MONET: Multi-omic patient module detection by omic selection

Nimrod Rappoport, Roy Safra and Ron Shamir*

The Blavatnik School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel

*To whom correspondence should be addressed. Tel: +972 3 640 5383; Fax: +972 3 640 5384;

Email: rshamir@tau.ac.il (RS)

Abstract

Recent advances in experimental biology allow creation of datasets where several genome-wide data types (called omics) are measured per sample. Integrative analysis of multi-omic datasets in general, and clustering of samples in such datasets specifically, can improve our understanding of biological processes and discover different disease subtypes. In this work we present Monet (Multi Omic clustering by Non-Exhaustive Types), which presents a unique approach to multi-omic clustering. Monet discovers modules of similar samples, such that each module is allowed to have a clustering structure for only a subset of the omics. This approach differs from most extant multi-omic clustering algorithms, which assume a common structure across all omics, and from several recent algorithms that model distinct cluster structures using Bayesian statistics. We tested Monet extensively on simulated data, on an image dataset, and on ten multi-omic cancer datasets from TCGA. Our analysis shows that Monet compares favorably with other multi-omic clustering methods. We demonstrate Monet's biological and clinical relevance by analyzing its results for Ovarian Serous Cystadenocarcinoma. We also show that Monet is robust to missing data, can cluster genes in multi-omic dataset, and reveal modules of cell types in single-cell multi-omic data. Our work shows that Monet is a valuable tool that can provide complementary results to those provided by extant algorithms for multi-omic analysis.

INTRODUCTION

Modern experimental methods can measure a myriad of genome-wide molecular parameters for a biological sample. Each type of such parameters is called "omic" and is measured by a different method. Analysis of omic data improved our understanding of biological processes and human disease, and is now used in therapeutic decisions[1]. While each experiment usually measures only one omic, several experiments can be performed on the same biological sample, resulting in multi-omic datasets. Large consortia such as TCGA and ICGC collected multi-omic data from tens of thousands of cancer tumors[2,3]. Analysis of these data can further improve our understanding of cancer biology and suggest novel treatments.

Many algorithms have been developed in recent years to analyze multi-omic data, and most prominently, to detect subtypes of cancer, a task termed multi-omic clustering[4,5]. The vast majority of multi-omic clustering algorithms assume that a *common underlying structure* exists across all omics, and use all omic datasets to reveal this structure. Among the algorithms developed under this assumption are SNF and NEMO[6,7]. However, this assumption does not always hold. For example, expression and mutation data do not seem to share the same structure. Even more closely related omics, such as expression and methylation, differ. This is demonstrated by the low agreement in clustering solutions that are produced based on different omics. Moreover, in a recent benchmark we performed, we observed that solutions based on single omics can sometimes be more clinically relevant than solutions based on multiple omics[5]. Algorithms that can cluster patients while *accounting for the disagreement between omics* are therefore required.

Several recent methods addressed the distinct structure in different omics by using Bayesian statistics and explicit modeling of the different omics and their correlations. Savage et al. performed clustering on two omics, while allowing samples to be *fused* or *unfused*[8]. A fused sample belongs to a cluster spanning both omics, while unfused samples can belong to different clusters in the two omics. PSDF extended this framework to support feature selection[9].

MDI supports more than two omics[10]. Each omic has its own clustering, but clusters in different omics match each other. The probability that a sample will belong to matching clusters in two different omics has a prior that is higher the more these two omics are similar. BCC assumes a model with a global clustering and a clustering for each omic separately, and the global clustering serves as a Bayesian prior for each omic-specific clustering[11]. Finally, clusternomics represents the global clustering as a Cartesian product of the omic-specific clusters, and can also map several such clusters into the same global cluster[12]. These methods have several limitations. They are based on Bayesian statistics, which requires explicit modeling of each omic, and is slow to optimize. All methods except PSDF require a sample to belong to a coherent cluster in each of the omics, and PSDF is limited to only two omics.

Here we present MONET (Multi Omic clustering by Non-Exhaustive Types), an algorithm for detection of patient modules for multi-omic cancer data. Monet uses ideas from Matisse[13], an algorithm to detect gene modules, and generalizes its algorithmic approach to multi-omic data. In Monet's unique approach to multi-omic clustering, the goal is to form patient modules, such that each module can use only a *subset* of the omics. Thus, Monet can detect common structure across omics when it is present, but can also disregard omics with a different structure. The solution allows outlier patients, who do not belong to any module. We show that Monet finds biologically and clinically relevant patient modules in several datasets, giving results that compare favorably to those obtained from extant multi-omic clustering methods. Furthermore, we show that Monet is useful for other biomedical tasks, as it successfully finds clusters of genes, and of cells in single-cell data.

METHODS

Overview. The input to Monet is a set of L omic matrices. Matrix l has n samples and p_l features. The output is a set of modules, where each module is a subset of the samples. Modules are disjoint, and not all samples necessarily belong to a module. Samples not belonging to a module are called *lonely*. Each module M is characterized by its samples, denoted $samples(M)$, and by a set of omics that it covers, denoted $omics(M)$. Intuitively, $samples(M)$ are similar to one another in $omics(M)$.

Monet works in two phases. It first constructs an edge-weighted graph per omic, such that nodes are samples and weights correspond to the similarity between samples in that omic. In the second phase, it detects modules by looking for heavy subgraphs common to multiple omic graphs.

Omic graphs. Monet constructs a graph G_l for each omic l separately. G_l is a full graph on n nodes. Denote by $sim_l(u,v)$ some similarity measure between samples u and v in omic l. We define a binary variable $A_l(u,v)$ to indicate whether samples (u,v) belong to the same module in omic l or not. The weight assigned to edge (u,v) in omic l, denoted by $w_l(u,v)$ is:

$$w_l(u,v) = \log\left(\frac{\Pr(sim_l(u,v) \mid A_l(u,v))}{\Pr\left(sim_l(u,v) \mid \overline{A_l(u,v)}\right)}\right)$$

The weight of a module is defined as:

$$\begin{aligned} weight(M) &= \Sigma_{l \in omics(M)} \Sigma_{u,v \in samples(M)} w_l(u,v) \\ &= \Sigma_{l \in omics(M)} \Sigma_{u,v \in samples(M)} \log\left(\frac{\Pr(sim_l(u,v) \mid A_l(u,v))}{\Pr\left(sim_l(u,v) \mid \overline{A_l(u,v)}\right)}\right) \end{aligned}$$

The weight of the module is therefore the score for a log-likelihood ratio test for whether $samples(M)$ form a module on $omics(M)$, under the simplifying assumption that modules and sample pairs are independent. A positive weight indicates that this set of samples is likely to form a module on the set of omics. Modules with high positive weight therefore correspond to likely modules under a hypothesis-testing framework.

To construct the omics graphs, any weighting scheme can be used. Here, we used two schemes. In the first we applied NEMO[7], a multi-omic clustering algorithm we recently developed, to each omic separately multiple times, each time on randomly selected 80% of the samples. We set $c_l^r(u,v)$ to 1 if samples u and v clustered together in the r'th run on omic l, and to 0 otherwise. Denote by $avg(c_l^r)$ the average value of the c_l^r matrix, and by $R(u,v)$ the set of NEMO executions in which both u and v were sampled. We set $w_l(u,v) = mean_{r \in R(u,v)}(c_l^r(u,v) - avg(c_l^r)) - C$. The constant C controls the balance between modules that cover one omic (higher C value) and modules that cover multiple omics (lower C value). Note that setting C is equivalent to placing a Bayesian prior on the probability that two samples belong to the same module. Here we used $C = 0.2$.

The second weighting scheme calculates similarity (e.g. correlation) between pairs of samples, and considers these values to originate from a Gaussian mixture model of two distributions: one distribution for samples that are *mates*, and the other for samples that are not. This modeling has theoretical justifications in certain conditions[14]. The parameters of the Gaussian mixture model are learned from a small sample of the data. $w_l(u,v)$ is set by calculating $\Pr(sim_l(u,v) \mid A_l(u,v))$ and $\Pr\left(sim_l(u,v) \mid \overline{A_l(u,v)}\right)$ from the mixture model, by assuming that 90% of sample pairs in a module are mates, while 95% of sample pairs in different modules are not (see MATISSE[13]). We used this weighting scheme only in the classification experiments.

Heavy module detection. Given all the omic graphs, Monet now detects modules with high weight by maximizing the objective function $\Sigma_M weight(M)$. There is no constraint on the number of modules, or an upper bound on module sizes, so the weighting scheme must create both positive and negative edges. The problem of detecting heavy

subgraphs in this setting is NP-hard even for the case of a single graph[13]. We therefore developed an iterative greedy heuristic for detecting heavy modules. The algorithm is initialized with a set of modules termed seeds. After seed finding, at every iteration Monet considers several possible actions, described below, that can increase the objective function. It then performs an action that provides the greatest improvement.

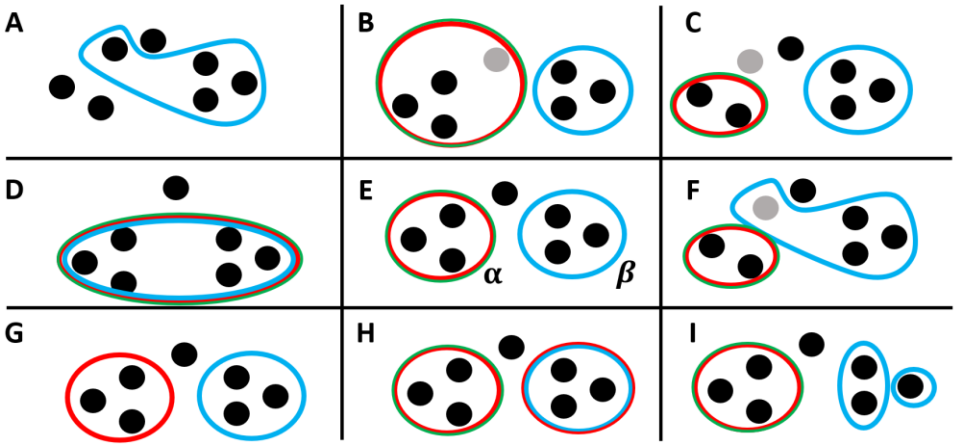

Figure 1. Actions performed by Monet when detecting heavy modules. Dots represent samples, and enclosing circles represent modules. The colors of the enclosing circle represent the omics covered by the module. Panel E shows the current state – two modules, where the left module (α) is covered by two omics and the right module (β) by one. An additional sample is lonely, i.e., does not belong to any module. Each other panel shows one action. B: the grey sample is added to module α. C: the grey sample is removed from module α. F: the grey sample moves into module β. I: module β is split. H: an omic is added to module β. G: an omic is removed from module α. D: modules α and β are merged. A: module α is discarded. In the shown case one of its samples is added to module β, and the other two become lonely. Actions for splitting module with omic or by adding omic are not shown.

▪Seed finding: Seeds are found iteratively. The first seed is determined by constructing a graph where edge weights are the sum of the edge weights in all individual omics, randomly selecting a first sample, and constructing a module containing all omics, which contains the first sample and its k neighbors with highest positive edge weights. All samples that were assigned to a module are removed from the graph, and the next seed module is sought. The procedure ends once S seeds were found. In this work we used $S = 15$ seeds for all datasets, and $k = floor(\frac{n}{15})$.

▪Optimization actions: Once a set of seeds is found, Monet improves the modules iteratively in a greedy manner. In each iteration, a module M' is selected at random, and Monet calculates the gain in the objective function from a set of possible actions concerning the module. It then chooses the action with maximal gain. It stops when no action provides a gain in any module. The actions considered are (see **Fig 1**):

- Add a sample to M'. All lonely samples are considered. Since we observed that this action is commonly chosen in initial iterations when S and k are both small, we allowed up to 10 samples to be added in a single action, to reduce the number of iterations.

- Remove a sample from M'.

- Move sample from module M' to another module, or move a sample from another module to M'. All possible samples and modules are considered. Similarly to adding samples, we allow up to 10 sample switches in a single action.

- Add an additional omic to a module. All omics are considered.

- Remove an omic from a module. All the covered omics of the module are considered.

- Merge modules M' and M''. The set of samples for the new module is $samples(M') \cup samples(M'')$. The omics for the new module are one of the following: 1. $omics(M') \cup$

$omics(M'')$ 2. $omics(M') \cap omics(M'')$ 3. $omics(M')$ 4. $omics(M'')$. All four options are considered.

- Split M' into two modules. For this action, a graph is constructed with nodes $samples(M')$, and where the weight of the edge between u and v is $\Sigma_{l \in omics(M')} w_l(u,v)$. In this graph we find a heavy subgraph M'', and create two modules, M'' and $M \backslash M''$. The omics of both modules are $omics(M')$.

- Discard M'. Each sample u in M' is moved to the module M'' with the highest sum of weights from u to M'' using $omics(M'')$. If all these sums are negative, u is made lonely.

- Create a new module using all lonely samples. Monet finds a heavy subgraph in each omic separately, and a module is created from the heaviest subgraph found.

- Split M' by adding an omic. For every omic $l \notin omics(M')$, Monet looks at the subgraph induced by $samples(M')$ on G_l, denoted $G_l[samples(M')]$, and detects in it a heavy subgraph. Denote the nodes of the heavy subgraph by U. We then split M' into two modules. In one module the nodes are U, and the omics are $omics(M') \cup \{l\}$. In the second module the nodes are $samples(M') \backslash U$ and the omics are $omics(M')$.

- Split M' with an omic. As in the previous action, a heavy subgraph with nodes U is found in $G_l[samples(M')]$, but here for every $l \in omics(M')$. Two modules are constructed. In one the nodes are $samples(M') \backslash U$ and omics are $omics(M')$. In the other samples are U and the only omic is l that produced the heavy subgraph.

Monet uses a parameter η for the minimum module size. Actions that reduce the number of samples below η are not executed, and module splits are considered under this restriction. Here we used $\eta = 10$.

To find a heavy subgraph in a graph, we use a heuristic based on Charikar's 2-approximation to the problem of maximum density subgraph[15]. We iteratively find the node with lowest (weighted) degree and remove it from the graph, until no node is left. We then choose the heaviest of the sequence of subgraphs obtained during this process. The complexity of the heuristic on an n-node weighted full graph is $O(n^2)$.

The Monet algorithm is guaranteed to converge to a local maximum, because the sum of weights within all modules is increasing in each iteration. The algorithm stops when no action on any module improves the objective.

In each iteration, all actions that do not involve finding heavy subgraphs consider each edge in each of the omic graphs a constant number of times. The complexity of all these actions is therefore $O(\Sigma_l(n + |E_l|))$, where E_l is the number of edges in G_l. The complexity of splitting a module and of creating a new module involves finding a heavy subgraph and is thus $O(\Sigma_l(n + |E_l|) + n^2)$. For the last two actions, for the same reason, the same complexity is needed for each omic considered for the split, and the overall complexity is $O(L(\Sigma_l(n + |E_l|) + n^2))$, which is therefore the overall complexity of each iteration. For full graphs, this gives a worst case complexity of $O(L^2 n^2)$. The space complexity is $O(Ln^2)$.

In a post-processing step we perform empirical significance testing to filter modules. Given a module, we sample 500 modules of the same size and omics, and only keep the module if its weight is in the highest 1%. In practice we only performed the testing for modules of minimal size ($\eta = 10$ here), as we never found larger non-significant modules. Samples that do not belong to any module after filtering are marked as lonely.

Additional Monet features.

▪Partial datasets: Monet can handle datasets where only a subset of the omics were measured for some samples. Such samples are added to all omic graphs, but in omics where these samples were not measured their nodes have no edges. This way, omics

in which no data were measured for a sample do not affect the decision of assigning the sample to a module.

■Sample classification after clustering: Once modules were calculated from the data, Monet can naturally classify new samples into modules. For each module M, Monet calculates the gain in $weight(M)$ from adding the new sample u to M: $\Sigma_{v \in samples(M)} w_l(u,v)$, and classifies the sample to the module with maximal gain. If the gain is always negative, the sample is not classified to any module. This computation takes $O(nL)$ given that the edge weights were already calculated.

Testing methodology.
We applied Monet and several other algorithms to simulated, image and cancer datasets that are described later. Here we outline the way we evaluated the results.

Clustering assessment: To assess a clustering solution where the true clustering of the data is known, we used the Adjusted Rand Index (ARI)[16]. On cancer datasets from TCGA we performed survival analysis to assess the distinction in survival between the different groups of samples. We used a permutation-based approach to perform the log-rank test, since the widely used asymptotical version of this test tends to overstate significance, and specifically for TCGA data[17–19].

Partial datasets experiments: For cancer datasets, we sampled 40% of the patients, partitioned them into three equal groups, and removed every group from one of the omics. For the image dataset we removed 20% of the samples in each omic independently. We then applied Monet to the data and calculated ARI with Monet's solution on all data. We repeated this experiment 10 times.

Classification experiments: to perform experiments on a dataset we first applied Monet to it. Denote Monet's solution by Sol_{all}. We then partitioned the samples in the dataset into 10 equal folds. For every fold i, we applied Monet to all samples except those in the fold, and denote the solution by Sol_i. We define the *stability* of the fold to be $ARI(Sol_{all}, Sol_i)$ where the ARI is computed using only samples that appear in both Sol_{all} and Sol_i. We then classified the held out samples to the modules from Sol_i, and denote the solution after classification by $\widehat{Sol_i}$. We define the *Rand Index following classification (RFC)* of the fold to be $ARI(Sol_{all}, \widehat{Sol_i})$, where the ARI is now measured across all samples. For datasets where the ground truth is known we also measured $ARI(ground\_truth, Sol_i)$, and $ARI(ground\_truth, \widehat{Sol_i})$, and term them the *pre-classification accuracy* (preCA) and *post-classification accuracy* (postCA) respectively.

Simulations: The simulations are described in the appendix.

RESULTS

Simulated datasets. We first performed two simulations to test Monet's approach to multi-omics clustering. In the first, we simulated 300 samples from five equal-size modules in two omics. Modules 1-3 cover both omics, module 4 only the first omic, and module 5 only the second omic (**SFig 1**). We added five outlier samples that do not belong to any module. Monet correctly identified the modules and their corresponding omics (**SFig 2, 3**). In another experiment, we simulated 150 samples from five modules in three omics (**SFig 4**). Module 1 is present in all omics. Modules 2-4 all cluster together in the first omic, but belong to different clusters in omics 2 and 3. The clustering structure in omic 2 is weak. When presented with only omics 1 and 2, Monet chose to treat modules 2-4 as one module that only covers the first omic (**SFig 5, 6**). When faced with omic 3 as well, Monet identified these samples as coming from different modules that cover all omics (except for one module with very weak clustering in omic 2, which does not cover that omic) (**SFig 7, 8**). These simulations highlight Monet's approach to multi-omic integration, where sample modules can cover only a

subset of the omics, based on the strength of the clustering structure in these omics. Full details on the simulations are in the appendix.

Digits dataset. We next tested Monet in a dataset where the ground truth is known. The dataset[20] contains six types of features ("omics") of 2000 images of the handwritten digits 0-9. For performance reasons, we used 400 images. See additional details in the appendix.
We applied Monet and five other methods to the data. We chose BCC, MDI and clusternomics, which model disagreement between omics. We also chose SNF and NEMO to represent general multi-omic clustering methods. SNF is widely used, and we recently showed NEMO's high performance[7]. Each method clustered the data into 10 groups. **Fig 2a** shows that Monet outperformed the other methods modeling omic disagreement, and was comparable to SNF and NEMO. When ignoring lonely samples, Monet was slightly better than SNF and NEMO. Several modules found by Monet covered only a subset of the omics, highlighting the different structure in different omics (**SFig 9**). Methods modeling omic disagreement were much slower than SNF, NEMO and Monet, which required a few seconds or minutes (**Fig 2b**).

Cancer datasets. We next executed the same six methods on real cancer datasets from TCGA, each containing three omics: mRNA expression, DNA methylation and miRNA expression. We used ten cancer types: Acute Myeloid Leukemia (AML), Breast Invasive Carcinoma (BIC), Colon Adenocarcinoma, Glioblastoma Multiforme (GBM), Kidney Renal Clear Cell Carcinoma (KRCCC), Liver Hepatocellular Carcinoma, Lung Squamous Cell Carcinoma, Skin Cutaneous Melanoma, Ovarian serous cystadenocarcinoma and Sarcoma. Dataset sizes ranged from 170 to 621 patients. Full details on the datasets are available in our recent benchmark[5]. We used differential survival between clusters as an assessment criterion for the quality of a clustering solution (see Methods).
As we can see in **Fig 2c**, Monet and NEMO had the highest number of cancer types with significantly different survival (at significance level 0.05), with 6 such types. MDI came next with 5, and the other methods had 3-4. Remarkably, in our recent benchmark, eight other multi-omic clustering methods achieved significance for at most five cancer types. The cancer types for which Monet and NEMO obtained a significant difference in survival were not identical. While both had different survival in AML, GBM, liver hepatocellular carcinoma and Sarcoma, NEMO found differential survival in BIC and melanoma, and Monet in KRCCC and ovarian cancer. These results suggest that NEMO and Monet can be used complementarily. In terms of runtime, SNF and NEMO required seconds per dataset, Monet a few minutes, and the methods that rely on Bayesian statistics were an order of magnitude slower (**Fig 2d**).

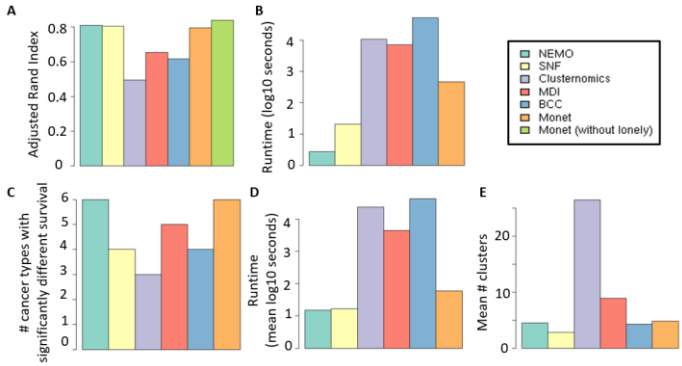

Figure 2. Performance results. A-B: Digits dataset. A: ARI of methods for multi-omic clustering. B: Run time. C-E: Results on ten TCGA cancer datasets. C: Number of cancer subtypes for which each method found a clustering with statistically different survival. D: Run time. E: Mean number of clusters found by each method.

The number of clusters chosen varied considerably among algorithms (**Fig 2e**). SNF had a mean of 2.8, NEMO, Monet and BCC 4-5, MDI 8.9 and clusternomics 26.5. The high numbers of MDI and clusternomics are possibly due to attempting to model

clustering in each individual omic. The log-rank p-value, running time and number of clusters for each method and dataset are presented in **STables 1-3**.

Monet discovered modules that use different combinations of omics (**SFig 10**). Most of the modules were based on only a single omic, and for several cancer types all modules covered only one omic. For some cancer types, this omic was the same for all modules, signifying a strong clustering structure in that omic. In none of the cancer types the solution contained only modules that covered all omics. Monet also reported several (between 0 and 12) lonely samples per cancer (**SFig 11**).

Additional analysis of the cancer results. We examined in more detail the clustering solution of Monet on the 287-patient ovarian cancer dataset. Monet found four modules in this dataset, with sizes 77, 115, 22 and 63, named M1-M4, and identified 10 samples as outliers. While SNF and MDI seek to integrate structure across all omics (**Fig 3a**), Monet chooses the omics covered by each module. In its solution all modules cover the gene expression omic, and M3 also covers miRNA expression (**Fig 3b**). To assess the clinical relevance of Monet's modules, we examined the distribution of different clinical parameters across the modules. The modules showed significant differential survival (p=0.038, **Fig 3c**), with M3 showing significantly better survival than the others (p=4e-3). The clusters showed differential survival even after correcting for age at diagnosis and clinical stage (p=2e-4 using a Cox proportional hazards model). None of the other clustering algorithms found a solution with a significant difference in survival (**Fig 3d**). The clusters were not significantly dependent of the clinical stage (0.056, chi-square test, 0.08 for Kruskal-Wallis), and they were enriched for venous invasion status (8e-4, chi-square test, **STable 4**) and for age at initial diagnosis (p=7e-3 by Kruskal-Wallis, **SFig 12**). No module was enriched for any mutation from a list of known driver mutations, or from the top 30 most frequently mutated genes in the data (see **STable 5**).

We next characterized each module in more detail using clinical parameters and GO enrichment analysis of highly expressed genes (performed with GOrilla[21]). M1 was characterized by older samples (p=4e-3, Wilcoxon test) without venous invasion (p=2e-4, chi-square), and upregulation of genes involved in microtubule-based process (e.g. TUBB2B, TUBB4A). Samples in M2 were enriched for venous invasion (p=0.02, chi-square) and high expression of immune response and extracellular matrix organization related genes (e.g. MMP9 and multiple collagen subunits). M3 had younger patients (p=0.02, Wilcoxon test). It was the only module that included the miRNA omic. We found 20 miRNAs that were highly expressed in M3's patients **(Fig 3e, STable 6**), including mir-514, which was far higher on samples in M3 compared to all other samples (**Fig 3f**). It was recently reported to regulate proliferation and cisplatin chemoresistance in ovarian cancer[22]. Finally, M4 had significantly better survival, and its highly expressed genes were also enriched for immune response. To understand the differences between M2 and M4, we found genes differentially expressed between them. M2 had higher expression of genes related to cell adhesion (e.g. collagen subunits), extracellular matrix (ECM) organization, and regulation of developmental process (e.g. WNT7A, WNT7B). Both the extracellular matrix and WNT signaling were previously reported to regulate ovarian cancer progression[23,24], and may explain the difference in venous invasion and survival between the modules. The high expression of ECM proteins may link M2 with the previously reported Mesenchymal subtype[25].

When clustering each omic separately into four clusters using spectral clustering we found a significant difference in survival for mRNA, but not for methylation or miRNA (p=0.045, 0.20 and 0.84 respectively). This demonstrates Monet's ability to select effectively clinically relevant omics. We observed this behavior for other cancer types as well. For example, Monet's solution on GBM used only methylation in all modules. Indeed, running spectral clustering and NEMO on each GBM omic separately found a solution with significant difference in survival only for the methylation dataset. Note

however that Monet's solution often uses multiple omics (see **SFig 10** for all cancer datasets and **SFigs 13-16** for the solutions on BIC and Sarcoma).

We also executed NEMO and Monet on each individual omic in the ovarian cancer data. Monet found a significant separation in survival for all omics individually, while NEMO did not find such separation for any. This shows Monet's effectiveness as a single-omic clustering approach (in this setting it is very similar to Matisse).

Monet's solution can be used to create for every sample and module a score for the linking of the sample to that module: the sum of weights between the sample and all the module's samples across all omics covered by the module. We observed that these scores could have clinical relevance. For example, for one GBM module, the linking scores of its samples were significantly associated with survival (Cox PH model, p=7.7e-3), even though the module did not have significantly different survival from other modules. We found a similar case for the Colon data, where even though modules did not have significantly different survival, the linking score for samples in one module to their module was significantly associated with survival (p=0.015).

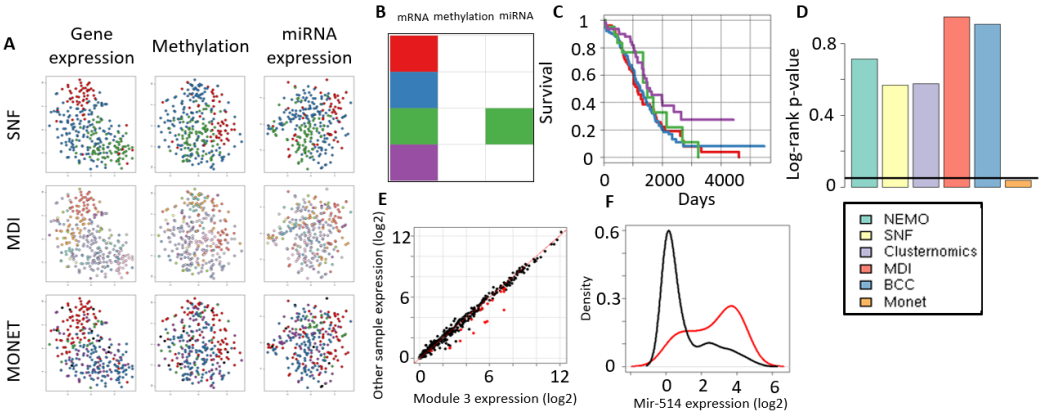

Figure 3. Analysis of Ovarian cancer. A. t-sne[26] visualization of the solutions obtained by SNF, MDI and Monet on the data. Samples are colored by their assigned module. In Monet's panels, lonely samples are black. B. Omics covered by each Monet module. Columns are omics and rows are modules. C. Kaplan-Meier plot for the different Monet modules. D. p-value of the log-rank test for the clustering solutions of different methods. E. Comparison of miRNA expression for samples in Monet's Module 3 (x axis) and other samples (y axis). Genes that are significantly highly expressed in Module 3 are colored in red. F. Distribution of mir-514 expression in samples in Module 3 (red) and in other samples (black).

Partial datasets. Often in multi-omic datasets, some samples have measurements for only a subset of the omics. Such datasets are called *partial*. Monet can address such datasets by assigning edge weight 0 to samples in the omics that were not measured. We tested this ability using the Sarcoma dataset, which had modules covering all omics, and using the digits dataset. In each dataset we randomly removed samples from some omics (see Methods), applied Monet, and compared its solution to the solution using all samples, and to the ground truth in case of the digits dataset. The results are presented in **Fig 4a** and **Fig 4b**.
Monet's output on the digits dataset was quite robust, with only a slight deterioration in performance. The Sarcoma results were less stable, but still had an ARI of about 0.5. Interestingly, samples removed from the gene expression omic had higher ARI compared to samples removed from other omics, possibly indicating that Monet's solution is less affected by that omic for the Sarcoma dataset. The ARI slightly differed for samples in the digits dataset as well depending on the omic from which they were removed (**SFig 17, 18**). These results suggest that Monet can be robustly applied to partial datasets.

Classification. Given a clustering solution, Monet's probabilistic framework allows classification of new samples into modules (see Methods). We tested Monet's

robustness and classification on the Sarcoma and digits datasets. For each dataset we performed an unsupervised version of 10-fold cross validation. We define the *stability* of a fold as the ARI between Monet's solution on all samples and Monet's solution for the current fold (which excludes 10% of the samples). We define the *Rand Index following classification* (RFC) of a fold as the ARI between Monet's solution on all samples and its solution on the fold following the classification of the 10% held out samples (see Methods). For the digits dataset, we also compared the result of every fold to the ground truth, with and without the 10% of held out samples, and term them the *pre-classification accuracy* (preCA) and *post-classification accuracy* (postCA). Note that we used here the Gaussian mixture weighting scheme, as in order to perform classification Monet calculates the edge weights for the new samples.
The results are presented in **Fig 4c** and **Fig 4d**. In all the runs the stability and RFC remained high, suggesting that the classification is highly accurate, and that decrease in performance stems largely from the different clustering structure that is obtained from sampling the datasets. Overall, these results show that Monet's framework can be used to perform classification given new samples.

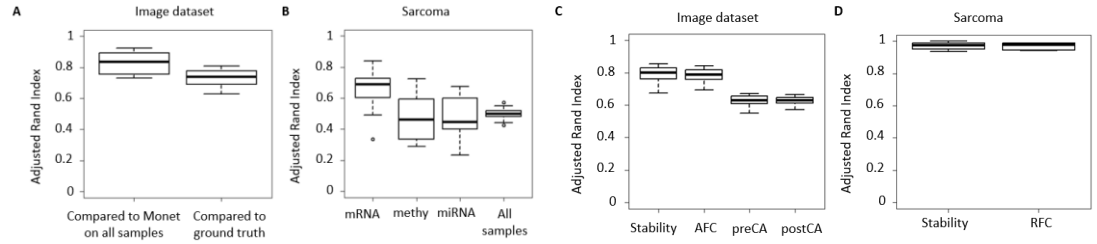

Figure 4. Performance of Monet on partial datasets and in classification. A. ARI on a partial version of the digits dataset compared to its solution on the full dataset and to the ground truth. B. ARI on a partial version of the Sarcoma dataset compared to its solution on all samples. Shown is the ARI when samples were dropped from one of the omics (three left boxplots), and from all samples together (rightmost boxplot). C. Performance in classification experiments on the digits dataset. See Methods for the assessment criteria. D. Performance in classification experiments on the Sarcoma dataset. All boxplots are distributions over 10 random runs.

Other biological tasks: gene and single cell clustering. We next tested Monet on additional biological tasks. We used Monet to cluster 1532 genes measured by both RNA-seq and microarrays of the BIC TCGA dataset that exhibited high variance in both these omics. Monet reported six main gene modules (**Fig 5a**, **SFig 19**). We used GOrilla[21] to perform enrichment analysis for these gene modules. Reassuringly, we found enrichment of biological processes that vary across breast cancer patients in several modules, including "mitotic cell cycle process", "immune system process", and "extracellular matrix organization". As expected, all gene modules covered both omics.
Finally, we applied Monet to single-cell data. Argelaguet et al. recently developed scNMT, a method that measured gene expression, DNA methylation and DNA accessibility at single cell resolution, and applied it to mouse embryos at embryonic days 4.5-7.5[27]. We applied Monet to the gene expression and promoter methylation data of 619 single cells (**Fig 5b**, **5c**). The modules obtained were highly enriched for specific cell types and embryonic days of development (**STables 7-9**). Several modules, across different cell types and stages of development, covered both omics, reflecting the widespread changes in expression and methylation during the onset of gastrulation[28,29]. Other modules used only gene expression, suggesting an overall stronger distinction between cell types at the expression level. One module covered only DNA methylation. This module comprised cells from different cell types at E7.5, again highlighting that while the transcriptional signatures of different cell types differ at that stage, the promoter methylation profile of the different germ layers is still quite similar[27]. Overall, these results demonstrate that Monet can be applied and lead to insights in diverse biological scenarios.

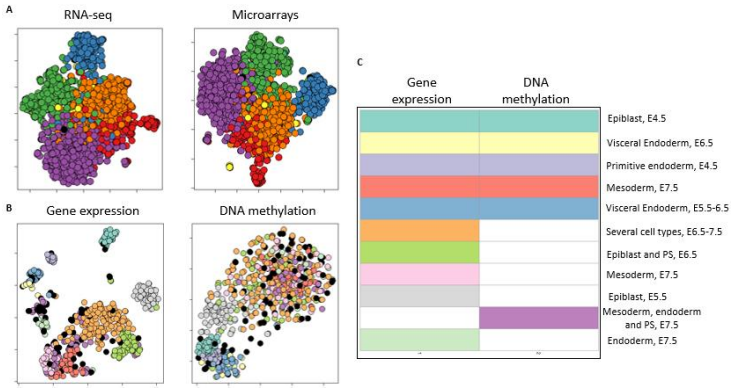

Figure 5. Using Monet to cluster genes and single cells. A. Gene clustering. t-sne visualization of Monet's gene modules on the BIC dataset. Genes are colored by Monet's output. Lonely samples are colored in black. B-C. Single cell clustering based on gene expression and DNA methylation of promoters, using the scNMT mouse embryonic development dataset. B. Like A, for Monet's solution on the dataset. C. Module omics identified by Monet. Rows represent modules and columns correspond to omics. Colored panels indicate that the module covers the omic. PS: primitive streak.

DISCUSSION

We presented Monet, a novel multi-omic clustering algorithm. Monet can identify modules with structures present in some of the omics, without imposing these structures on other omics. Monet can also identify samples that do not fit any detected module. State-of-the-art methods that seek clusters across all omics often perform quite well. We view these approaches as complementary to Monet, and suggest using both for multi-omic analysis.

The edge weighting in Monet's omic graphs can be done by schemes tailored to the omic and data, allowing flexibility in the analysis. The weighting schemes used here to cluster patients, genes, and single-cells show Monet's ability in different biomedical domains. The weighting scheme can also shift the balance between modules with single or multiple omics, or place more emphasis on one particular omic.

Most multi-omic analysis methods assume that samples are present in all omics. This is rarely the case in extant datasets, such as TCGA. It is also likely that partial datasets will be prevalent in single-cell analysis, where measuring multiple omics from a cell is just beginning and is experimentally challenging. Monet's ability to analyze partial datasets will make it valuable in this setting.

Monet has several limitations. Using different weighting schemes allows flexibility, but it can be challenging to choose one that balances finding omic-specific signals and signals reinforced by different omics. The optimization problem Monet solves is NP-hard, so the algorithm is heuristic. Adding new actions to Monet's heavy subgraph algorithm can improve its output. While Monet is faster than methods modeling disagreement between omics, it is currently slower than SNF and NEMO. Future work can improve Monet's runtime, for example by removing edges in the omic graphs. Finally, as Monet does not model the features in the dataset, understanding the molecular differences between modules requires additional analysis.

ACKNOWLEDGEMENTS

The results published here are based upon data generated by The Cancer Genome Atlas managed by the NCI and NHGRI. Information about TCGA can be found at http://cancergenome.nih.gov. The contribution of N.R. is part of Ph.D. thesis research conducted at Tel Aviv University.

FUNDING

Study supported in part by the Israel Science Foundation (grant 1339/18 and grant 3165/19 within the Israel Precision Medicine Partnership program), German-Israeli Project DFG RE 4193/1-1. NR was supported in part by a fellowship from the Edmond J. Safra Center for Bioinformatics, Tel Aviv University, and by the Planning and Budgeting Committee (PBC) fellowship for excellent PhD students in Data Sciences.

BIBLIOGRAPHY

1. Prasad, V., Fojo, T. & Brada, M. Precision oncology: origins, optimism, and potential. *Lancet. Oncol.* **17,** e81–e86 (2016).
2. McLendon, R. *et al.* Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature* **455,** 1061–1068 (2008).
3. Zhang, J. *et al.* International Cancer Genome Consortium Data Portal--a one-stop shop for cancer genomics data. *Database (Oxford).* **2011,** bar026 (2011).
4. Huang, S., Chaudhary, K. & Garmire, L. X. More Is Better: Recent Progress in Multi-Omics Data Integration Methods. *Front. Genet.* **8,** 84 (2017).
5. Rappoport, N. & Shamir, R. Multi-omic and multi-view clustering algorithms: review and cancer benchmark. *Nucleic Acids Res.* **46,** 10546–10562 (2018).
6. Wang, B. *et al.* Similarity network fusion for aggregating data types on a genomic scale. *Nat. Methods* **11,** 333–337 (2014).
7. Rappoport, N. & Shamir, R. NEMO: cancer subtyping by integration of partial multi-omic data. *Bioinformatics* **35,** 3348–3356 (2019).
8. Savage, R. S., Ghahramani, Z., Griffin, J. E., de la Cruz, B. J. & Wild, D. L. Discovering transcriptional modules by Bayesian data integration. *Bioinformatics* **26,** i158–i167 (2010).
9. Yuan, Y., Savage, R. S. & Markowetz, F. Patient-Specific Data Fusion Defines Prognostic Cancer Subtypes. *PLoS Comput. Biol.* **7,** e1002227 (2011).
10. Kirk, P., Griffin, J. E., Savage, R. S., Ghahramani, Z. & Wild, D. L. Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics* **28,** 3290–3297 (2012).
11. Lock, E. F. & Dunson, D. B. Bayesian consensus clustering. *Bioinformatics* **29,** 2610–2616 (2013).
12. Gabasova, E., Reid, J. & Wernisch, L. Clusternomics: Integrative context-dependent clustering for heterogeneous datasets. *PLOS Comput. Biol.* **13,** e1005781 (2017).
13. Ulitsky, I. & Shamir, R. Identification of functional modules using network topology and high-throughput data. *BMC Syst. Biol.* **1,** 8 (2007).
14. Sharan, R. & Shamir, R. CLICK: A clustering algorithm with applications to gene expression analysis. 307–316 (2000).
15. Charikar, M. Greedy Approximation Algorithms for Finding Dense Components in a Graph. in *Lecture Notes in Computer Science* **1913,** 84–95 (2000).
16. Hubert, L. & Arabie, P. Comparing partitions. *J. Classif.* **2,** 193–218 (1985).
17. Heinze, G., Gnant, M. & Schemper, M. Exact log-rank tests for unequal follow-up. *Biometrics* **59,** 1151–7 (2003).
18. Vandin, F., Papoutsaki, A., Raphael, B. J. & Upfal, E. Accurate Computation of Survival Statistics in Genome-Wide Studies. *PLOS Comput. Biol.* **11,** e1004071 (2015).
19. Rappoport, N. & Shamir, R. Inaccuracy of the log-rank approximation in cancer data analysis. *Mol. Syst. Biol.* **15,** (2019).
20. Van Breukelen, M., Duin, R. P. W., Tax, D. M. J. & Den Hartog, J. E. Handwritten digit recognition by combined classifiers. *Kybernetika* **34,** 381–386 (1998).
21. Eden, E., Navon, R., Steinfeld, I., Lipson, D. & Yakhini, Z. GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. *BMC Bioinformatics* **10,** 48 (2009).
22. Xiao, S., Zhang, M., Liu, C. & Wang, D. MiR-514 attenuates proliferation and increases chemoresistance by targeting ATP binding cassette subfamily in ovarian cancer. *Mol. Genet. Genomics* **293,** 1159–1167 (2018).

23. Yoshioka, S. *et al.* WNT7A regulates tumor growth and progression in ovarian cancer through the WNT/β-catenin pathway. *Mol. Cancer Res.* **10,** 469–82 (2012).
24. Cho, A., Howell, V. M. & Colvin, E. K. The Extracellular Matrix in Epithelial Ovarian Cancer - A Piece of a Puzzle. *Front. Oncol.* **5,** 245 (2015).
25. Integrated genomic analyses of ovarian carcinoma. *Nature* **474,** 609–615 (2011).
26. Maaten, L. van der & Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **9,** 2579–2605 (2008).
27. Argelaguet, R. *et al.* Multi-omics profiling of mouse gastrulation at single-cell resolution. *Nature* **576,** 487–491 (2019).
28. Mohammed, H. *et al.* Single-Cell Landscape of Transcriptional Heterogeneity and Cell Fate Decisions during Mouse Early Gastrulation. *Cell Rep.* **20,** 1215–1228 (2017).
29. Smith, Z. D. & Meissner, A. DNA methylation: Roles in mammalian development. *Nature Reviews Genetics* **14,** 204–220 (2013).