 4*B*), with only Plaid showing a high increase in relevance on the multiomics data compared to the other datasets, and not identifying any biclusters on the protein data. Only with MoSBi, a relevance and recovery higher than 0.25 could be observed in all four datasets. This shows that the multiomics data did not yield a big performance increase for most algorithms, but rather that all data types carry the information to identify subtypes, with the ensemble approach being the most robust throughout all data types.

While we did not find big differences in the overall performance, we next looked at the recovery of the subtypes individually (Fig. 4*C*). Most algorithms did not recover all subtypes equally well. Isa has the highest recovery for basal (above 0.75 for RNAseq and multiomics) and worst for normal (all below 0.25). Fabia

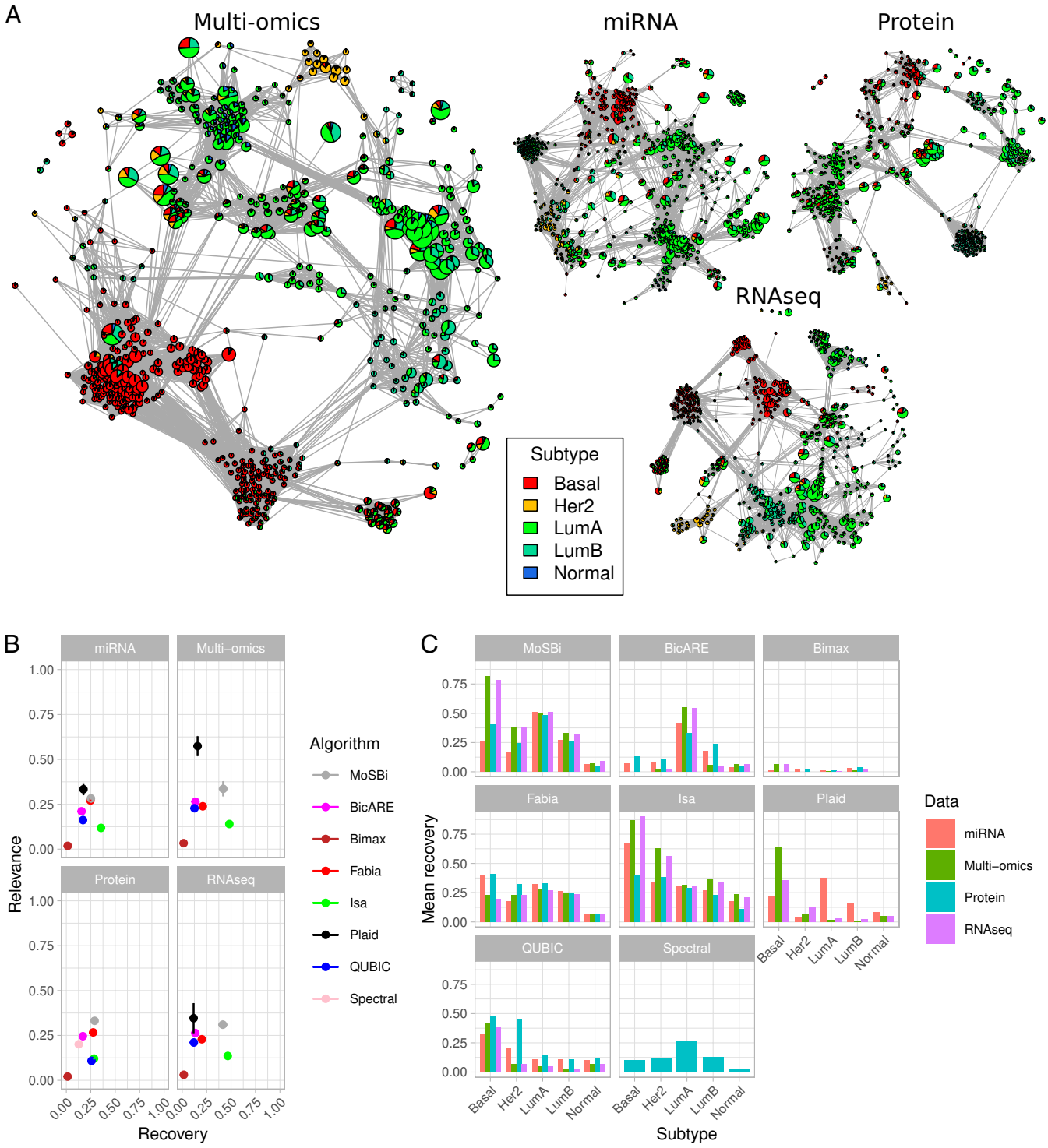

Fig. 4. Biclustering on breast cancer multiomics data. (*A*) Bicluster similarity networks on TCGA breast cancer miRNA, Protein expression, RNAseq, and combined data. Biclusters are colored by subtype, and node size is proportional to sample size. (*B*) Relevance and recovery for the condition match score on the datasets from *A* for each algorithm individually and combined with MoSBi. All algorithms were executed 10 times. (*C*) Recovery for subtypes on the datasets from *A* for each algorithm individually and combined with MoSBi. All algorithms were executed 10 times.

exhibits a more equal distribution, except for normal, which has a low recovery throughout all algorithms. It can again be observed that all data types are similarly able to identify subtypes. In MoSBi, the basal subtype has a better recovery in RNAseq and multiomics data. Another interesting observation is that BicARE consistently recovers the LumA subtype through all data types.

In this analysis, we can show that multiomics biclustering is possible and can add value to the results. However, an individual biclustering analysis on all data types is also possible and yields similar performance. However, a combined analysis might be beneficial for a biological interpretation of biclusters, which consists of features from different omics types.

The MoSBi Software Suite. To make our ensemble approach and biclustering algorithms, in general, accessible for scientists and provide an easy-to-use interface, we developed the MoSBi suite for the identification of molecular signatures using biclustering. MoSBi is available as an R package on biocon-

ductor (https://bioconductor.org/packages/mosbi/) and web-app (https://exbio.wzw.tum.de/mosbi).

Many biclustering algorithms, such as Isa (41) and Fabia (27) or the biclust package, use different result formats for returning biclusters. Therefore, we developed a unified framework, which is able to import predictions from various biclustering algorithms to simplify the analysis of biclustering algorithms and apply our ensemble approach. Our network-based visualizations are also available in MoSBi, which can be used with our ensemble approach or single biclustering algorithms. The framework can be extended to offer support for new biclustering algorithms and integrate them into the workflow. Networks can be exported as graphML for compatibility with tools such as Cytoscape (42).

The web app allows users without programming knowledge to stratify samples with our ensemble approach and profit from visualizations. Additionally, all biclustering algorithms can be accessed and executed with all parameters independently, if users are interested in specific algorithms. We also provide a docker image of the web tool, which allows it to be deployed locally.

Discussion

Stratification of patients based on molecular omics data is a challenging task and requires modern computational tools. Unsupervised approaches are suited to identify novel subgroups in the data. Biclustering is able to find meaningful patterns in modern omics data. In contrast to traditional clustering, algorithms not only output sample subgroups but, additionally, feature subsets that characterize this similarity and can be further analyzed, for example, for functional associations to find disease mechanisms. We developed a biclustering ensemble approach, which takes the results of multiple biclustering algorithms and computes ensemble biclusters using a network-based approach. This is based on the assumption that biclustering algorithms predict highly overlapping biclusters, which we could validate in our work. Various biclusters pointing to the same underlying data structures can indicate robust biclusters, which are then identified with MoSBi. We showed this on thymic epithelial tumor data (35), where we were able to retrieve known cancer subtypes.

We demonstrated the application of MoSBi on cancer-related datasets and showed the possibility of performing a multiomics analysis using biclustering. On various synthetic and experimental datasets, we assessed the performance of different biclustering algorithms and compared them to our ensemble approach. While Fabia and Isa, on average, performed best of all considered biclustering algorithms, no algorithm performed best in all scenarios and can be universally recommended. MoSBi did not always stand out, but it achieved a robust good performance in most scenarios. While the optimization of algorithm parameters on synthetic data could significantly improve the results, it leads to extensive run times and requires gold standard annotations, which are usually not available in real data, indicating that MoSBi is a preferable choice for biclustering. Additionally, it markedly reduces the number of biclusters. The network visualization gives an overview of the results and, compared to other methods (18), scales well with an increased number of biclusters.

The advantage of our ensemble approach over other biclustering ensemble approaches is that it is not algorithm specific and, via the MoSBi suite, is accessible as an application programming interface (API) and graphical user interface. Unfortunately, some proposed approaches lack implementation (10, 13). An ensemble method based on the calculation of similarities between biclusters was proposed by Hanczar and Nadif (12), where the authors calculated overlaps based on sums of overlaps of rows and columns, which can result in nonzero similarities for biclusters that share rows but no columns and are, in fact, not overlapping (*SI Appendix*, Fig. S14). They proposed the method as a single-algorithm ensemble approach that applied hierarchical clustering on the similarity matrix. This introduces another parameter for the number of consensus biclusters and assigns each bicluster to an ensemble bicluster, even with low overlap. Our approach avoids this by using the Louvain modularity to find the optimal split of the network into communities. We also introduce an error model for ensemble biclustering that removes random, and therefore misleading, overlaps from the similarity network. Additionally, MoSBi makes further analysis easier, since it reduces the number of predictions while maintaining similar performance. With MoSBi, we provide a tool to make the application of multialgorithm ensemble biclustering with scalable visualizations applicable for all kinds of noninformatics users possible. However, as an ensemble approach, MoSBi relies on the performance of multiple biclustering algorithms. We showed how the selection of biclustering algorithms can influence the results of MoSBi (*SI Appendix*, Fig. S11 *C* and *D*). While MoSBi is robust against a few badly performing algorithms, the majority of algorithms need to identify reasonable biclusters for MoSBi in order to work correctly. With new developments and available algorithms, MoSBi can be extended to improve performance in the future.

Similar to other unsupervised methods such as clustering, biclustering is often only the first step in data analysis. This comes with the challenge to inspect and interpret the results before further deciding about follow-up analysis steps. A particular challenge can be the difference in sizes of (ensemble) biclusters. It is important to consider the number of samples included for a molecular signature that corresponds to a phenotype, to evaluate its robustness. A direct comparison of biclusters with big differences in size should therefore be handled with care. The MoSBi framework allows for simple visualization of this but still requires manual supervision.

Our methodology offers an advanced perspective on biclustering and can visualize detailed properties of predictions. We demonstrated how a bicluster network analysis provides additional biological and structural insights into data. Clinical or experimental conditions can be associated with biological features. Using our approach, biclustering has the potential to play a significant role in disease subtyping and understanding.

Materials and Methods

The biclustering ensemble algorithm consists of four major steps. These are the execution of multiple biclustering algorithms, followed by a similarity computation for all returned biclusters, filtering of the similarity matrix for random overlaps, and community detection on the similarity network. In the following, all steps are described in detail.

Algorithms. Given an input matrix $M \in \mathbb{R}^{R \times C}$, we utilize different biclustering algorithms (Table 1) and collect their results in one combined list of biclusters $B = [B_1, B_2, ., B_n]$, where $B_i = (B_i^r, B_i^c)$ and $B_i^r \sqsubset [1, ., R]$, $B_i^c \sqsubset [1, ., C]$ is a set of row and column indices of the matrix M that belongs to a bicluster B_i. We implemented interfaces for all algorithms in our R package to generate this list using one unified API.

Similarity Metrics. In the next step, pairwise similarities between all biclusters in B are computed. This is done using common similarity metrics, where the similarity is expressed as a 2D overlap between biclusters. To do so, we treated a bicluster matrix as a 2D area and computed their similarity in terms of overlapping areas. This is different than the additive similarity as proposed by Hanczar

and Nadif (12). One implemented metric is the Jaccard index. Our adaption resulted in the following formula:

$$J(B_1, B_2) = \frac{|B_1 \cap B_2|}{|B_1 \cup B_2|} = \frac{|B_1^c \cap B_2^c| \times |B_1^r \cap B_2^r|}{(|B_1^r| \times |B_1^c|) + (|B_2^r| \times |B_2^c|) - (|B_1^c \cap B_2^c| \times |B_1^r \cap B_2^r|)}.$$

Besides the widely used Jaccard index, also the Bray–Curtis similarity, overlap coefficient, and Fowlkes–Mallows index were implemented in a similar 2D fashion. This results in a similarity matrix S with $S_{i,j} = J(B_i, B_j)$. Note that MoSBi can use any other definition of similarity as well. Biclusters fully contained in other ones are evaluated with the same metric. Hence, they exhibit a similarity based on their overlap as described above.

Error Model. Since biclusters can have random overlaps that do not represent meaningful interactions, we estimate a cutoff to filter for such overlaps in the similarity matrix. This is done by randomly generating a list of biclusters B' such that $B' = [B'_1, B'_2, ., B'_n]$, $|B'| = |B|$, and $B'_i = (B_i^{r\prime}, B_i^{c\prime})$, where $B_i^{r\prime}$ and $B_i^{c\prime}$ are randomly drawn without replacement from $[1, \ldots, R]$ and $[1, \ldots, C]$ correspondingly such that $|B_i^{r\prime}| = |B_i^r|$ and $|B_i^{c\prime}| = |B_i^c|$. To estimate the best cutoff c^* for the values in the similarity matrix S, we treat the S as an adjacency matrix and optimize c* for the biggest ratio between remaining edges in S and S', where $S'_{i,j} = J(B'_i, B'_j)$ (to increase robustness, multiple randomizations K of B are used),

$$c^* = \underset{c}{\operatorname{argmax}} \frac{\sum_{i,j} \Theta_c(S_{i,j})}{\sum_k (\sum_{i,j} \Theta_c(S'^k_{i,j}))/K},$$

with

$$\Theta_c : \mathbb{R} \to \{0, 1\}, \quad x \mapsto \begin{cases} 0: & x < c \\ 1: & x \geq c \end{cases}.$$

This results in the final and filtered similarity matrix S^{c*} where

$$S^{c^*}_{i,j} \mapsto \begin{cases} 0: & S_{i,j} < c^* \\ S_{i,j}: & S_{i,j} \geq c^* \end{cases}.$$

Community Detection. Finally, S^{c*} is used as an adjacency matrix with biclusters as nodes, and edges representing similarities. We compute the weighted Louvain modularity (43), with similarities as weights, to find bicluster communities in the network. These highly similar bicluster communities can then be converted into ensemble biclusters using three parameters: min_size (default = 2) which defines the minimum number of biclusters in a community to convert a community into a bicluster, where smaller communities are not considered; and row_threshold and col_threshold (default = 0.1), the minimum frequency of occurrence of a row/column element in a bicluster community to be taken over into an ensemble bicluster: For example, with values of 0.5, only genes and samples will be part of the new ensemble bicluster if they occur in at least 50% of all biclusters in the corresponding community.

Implementation. MoSBi is free software. The workflow was implemented in the R programming language (version $\geq$ 3.6) and C++17. The web interface was realized with the Shiny web framework for R (version 1.4.0.2). The workflow can be executed from our web app on our servers or on a local machine using a public Docker image. For higher throughput or for the integration of our approach into a bioinformatics pipeline, the R package can be used directly.

Visualizations. Network visualizations of the MoSBi package are implemented in R using the "igraph" package. Interactive plots in the web tool use the "visNetwork" library. All other visualizations use the "ggplot2" library in R.

Cooccurrence Networks. For cooccurrence networks, biclusters from one community were selected. From this, a new network is computed with samples and features as nodes. Edges can occur between samples and samples, samples and features, and features and features. An edge is drawn between two nodes if they occur together in at least one bicluster of the community. Edges are weighted by the number of biclusters, where two nodes cooccur. For the visualization, a network layout is computed, which takes the edge weights into account.

Match Score. The performance of biclustering algorithms and MoSBi was evaluated by comparing their overlap to labeled gold standard data. We used the commonly applied gene match score,

$$\mathrm{MS}_G(M_1, M_2) = \frac{1}{|M_1|} \sum_{(G_1, C_1) \in M_1} \max_{(G_2, C_2) \in M_2} \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|},$$

where M_1 and M_2 are two sets of biclusters, with each bicluster consisting of a set of genes G_i and conditions C_i (rows and columns) (25). To investigate sample/condition overlaps, we define the according condition match score,

$$\mathrm{MS}_C(M_1, M_2) = \frac{1}{|M_1|} \sum_{(G_1, C_1) \in M_1} \max_{(G_2, C_2) \in M_2} \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|}.$$

On synthetic data, where a 2D gold standard is available, we define the 2D match score as the multiplicative score of both dimensions,

$$\mathrm{MS}_{2D}(M_1, M_2) = \frac{1}{|M_1|} \sum_{(G_1, C_1) \in M_1} \max_{(G_2, C_2) \in M_2} \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|} \times \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|}.$$

The scores can be used to compute relevance and recovery. Let M_{opt} be a set of implanted biclusters or a gold standard, and let M be the output of a biclustering algorithm. Then, the average bicluster relevance is defined as $\mathrm{MS}(M, M_{opt})$ and describes to what extent the biclusters found by the algorithm correspond to the true hidden biclusters in the gene, condition, or both dimensions. Similarly, the average bicluster recovery is defined as $\mathrm{MS}(M_{opt}, M)$ and describes how well each of the true biclusters is recovered by the algorithm. The recovery and relevance score both have an optimal value of one, indicating a perfect overlap, and zero, indicating no overlap.

The match scores describe a normalized sum of values. To investigate how well all individual biclusters predicted by one algorithm match the gold standard, we investigated the relevance distribution $\mathrm{RD} = [rd_1, rd_2, \ldots, rd_n]$ with n as the number of biclusters in set of biclusters M and

$$rd_i = \max_{(G_{opt}, C_{opt}) \in M_{opt}} \frac{|C_i \cap C_{opt}|}{|C_i \cup C_{opt}|} \times \frac{|G_i \cap G_{opt}|}{|G_i \cup G_{opt}|},$$

where C_i and G_i are the columns and rows of bicluster M_i.

Experimental Omics Data. We evaluated the biclustering algorithms and MoSBi on six publicly available metabolomics (36, 37), proteomics (35, 38), and transcriptomics (41, 44) datasets (*SI Appendix*, Table S1). Feature-wise z scores were computed for all datasets, and, prior to that, log2 transformed [except for Ku et al. (35) and Curtis et al. (44), which already showed a normal distribution]. Transcriptomics data were filtered for genes with 80% coverage in all samples and filtered the 5,000 most variant genes, to reduce algorithm runtime. Gene set/pathway enrichment was performed using the "clusterProfiler" R package using the "enrichGO" (biological process enrichment) and "enrichKEGG" functions.

TCGA breast cancer data were downloaded from the Xena Platform (40, 45). RNAseq transcriptomics data were processed as described above, and miRNA and protein data were filtered for 80% coverage in all samples and z-score transformed. Only samples occurring in all three datasets were considered for the individual and multiomics analysis, which resulted in 484 samples with measurements for all three data types.

Synthetic Data Generation. To investigate the performance of tools in a controlled environment with a fully known gold standard, we developed a pipeline to generate synthetic datasets with implanted biclusters and additional properties such as noise and sparsity. The pipeline is shown in *SI Appendix*, Fig. S1. A detailed description of all synthetic scenarios is available in *SI Appendix*.

Data Availability. The source code is available for the R package (https://github.com/tdrose/mosbi) and for the web application (https://gitlab.lrz.de/lipitum-projects/mosbi-webapp). Both are published under the aGLPv3 license. The code and all used data for the evaluation that was performed for this work is available on figshare: https://doi.org/10.6084/m9.figshare.19096070.v1 (46).

Previously published data were used for this work (35–38, 40, 41, 44).

All other study data are included in the article and/or *SI Appendix*.

ACKNOWLEDGMENTS. T.D.R., N.K., and J.K.P. are funded by the Bavarian State Ministry of Science and the Arts in the framework of the Bavarian Research Institute for Digital Transformation (Grant LipiTUM). J.B. was partially funded by his VILLUM Young Investigator Grant 13154. The work by J.B. was also supported by the German Federal Ministry of Education and Research within the framework of the e:Med research and funding concept (Grant 01ZX1910D).

Author affiliations: [a]LipiTUM, TUM School of Life Sciences, Technical University of Munich (TUM), 65354 Freising, Germany; [b]Department for Mathematics and Computer Science, University of Southern Denmark, 5230 Odense, Denmark; and [c]Institute for Computational Systems Biology, University of Hamburg, 22607 Hamburg, Germany

1. M. R. Trusheim, E. R. Berndt, F. L. Douglas, Stratified medicine: Strategic and economic implications of combining drugs and clinical biomarkers. *Nat. Rev. Drug Discov.* **6**, 287–293 (2007).
2. S. Khakabimamaghani, M. Ester, "Bayesian biclustering for patient stratification" in *Pacific Symposium on Biocomputing 2016*, R. A. Altman *et al.*, Eds. (World Scientific, 2016), pp. 345–356.
3. O. Lazareva *et al.*, BiCoN: Network-constrained biclustering of patients and omics data. *Bioinformatics* **37**, 2398–2404 (2020).
4. S. C. Madeira, A. L. Oliveira, Biclustering algorithms for biological data analysis: A survey. *IEEE ACM Trans. Comput. Biol. Bionformatics* **1**, 24–45 (2004).
5. C. Wiwie, J. Baumbach, R. Röttger, Comparing the performance of biomedical clustering methods. *Nat. Methods* **12**, 1033–1038 (2015).
6. V. A. Padilha, J. G. Ricardo, A systematic comparative evaluation of biclustering techniques. *BMC Bioinformatics* **18**, 55 (2017).
7. B. Pontes, R. Giráldez, J. S. Aguilar-Ruiz, Biclustering on expression data: A review. *J. Biomed. Inform.* **57**, 163–180 (2015).
8. U. Neumann *et al.*, Compensation of feature selection biases accompanied with improved predictive performance for binary classification by using a novel ensemble feature selection approach. *BioData Min.* **9**, 36 (2016).
9. S. Vega-Pons, J. Ruiz-Shulcloper, A survey of clustering ensemble algorithms. *Int. J. Pattern Recognit. Artif. Intell.* **25**, 337–372 (2011).
10. B. Hanczar, M. Nadif, Ensemble methods for biclustering tasks. *Pattern Recognit.* **45**, 3938–3949 (2012).
11. G. Aggarwal, N. Gupta, "BiETopti-BiClustering ensemble using optimization techniques" in *IEEE International Conference on Data Mining*, H. Xiong, G. Karypis, B. M. Thuraisingham, D. J. Cook, X. Wu, Eds. (Institute of Electrical and Electronics Engineers, 2013), pp. 181–192.
12. B. Hanczar, M. Nadif, Using the bagging approach for biclustering of gene expression data. *Neurocomputing* **74**, 1595–1605 (2011).
13. B. Hanczar, M. Nadif, "Unsupervised consensus functions applied to ensemble biclustering"in *Proceedings of the 3rd International Conference on Pattern Recognition Applications and Methods*, M. De Marsico, A. Tabbone, A. Fred, Eds. (SciTePress, 2014), pp. 30–39.
14. G. Aggarwal, N. Gupta, "BEMI bicluster ensemble using mutual information" in *2013 12th International Conference on Machine Learning and Applications*, M. Arif Wani *et al.*, Eds. (IEEE Computer Society, 2013), pp. 321–324.
15. L. Yin, Y. Liu, Ensemble biclustering gene expression data based on the spectral clustering. *Neural Comput. Applic. Ensemble biclustering gene expression data based on the spectral clustering*, 2403–2416 (2018).
16. A. Kasim, *Applied Biclustering Methods for Big and High-Dimensional Data using R* (Chapman and Hall, 2016).
17. J. Heinrich, R. Seifert, M. Burch, D. Weiskopf, *BiCluster Viewer: A Visualization Tool for Analyzing Gene Expression Data* (Springer, 2011).
18. H. Aouabed, R. SantamaríA, M. Elloumi, VisBicluster: A Matrix-Based bicluster visualization of expression data. *J. Comput. Biol.* **27**, 1384–1396 (2020).
19. G. A. Grothaus, A. Mufti, T. M. Murali, Automatic layout and visualization of biclusters. *Algorithms Mol. Biol.* **1**, 15 (2006).
20. R. Santamaria, R. Theron, L. Quintales, BicOverlapper 2.0: Visual analysis for gene expression. *Bioinformatics* **30**, 1785–1786 (2014).
21. S. Barkow, S. Bleuler, A. Prelic, P. Zimmermann, E. Zitzler, BicAT: A biclustering analysis toolbox. *Bioinformatics* **22**, 1282–1283 (2006).
22. M. Streit, et al., Furby: Fuzzy force-directed bicluster visualization. *BMC Bioinformatics* **15**, S4 (2014).
23. R. Santamaria, R. Theron, L. Quintales, BicOverlapper: A tool for bicluster visualization. *Bioinformatics* **24**, 1212–1213 (2008).
24. P. Gestraud, I. Brito, E. Barillot, *BicARE : Biclustering Analysis and Results Exploration*, R package version 1.52.0, https://doi.org/doi:10.18129/B9.bioc.BicARE (2020).
25. A. Prelić *et al.*, A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics* **22**, 1122–1129 (2006).
26. Y. Cheng, G. M. Church, Biclustering of expression data. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **8**, 93–103 (2000).
27. S. Hochreiter, et al., FABIA: Factor analysis for bicluster acquisition. *Bioinformatics* **26**, 1520–1527 (2010).
28. S. Bergmann, J. Ihmels, N. Barkai, Iterative signature algorithm for the analysis of large-scale gene expression data. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* **67**, 031902 (2003).
29. L. Lazzeroni, A. Owen, Plaid models for gene expression data. *Stat. Sin.* **12**, 61–86 (2002).
30. Y. Zhang, et al., QUBIC: A bioconductor package for qualitative biclustering analysis of gene co-expression data. *Bioinformatics* **33**, 450–452 (2016).
31. T. M. Murali, S. Kasif, Extracting conserved gene expression motifs from gene expression data. *Pac. Symp. Biocomput.* **2003**, 77–88 (2002).
32. Y. Kluger, R. Basri, J. T. Chang, M. Gerstein, Spectral biclustering of microarray data: Coclustering genes and conditions. *Genome Res.* **13**, 703–716 (2003).
33. Z. Wang, G. Li, R. W. Robinson, X. Huang, UniBic: Sequential row-based biclustering algorithm for analysis of gene expression data. *Sci. Rep.* **6**, 23466 (2016).
34. O. Vvedenskaya *et al.*, Nonalcoholic fatty liver disease stratification by liver lipidomics. *J. Lipid Res.* **62**, 100104 (2021).
35. X. Ku *et al.*, Deciphering tissue-based proteome signatures revealed novel subtyping and prognostic markers for thymic epithelial tumors. *Mol. Oncol.* **14**, 721–741 (2020).
36. X. Tang *et al.*, A joint analysis of metabolomics and genetics of breast cancer. *Breast Cancer Res.* **16**, 415 (2014).
37. Y. Yang *et al.*, Integrated microbiome and metabolome analysis reveals a novel interplay between commensal bacteria and metabolites in colorectal cancer. *Theranostics* **9**, 4101–4114 (2019).
38. J. R. Wiśniewski *et al.*, Absolute proteome analysis of colorectal mucosa, adenoma, and cancer reveals drastic changes in fatty acid metabolism and plasma membrane transporters. *J. Proteome Res.* **14**, 4005–4018 (2015).
39. K. Eren, M. Deveci, O. Küçüktunç, Ü. V. Çatalyürek, A comparative analysis of biclustering algorithms for gene expression data. *Brief Bioinform.* **14**, 279–292 (2013).
40. M. J. Goldman *et al.*, Visualizing and interpreting cancer genomics data via the Xena platform. *Nat. Biotechnol.* **38**, 675–678 (2020).
41. A. C. Berger *et al.*; Cancer Genome Atlas Research Network, A comprehensive pan-cancer molecular study of gynecologic and breast cancers. *Cancer Cell* **33**, 690–705.e9 (2018).
42. P. Shannon *et al.*, Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Res.* **13**, 2498–2504 (2003).
43. V. D. Blondel, J. L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks. *J. Stat. Mechanics Theory Exper.* **2008**, P10008 (2008).
44. C. Curtis *et al.*; METABRIC Group, The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature* **486**, 346–352 (2012).
45. M. J. Goldman, TCGA Breast Cancer (BRCA). Xene Browser. https://xenabrowser.net/datapages/?cohort=TCGA%20Breast%20Cancer%20(BRCA)&removeHub=https%3A%2F%2Fxena.treehouse.gi.ucsc.edu%3A443. Accessed 14 December 2020.
46. T. D. Rose et al., MoSBi - Data & scripts for biclustering algorithm evaluation. Figshare. https://doi.org/10.6084/m9.figshare.19096070.v1. Deposited 31 January 2022.