Multi-Omics Regulatory Network Inference in the Presence of Missing Data

Juan D. Henao[1], Michael Lauber[2], Manuel Azevedo[1], Anastasiia Grekova[1], Fabian Theis[1,3], Markus List[2], Christoph Ogris[1,#], and Benjamin Schubert[1,3,#,*]

[1]Helmholtz Zentrum München, Computational Health Department, Ingolstädter Landstraße 1, 85764 Munich, Germany, Member of the German Center for Lung Research (DZL); [2]Chair of Experimental Bioinformatics, TUM School of Life Sciences, Technical University of Munich, Maximus-von-Imhof-Forum 3, 85354 Freising; [3]Department of Mathematics, Technical University of Munich, 85748 Garching bei München, Germany

* To whom correspondence should be addressed. benjamin.schubert@helmholtz-muenchen.de

[#] Joint last authors

Abstract

A key problem in systems biology is the discovery of regulatory mechanisms that drive phenotypic behaviour of complex biological systems in the form of multi-level networks. Modern multi-omics profiling techniques probe these fundamental regulatory networks but are often hampered by experimental restrictions leading to missing data or partially measured omics types for subsets of individuals due to cost restrictions. In such scenarios, in which missing data is present, classical computational approaches to infer regulatory networks are limited. In recent years, approaches have been proposed to infer sparse regression models in the presence of missing information. Nevertheless, these methods have not been adopted for regulatory network inference yet.

In this study, we integrated regression-based methods that can handle missingness into KiMONo, a **K**nowledge gu**I**ded **M**ulti-**O**mics **N**etw**o**rk inference approach, and benchmarked their performance on commonly encountered missing data scenarios in single- and multi-omics studies. Overall, two-step approaches that explicitly handle missingness performed best for a wide range of random- and block-missingness scenarios on low-dimensional data, while methods implicitly handling missingness performed best on high-dimensional, multi-omics data. Our results show that robust multi-omics network inference in the presence of missing data with KiMONo is feasible and thus allows users to leverage available multi-omics data to its full extent.

Keywords:

Multi-omics integration, Network inference, Data missingness, Lasso model, Data imputation

Juan Henao is a 3rd year PhD candidate at Computational Health Center at Helmholtz Zentrum Munich working on multi-omics and clinical data integration using both, bulk and single-cell data.

Michael Lauber is a PhD Candidate at the Chair of Experimental Bioinformatics at the Technical University Munich. Currently, he is working on an approach for inference of reprogramming transcription factors for trans-differentiation.

Manuel Azevedo is a Master's student at the Technical University of Munich in Mathematics with a focus on Biomathematics and Biostatistics. Currently, he is working as a Student Assistant at Helmholtz Munich, where he is also doing his master's thesis.

Anastasiia Grekova is a Master's student of bioinformatics at the Technical University of Munich and the Ludwig-Maximilians-University Munich, working on multi-omics data integration in Marsico Lab at HMGU.

Fabian Theis is the Head of the Institute of Computational Biology and leading the group for Machine Learning at Helmholtz Zentrum München. He also holds the chair of ‘Mathematical modelling of biological systems’, Department of Mathematics, Technical University of Munich as an Associate Professor.

Markus List obtained his PhD at the University of Southern Denmark and worked as a postdoctoral fellow at the Max Planck Institute for Informatics before starting his group Big Data in BioMedicine at the Technical University of Munich.

Christoph Ogris holds a PostDoc position in the Marsico Lab at Helmholtz-Center Munich. His research focuses on predicting and exploiting multi-modal biological networks to identify disease-specific cross-omic interactions.

Benjamin Schubert obtained his PhD at the University of Tübingen and worked as a postdoctoral fellow at Harvard Medical School and Dana-Farber Cancer Institute USA before starting his group for Translational Immmunomics at the Helmholtz Center Munich.

Introduction

Complex biological systems are organised in multi-level, dynamically controlled networks that regulate and maintain the phenotypic behaviour of individual cells and their response to environmental changes [1]. Uncovering these multi-level networks and systemically understanding the interplay of their elements is a key problem in computational biology. Modern high-throughput multi-omics techniques now enable access to each regulatory network level, even at single-cell resolution [2,3]. However, combining multi-omics measurements and reconstructing the underlying regulatory network remains challenging [4]. Generally, sparse interaction networks in the form of directed or undirected graphs are constructed from dynamic interventional omics or large observational data using different classes of statistical methods [4]. Common approaches are either correlation-based [5], use techniques from information theory [6–8], or use (regularised) regression and variable selection frameworks to infer graphical models [9–11]. Most recent methods also integrate prior knowledge [12,13], such as experimentally determined protein-protein interaction networks, known metabolic pathways, or even predicted miRNA-mRNA interactions [14].

One such recent approach is KiMONo, **K**nowledge gu**i**ded **M**ulti-**O**mics **N**etw**o**rk inference [15], a two-step prior knowledge-based approach for multi-omics regulatory network inference. In the first step, the framework uses the whole dataset to model each omics element individually, detecting statistical effects between them. The framework combines all models in a second step, assembling a multi-omics graph with the input features as nodes linked via edges representing the detected effects.

However, a major drawback of most network inference methods is their inability to handle missing data. Many omics technologies such as mass-spectrometry-based proteomics and metabolomics or single-cell transcriptomics suffer from inherent missingness due to the stochasticity of the biological processes and technical limitations. Additionally, it is often necessary to combine multiple studies that only partially measure the same omics levels to reach sample sizes adequate for network inference, creating patterns of block-wise missingness. Many classical regulatory network inference methods ignore missing data and focus only on analysing complete cases, thus underutilizing the collected

data and severely limiting the amount of information used. Removing samples with missing features can also lead to biased estimates if the missingness is not completely random [16], potentially affecting the extracted regulatory network. Multiple imputation [17] is another popular approach to deal with missingness, followed by applying any classical network inference method using *ad hoc* rules to harmonise variable selection across multiply-imputed datasets [18]. However, Ganti and Willet demonstrated that such two-step approaches can be sub-optimal [19] and instead require integrated or more general frameworks to handle missing data and variable selection jointly.

In recent years, advances have been made in using sparse graphical models for data with missing information. These approaches can be roughly categorised in Bayesian methods using data augmentation strategies [20], methods using pooled posterior [21], bootstrapped inclusion probabilities [22,23], methods performing variable selection through stacked [18,24] or group Lasso integrated multiple imputation methods [25–28], low-rank matrix completion [19,29], inverse probability weighting [30], Lasso regularised inverse covariance estimation [31–34], and Expectation-Maximisation-based approaches [35,36]. While most methods address the missingness of individual features, some methods exist that also explicitly model block-missingness [28,37–39].

Incorporating such approaches in multi-omics network inference is attractive. However, a comprehensive benchmark of existing methods that can handle missing data is lacking. We, therefore, extended KiMONo with various regression-based approaches that integrate and combine prior imputed data [28] and Lasso-regularised inverse covariance estimation methods [33,34]. We systematically evaluated how these methods handle gradually increasing levels of artificial noise as and missingness for regulatory network inference on single- and multi-omics data.

We observed that approaches explicitly handling missingness in a two-step manner performed best over a wide range of random, block-missingness, and noise levels in relatively low-dimensional datasets while implicit covariance-estimation-based methods performed best in high-dimensional, multi-omics settings with block-missingness.

Methods

Knowledge gulded Multi-Omics Network (KiMONo) inference

KiMONo, **K**nowledge gu**l**ded **M**ulti-**O**mics **N**etw**o**rk inference [15], is a two-step inference procedure to detect statistical dependencies between omics features and construct a multi-omics network. Here, the inference complexity is decreased by pre-selecting feature dependencies based on existing prior knowledge of biological mechanisms such as known protein-protein interaction. Based on such prior knowledge, a system of linear multivariate regression models with sparse-group Lasso penalty is constructed [40]:

$$min_{\beta} \frac{1}{2}||y - \sum_{l=1}^{m} X^{(l)}\beta^{(l)}||_2^2 + (1-\alpha)\lambda \sum_{l=1}^{m} \sqrt{p_{(l)}}||\beta^{(l)}||_2 + \alpha\lambda||\beta||_1 \quad (1)$$

where, $\alpha = [0,1]$, and l the group index of m groups (omics layer), being $X^{(l)}, \beta^{(l)}, p_{(l)}$ the submatrix, coefficients, and length of coefficients for group l, respectively. This allows two types of sparsities: (1) “groupwise sparsity”, referring to all groups with at least one non-zero coefficient, and (2) “within group sparsity”, referring to all non-zero coefficients within every specific group (omics layer). The framework combines all models in a second step, assembling a multi-omics graph connecting regressed omics elements with its non-zero regressor elements if the regression performance (coefficient of determination) exceeds a predefined threshold.

Regression-based Methods for Network Inference and Imputation

We focused on methods with a working R implementation and consistent documentation. These requirements left us with five advanced statistical approaches of three categories (1) stacked and (2) grouped multiple-imputation, as well as (3) Lasso-based inverse covariance estimation approaches (Table 1). All mentioned methods have been integrated into the KiMONo framework (https://github.com/cellmapslab/kimono). The data underlying this article, including detailed

benchmarking results and code are available at Zenodo [41] (https://doi.org/10.5281/zenodo.7348113).

Table 1: *Inference models included in this benchmark and capable of dealing with missing data.*

Method	Category	Citation
knnKiMONo	single imputation + KiMONo	(Troyanskaya *et al.*, 2001);(Ogris *et al.*, 2021)
S(A)Lasso	stacked multiple imputation	(Du *et al.*, 2022)
G(A)Lasso	grouped multiple imputation	(Du *et al.*, 2022)
HMLasso	inverse covariance estimation	(Takada *et al.*, 2018)
CoCoLasso	inverse covariance estimation	(Datta and Zou, 2017); (Takada *et al.*, 2018)
BDCoCoLasso	inverse covariance estimation	(Escribe *et al.*, 2021)

kNN-imputation & Sparse Group Lasso (knnSGLasso): We implemented a two-step approach that first imputes missing information using nearest neighbour averaging followed by applying the Sparse Group Lasso approach of the classical KiMONo. The kNN-based imputation method [42] implemented in the R package impute v1.46.0 was applied separately to individual omics layers and other covariates. Originally designed for the imputation of gene expression data, the method replaces missing values by averaging non-missing values of its nearest neighbours. If the percentage of missing data allowed for every variable (e.g., 50% per gene (default)) is exceeded, the missing values are imputed using the overall mean per sample. Only samples with missingness less than 80% (default) were considered for the imputation. The algorithm's parameters were set to default values: the number of neighbours used in the imputation was set to k=10, and the largest block of variables imputed using the kNN algorithm before recursively dividing the feature into smaller chunks was set to max = 1500.

Stacked Adaptive LASSO (SALasso): Stacked approaches combine prior *D*-times multiply imputed datasets by averaging over them, making such approaches applicable to existing sparse regression frameworks:

$$\hat{\theta} = argmin_{\theta}\left\{-\frac{1}{n}\sum_{d=1}^{D}\sum_{i=1}^{n} o_i logL(\theta|\, Y_{d,i}, X_{d,i}) + \lambda\sum_{j=1}^{p}\hat{\alpha}_j\left|\beta_j\right|\right\} \quad (2)$$

This results in a pooled estimate $\hat{\theta}$ across all imputed datasets, thereby enforcing uniform variable selection across *p* covariates. Here α_j are so-called adaptive weights used to address estimation and selection inconsistencies that can occur and are defined as $\hat{\alpha}_j = (|\beta_j^0| + 1/n)^{-\gamma}$with some γ>0 and β_j^0 initial estimates obtained with (stacked)-Lasso or (stacked)-elastic-net. Following Du et al., γ is set to $[\frac{2v}{1} - v] + 1$ with $v = log(p)/log(n)$ [28]. As stacking multiple imputed datasets can be seen as artificially increasing the sample size thereby potentially creating oversampling biases, observation weights o_i for each subject can be included to account for the artificial inflation of the dataset with either uniform weights or with weights accounting for varying degrees of missingness per subject. We used the SALasso implementation released in the R package miselect 0.9.0 [28]. We tested 50 λ values with a lambda.min.ratio of 1e-4 in a 5-fold cross-validation with and without adaptive weights (SLasso), while the sample weights o_i were set to be uniform.

Grouped Adaptive LASSO (GALasso): Similar to SALasso, GALasso pools across prior imputed datasets by adding a group Lasso penalty term enforcing consistent variable selection across multiply imputed datasets yielding:

$$(\hat{\theta}_1, \ldots, \hat{\theta}_D) = argmin_{\theta}\left\{-\frac{1}{n}\sum_{d=1}^{D}\sum_{i=1}^{n} logL(\theta|Y_{d,i}, X_{d,i}) + \lambda\sum_{j=1}^{p}\hat{\alpha}_j\sqrt{\sum_{d=1}^{D}\beta_{d,j}^2}\right\} \quad (3)$$

with $\widehat{\alpha}_j = \left(\sqrt{\sum_{d=1}^{D} \hat{\beta}_{d,j}^2} + 1/(nD)\right)^{-\gamma}$, where $\widehat{\beta}_{d,j}$ is the estimate obtained with $\widehat{\alpha}_j$ =1, $\gamma = [2v/1 - v] + 1$, and $v = log(pD)/log(nD)$ [28]. Contrary to SALasso, individual sets of parameters θ_d are obtained for each of the *D* imputed datasets. However, the group-Lasso regularisation term jointly shrinks all $\beta_{\cdot,j}$ to zero, despite their potentially different estimates across the *D*-imputed datasets enabling a uniform variable selection. We used the GALasso implementation released in the R package miselect 0.9.0 [28]. We tested 50 λ values with a lambda.min.ratio of 1e-4 in a 5-fold cross-validation with and without adaptive weights (GLasso).

Convex Conditioned Lasso (CoCoLasso): CoCoLasso is an inverse covariance estimation method for high-dimensional data with missing values. The main idea is to reformulate the Lasso regression by working with the sample covariance matrix of X, $S = \frac{1}{n}X'X$, and the sample covariance vector of X and y, $\rho = \frac{1}{n}X'y$. With this reformulation β is estimated via S and ρ instead of X and y. Since we are dealing with missing data, CoCoLasso works with the pairwise covariance $S^{pair} := ({S_{jk}}^{pair})$ and $\rho^{pair} := ({\rho_{jk}}^{pair})$ instead of S and ρ, where

$$ {S_{jk}}^{pair} = \frac{1}{n_{jk}} \sum_{i \epsilon I_{jk}} X_{ij} X_{ik} \text{ and } {\rho_j}^{pair} := \frac{1}{n_{jj}} \sum_{i \epsilon I_{jj}} X_{ij} y_i, \quad (4) $$

where $I_{jk} := \{i: X_{ij} \text{ and } X_{ik} \text{ are observed}\}$, and n_{jk} is the number of elements of I_{jk}. Then the inference problem can be defined as:

$$ \beta = argmin_{\beta}\, \beta' \Sigma \beta - \rho^{pair'} \beta + \lambda ||\beta||_1, \quad (5) $$

$$ \Sigma = argmin_{\Sigma \geq 0}\, ||\Sigma - S^{pair}||_{max}, \quad (6) $$

Since the matrix S^{pair} needs to be positive semidefinite, CoCoLasso applies in (5) the alternating direction method of multipliers algorithm to obtain a positive semidefinite covariance matrix, Σ. Afterwards, CoCoLasso optimises the Lasso objective function (4). We used the CoCoLasso implementation released in the R package HMLasso 0.0.1 [34] with the following selection of

hyperparameters: For λ, we tested 50 values with a lambda.min.ratio of 1e-1 in a 5-fold cross-validation.

To improve the computational efficiency of CoCoLasso, BDCoCoLasso was developed by implementing a block coordinate descent strategy [43] over corrupted (with missing data) and uncorrupted (full data) covariates. However, our preliminary results showed the random missingness of 5% and more encompassed already all samples and therefore BDCoCoLasso could not be applied. We, thus, refrain from discussing the performance of BDCoCoLasso.

Lasso with High Missing rate (HMLasso): HMLasso can be seen as an optimally weighted modification of CoCoLasso according to the missing ratio. HMLasso uses the mean imputation method. Instead of X, the mean imputed data variable, Z is used, where $Z_{jk} = X_{jk}$ for an observed element and $Z_{jk} = 0$ otherwise. Followed by that the S^{pair} covariance matrix from CoCoLasso is modified by the covariance matrix of the mean imputed data, defined as $S^{imp} = RS^{pair}$, where $R_{jk} = n_{jk}/n$.

The HMLasso is then formulated as:

$$\beta = argmin_{\beta}\ \beta'\Sigma\beta - \rho^{pair'}\beta + \lambda||\beta||_1, \qquad (7)$$

$$\Sigma = argmin_{\Sigma\geq 0}\ ||W\odot(\Sigma - S^{pair})||^2_F, \qquad (8)$$

Where $W = {R_{jk}}^{\alpha}$ is defined as the weight matrix and α is the weight power parameter. We obtain the positive semi-definite matrix by minimising the weighted Frobenius norm in (8). Afterwards, HMLasso optimises the Lasso objective (7). Note that if we set $\alpha = 0$ (non-weighted case), it is equal to CoCoLasso, replacing the Frobenius norm with the max norm. We used the HMLasso implementation released in the R package HMLasso 0.0.1 [34] with the following selection of hyperparameters: For α, we tested values between 0.5 and 2 with an interval of 0.5. For λ, we tested 50 values with a lambda.min.ratio of 1e-1 in a 5-fold cross-validation.

Datasets

We collected two comparable datasets consisting of triple-omics data (transcriptomics, copy number variation (CNV), and methylation) from the PanCancer Projects [44] using The Cancer Genome Atlas (TCGA) data portal and the cBioPortal [45]. Both datasets were already pre-processed including normalised and log-transformed gene expression, beta-values for methylation, and linear CNVs.

The first dataset encompassed 871 patients with breast invasive carcinoma (retrieved on 03/07/2022), while the second dataset comprised 414 patients with muscle-invasive bladder cancer (retrieved on 23/08/2022). All samples containing missing information were removed to construct a complete data set as the baseline, thus restricting the data sets to 604 and 404 samples, respectively. Similarly, features with low variance were removed, resulting in 11,530 transcriptomics, 1,366 methylation, and 84 CNV features for the first dataset (relatively low-dimensional), and 20,085 transcriptomics, 15,460 methylation, and 24,765 CNV features for the second dataset (relatively high-dimensional), respectively. These two datasets allowed us to study the impact of data dimensionality on performance.

Prior network generation

For the first dataset, we used the prior network of the original KiMONo publication [15]. Briefly, protein-protein interactions were extracted from the BioGrid interactome (Release 3.5.188) [46], associated with the extracted gene expression information, and linked each CNV and methylation site to its associated gene since both omics layers were already annotated to gene identifiers. The final prior network contained 11,645 nodes (10,848 genes, 84 CNVs, and 713 methylation sites).

The construction of the prior network for the second dataset followed the same steps as the first prior network construction using FunCoup v5 [47] (built-in September 2020) as the basis. We used interactions with maximum Final Bayesian Score (FBS) supporting protein-protein interactions, and log-likelihood ratio for physical protein-protein interaction evidence $\geq$ 99 percentile to extract reliable

protein-protein interactions. The final prior network contained 51,318 nodes (18,990 genes, 18,460 CNVs, and 13,868 methylation sites).

Network-based multiple imputation

Stacked and grouped adaptive Lasso approaches require multiple imputed data as input. However, multiple imputation methods do not scale well to high-dimensional data with high missingness, and standard implementations such as those offered in the R package MICE, take multiple hours to days to finish. Thus, we developed a novel network-guided multiple imputation by chained equation approach (ngMICE) by utilising KiMONo's prior network. Instead of considering all covariates for imputation, we restrict each imputation attempt to the covariates that are directly linked to the missing covariate in a prior network as other covariates will be removed during network inference by KiMONo and therefore can be neglected. The number of covariates can be further reduced by correlation-based filtering. For missing elements retaining less than *k* covariates for imputation, the top *k* correlated covariates are used. Once the covariate matrix has been constructed as described, the standard MICE procedure is run. We used the R package MICE 3.14.0 [48], with Bayesian linear regression as a multiple imputation approach, an absolute Pearson correlation coefficient of 0.1 as the threshold, and k = 5 most correlated features in the final regression formula. ngMICE performed similarly to kNN-based imputation in terms of RMSE across omics types and missingness with slightly worse average performance (Supplementary Figure 1).

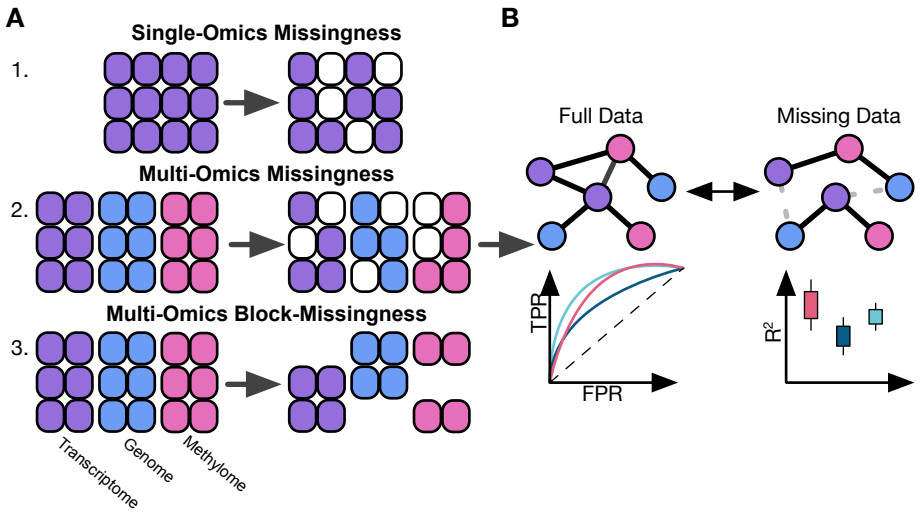

Figure 1: Benchmark schematic. A) *Three common missingness scenarios in regulatory network inferences are tested: 1) single-omics, 2) multi-omics random missingness of individual elements, and 3) block-wise missingness in which entire omics layers are missing for an individual.* ***B)*** *We tested five approaches of two categories: 1) Two-step approaches that first impute and then aggregate imputation through, and 2) Inverse covariance estimation approaches that implicitly handle missingness during inference. We inferred regulatory networks from full data and data missing for each method and compared the resulting networks with multiple performance metrics.*

Benchmark

We assessed the performance of the selected inference methods in the presence of missing data by simulating three typical scenarios - (1) random missing information in a single omics level and across (2) multiple levels, as well as (3) block-missingness structures (Figure 1a). To test the methods' capabilities even further, we decreased the signal-to-noise ratio by systematically adding covariate-specific white noise to the input data. Each experiment was repeated 5 times for robust performance estimation and corrected for confounding age and sex effects.

Single-Omics missing: We selected the transcriptomics level as a single-omics layer to test the different models' capabilities to infer gene regulatory networks with less directly informative co-

correlation structures that could be used to impute the missing gene expression information. We removed $m \in \{0\%, 10\%, 20\%, 30\%, 40\%, 50\%\}$ randomly selected entries from the input data. Additionally, we added noise with increasing intensity to the data by drawing from a normal distribution $\epsilon \sim N(0, a\sigma_g^2)$ per gene with a gene-specific variance term estimated from the real data and $a \in \{0, 0.5, 1.5\}$.

Multi-Omics missing: To test the models' capabilities to handle more complex co-correlation structures that could potentially be exploited for better imputation, we expanded the single-omics experiment to jointly consider the three available omics types. As before, we randomly removed $m \in \{0\%, 10\%, 20\%, 30\%, 40\%, 50\%\}$ of entries independently per omics layer and added feature-dependent noise to the data as described before while ensuring to bound the beta values of the methylation data to the range between 0 and 1.

Multi-Omics Block-Missing: To test the capabilities of the method to handle block-wise missing information, we removed $m \in \{0\%, 10\%, 20\%, 30\%, 40\%, 50\%\}$ samples per omics layer such that at least two omics-layers still remained per sample. Additionally, we added noise to the remaining samples as described before.

Downsampling: Similarly, to identify the minimal number of samples required to infer reliably the regulatory network, we downsampled the dataset to $k \in \{90\%, 80\%, 70\%, 60\%, 50\%\}$ of samples.

Runtime: Using the low-dimensional dataset, we tested the runtime of each method on a dedicated machine with an AMD EPYC 7502P 32-Core Processor with 2.5GHz base clock speed and 860 GB RAM using the multi-omics missingness experiment with the same configurations as before. We ran all experiments with ncores=60 (with hyperthreading enabled).

2.4 Evaluation Metrics

To construct the final network from the individual regressions, we applied a strict filter connecting independent to dependent variables if their beta coefficient was non-zero and their $R^2 > 0.1$.

Prediction Metrics: To measure the prediction qualities of each regression model, we recorded the root means squared error (RMSE) and R^2 respectively, and compared their distributions based on a Wilcoxon rank-sum test.

Network Reconstruction Metrics: To measure the methods' abilities to handle missing data well, we compared the inferred regulatory networks from missing data to their counterpart inferred on full data, and calculated precision, recall, and F1-scores of the recovered network edges.

Similarly, we inferred a ground truth network with KiMONo using stability selection by repeating the network inference 100 times fitting different random seeds and averaging over the resulting coefficients and R^2 values before constructing the network, improving the robustness of the graph resulting in a final network consisting of 2,458 nodes (2,182 genes, 63 CNVs, 211 methylations) and 6,554 edges for the low-dimensional dataset and 33,652 nodes (12,738 genes, 11,052 CNVs, and 9,862 methylations) and 87,229 edges for the high-dimensional dataset. We compared each stability-selected reference network to all inferred networks on missing data to determine performance differences across the individual methods.

Topological Metrics: The interpretation of complex heterogeneous networks and identification of key modules and important network nodes relies on the topological network features. Hence, it is also vital to evaluate if the methods can robustly infer topological structures. Therefore, we use multiple network-based metrics such as node-degree distribution, betweenness centrality, and clustering coefficient to quantify and compare the topological changes of the networks inferred from missing data. Node degree indicates the sparseness of the network, while betweenness centrality indicates

how interconnected the network is, and the global clustering coefficient (transitivity) indicates how densely connected neighbouring nodes are.

Results

Most topological features can be conserved in data with missingness

To investigate to what extend topologies of inferred networks are affected by noise and missingness, we computed multiple network properties across all benchmarking scenarios for all methods.

In single-omics scenarios, knnSGLasso demonstrated the most stable topology with increasing missingness rate on the lower-dimensional data (Figure 2A&B, Supplementary Figures 2&3), decomposing into small subnetwork at high missingness rates according to the increase in transitivity (0.054±0.007, missingness = 0.1; 0.067±0.003, missingness = 0.5) and reduction in betweenness centrality (0.005±0.007, missingness = 0.1; 0.002±0.004, missingness = 0.5; Figure 2A&B). All methods were topologically more stable on high-dimensional data except for CoCoLasso and HMLasso whose transitivity and betweenness centrality dropped at high missingness (transitivity: 0.077±0.002, betweenness: 0.003±0.002, missingness = 0.4; transitivity: 0.059±0.004, betweenness: 0.002±0.002, missingness = 0.5 for CoCoLasso; Figure 2A&C, Supplementary Figure 4).

In multi-omics scenarios, knnSGLasso had the most stable topology across increasing missingness rates on low-dimensional. However, in presence of the highest missingness rate, knnSGLasso suffered an abrupt decay in betweenness centrality (Figures 3A&B). In contrast, knnSGLasso networks became sparser as indicated by the increase in the betweenness centrality and the slight reduction of transitivity with increasing missingness on high-dimensional data (Figure 3A&C, Supplementary Figure 3). S(A)Lasso and G(A)Lasso inferred networks on high-dimensional data decreased their betweenness and transitivity abruptly compared to networks inferred on low-dimensional data (Figures 3B&C; Supplementary Figure 3). CoCoLasso and HMLasso slightly decreased their betweenness centrality for no to medium missingness rates (CoCoLasso and

HMLasso: 0.002±0.001, for missingness = 0 / 0.3), and droped in the presence of high missingness (CoCoLasso and HMLasso: 0.001±0.001, missingness = 0.5; Figures 3C).

knnSGLasso and SALasso presented the most stable topologies in presence of block-missingness on the low-dimensional dataset (Figure 4B), although knnSGLasso's full-data topology varied considerably compared to the counterpart networks inferred in presence of missing data (betweenness: 0.006±0.007 and 0.005±0.007, transitivity: 0.022±0.0 and 0.050±0.006, for missingness=0 and missingness=0.5, respectively; Figures 4A&B; Supplementary Figure 3). S(A)Lasso and G(A)Lasso were less stable, declining slightly in betweenness centrality with increasing missingness rate (Figure 4B). knnSGLasso behaved similarly on high-dimensional data, the betweenness centrality (0.001±0.001) was drastically different than the rest of its own inferred topologies in presence of missing data (0.004±0.003 for missingness = 0.1). The betweenness centrality for the approaches dealing with multiple imputed data dropped abruptly near to zero even at 10% of missingness (Figure 4C). CoCoLasso and HMLasso had the most stable topologies including the full-data inferred network. However, their betweenness and transitivity were lower than knnSGLasso (CoCoLasso and HMLasso: 0.002±0.001 and 0.096±5e-4, knnSGLasso: 0.003±0.003 and 0.099±0.001, respectively for missingness=0.5; Figure 4A&C; Supplementary Figure 3).

Across all benchmark settings, the methods showed a decreasing network size with increasing missingness. The constant loss of edges showed a stable ratio indicating no bias towards a specific omics layer (Supplementary Figure 4). GLasso was less affected in terms of noise, while knnSGLasso produced the most robust results in terms of missingness (Supplementary Figure 2).

kNN imputation-based model performs best for single- and multi-omics data with random missingness

In both the single- and multi-omics setting, knnSGLasso was the best performing method independently of the dimensionality of the dataset, reaching F1-scores of 0.921±0.005 and

0.912±0.001 on the single omics and 0.923±0.002 and 0.947±0.001 on the multi-omics data at 10% missingness (low- and high-dimensional data, respectively). Followed by SLasso (single-omics: 0.880±0.005 and 0.870±0.002) for single-omics, and CoCoLasso (0.919±0.001) for multi-omics when compared to the reference networks inferred on full data (Figures 2A & 3A; Supplementary Figure 5 & 6).

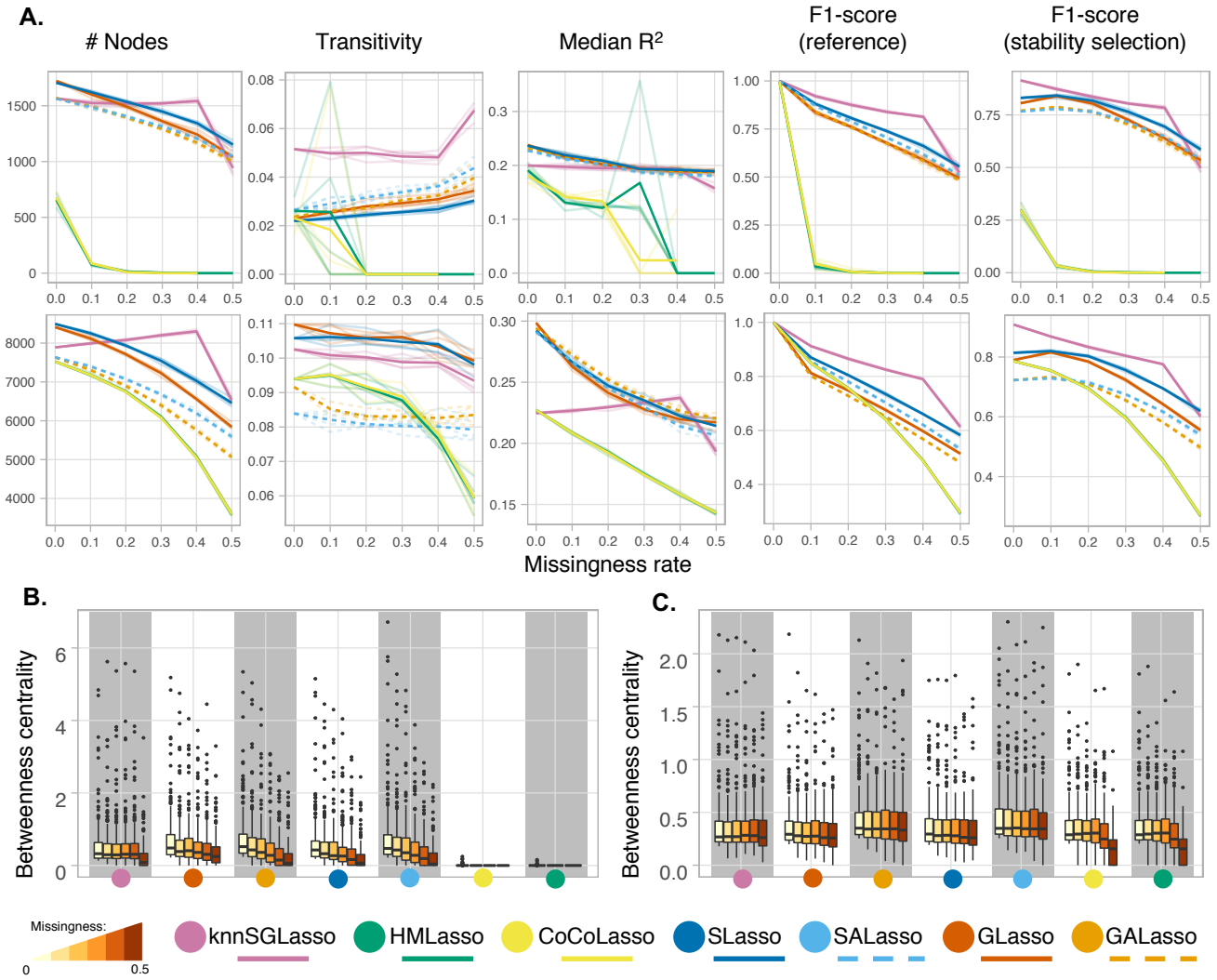

Figure2: Benchmark results across single-omics experimental setups for both datasets. *Transparent lines denote individual runs, while bold lines refer to the average performance.* ***A)*** *low-dimensional dataset (top), high-dimensional dataset (bottom). Performance was evaluated using network size (number of nodes), transitivity (global clustering coefficient), median R^2, and F1-scores compared to a reference (i.e., the same method applied on the full data and stability selected networks*

generated with KiMONo). ***B)*** *and* ***C)*** *Illustrating the topological change in low* ***B)*** *and high* ***C)*** *dimensional datasets scaling values for the top 200 nodes with the highest betweenness centrality through increased missingness.*

The performance gradually decreased with increasing missingness and noise levels to a similar degree for all the methods except for the specific case of multiple imputation-based methods in the multi-omics experiment on high-dimensional data. In this particular case, the F1-score increased to 0.485±0.003 for SLasso, 0.442±0.004 for SALasso, 0.405±0.005 for GLasso, and 0.398±0.004 for GALasso, at 50% missingness and a medium noise (a=0.5; Supplementary Figure 6). At higher noise levels, the performance of all methods declined dramatically reaching only F1-scores below 0.15.

For the low-dimensional dataset, HMLasso and CoCoLasso came in last, reaching F1-scores below 0.1 both on single- and multi-omics data even for samples with only 10% missingness and no noise (Figure 2A; Supplementary Figure 6). Both methods even failed to infer networks in 11/15 and 11/15 single-cell experiments at 30% and 40% missingness, as well as in 9/11 and 8/11 multi-omics experiments, at 40% and 50% missingness, respectively.

Similar behaviour could be observed when comparing the inferred networks to the stability selection-based reference network: knnSGLasso performed best with an F1-score of 0.873±0.005 and 0.869±0.008 on single-omics, and 0.820±0.004 and 0.858±0.007 on multi-omics at 10% missingness and no noise in the low- and high-dimensional dataset, respectively, followed by SLasso (low dimensionality: 0.841±0.007, high dimensionality: 0.820±0.003), for single-omics, and SLasso (low dimensionality: 0.772±0.003) and CoCoLasso (high dimensionality: 0.764±0.001) for multi-omics random missingness experiment (Figure 2A & 3A; Supplementary Figure 7). SLasso did marginally outperform knnSGLasso in the low-dimensional dataset in scenarios with high noise and/or high missingness and was almost equivalent at high noise and high missingness (50% missingness, a=0.5) in the multi-omics scenario on the high-dimensional dataset. At the highest noise levels

(a=1.5), even in the absence of missingness, none of the methods exceeded F1-scores of 0.18 (Supplementary Figure 7).

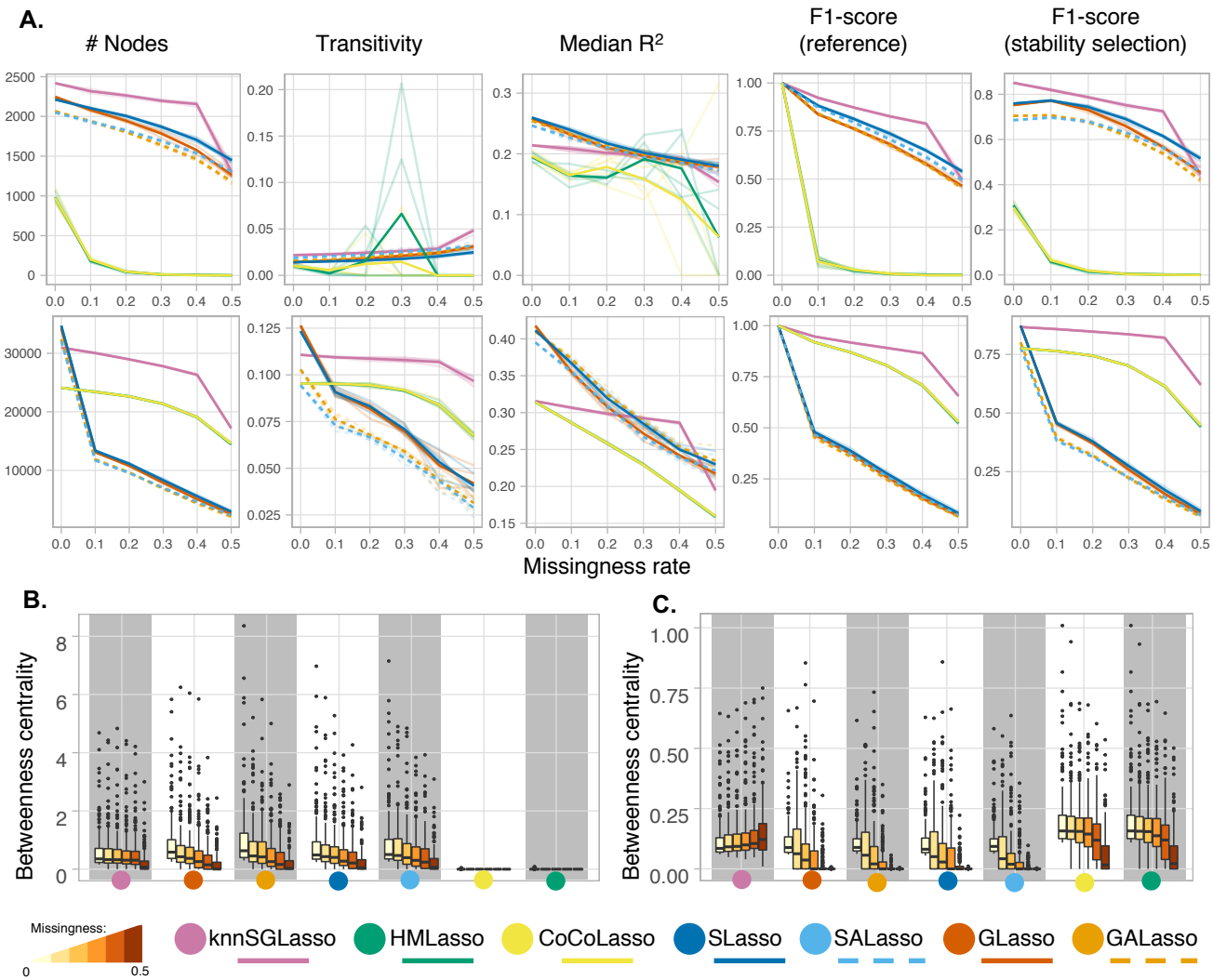

Figure 3: Benchmark results across multi-omics experimental setups for both datasets. *Transparent lines denote individual runs, while bold lines refer to the average performance.* ***A)*** *low dimensional dataset (top), high dimensional dataset (bottom). Performance was evaluated using network size (number of nodes), transitivity (global clustering coefficient), median R^2, and F1-scores compared to a reference (i.e., the same method applied on the full data and stability selected networks generated with KiMONo).* ***B)*** *and* ***C)*** *Illustrating the topological change in low* ***B)*** *and high* ***C)*** *dimensional datasets scaling values for the top 200 nodes with the highest betweenness centrality through increased missingness.*

Adaptive weights had a marginal impact on performance with a slight negative effect, irrespective of data dimensionality. Only at high noise levels did adaptive weights stabilise performance (Supplementary Figures 6 & 7).

SLasso and inverse covariance-based approaches perform best for data with block-missingness

When investigating block-missingness, i.e., where entire omics layers are missing for a subset of samples, SLasso performed best when low-dimensional data were used, while CoCoLasso and HMLasso performed best on high-dimensional data (Figure 4; Supplementary Figure 2). SLasso, CoCoLasso, and HMLasso reached high consistency with the networks inferred on full data with SLasso achieving F1-scores between 0.945$\pm$0.004 at 10% missingness (a=0) and 0.688$\pm$0.002 at 50% missingness (a=0.5) on low-dimensional data. CoCoLasso and HMLasso reached F1-scores of 0.967$\pm$0.002 at 10% missingness (a=0), and 0.810$\pm$0.004 at 50% missingness (a=0.5) respectively on high-dimensional input data (Figure 4A; Supplementary Figure 5 & 6).

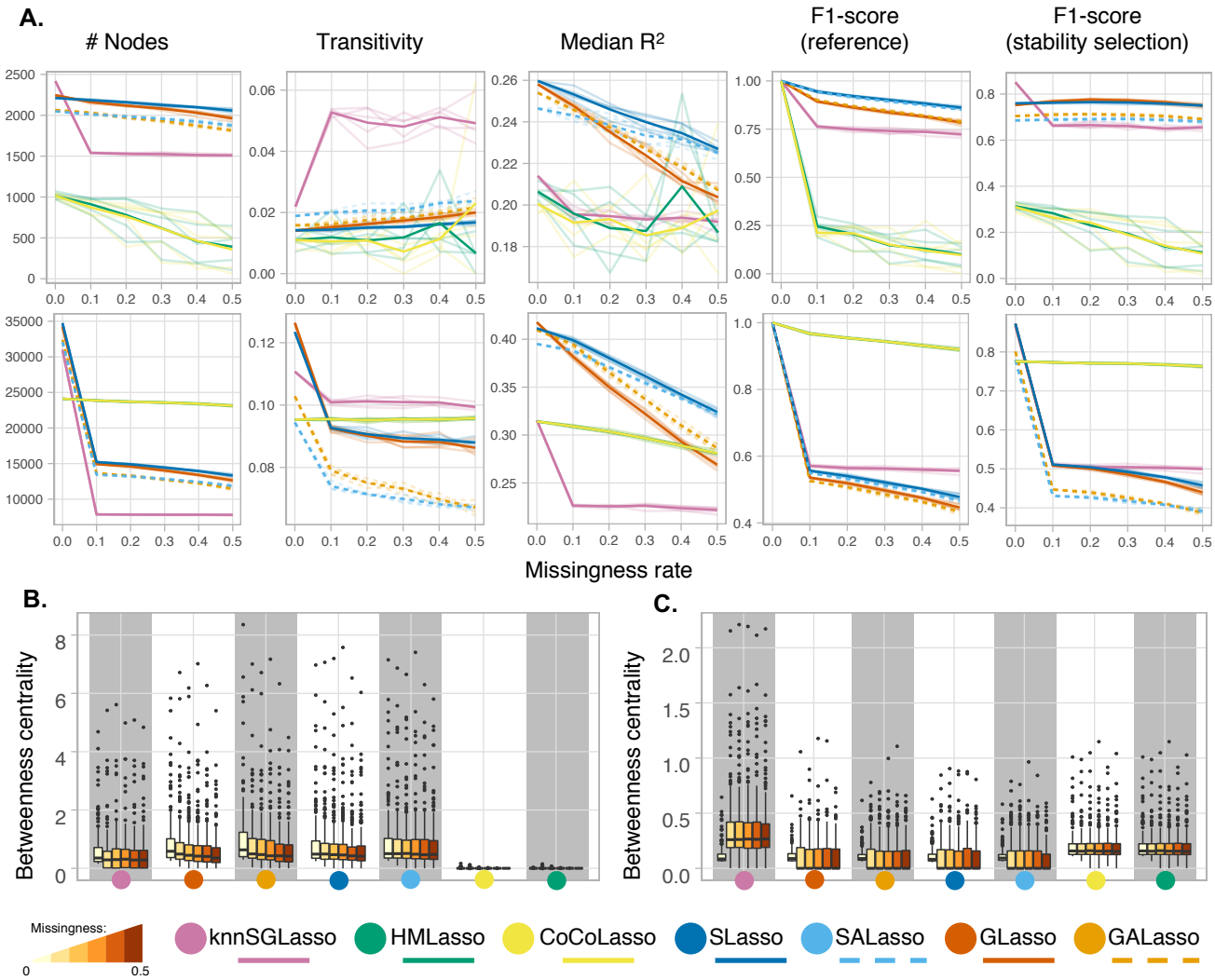

Figure 4: Benchmark results across block-missingness experimental setups for both datasets. *Transparent lines denote individual runs, while bold lines refer to the average performance.* ***A)*** *low dimensional dataset (top), high dimensional dataset (bottom). Performance was evaluated using network size (number of nodes), transitivity (global clustering coefficient), median R^2, and F1-scores compared to a reference (i.e., the same method applied on the full data and stability selected networks generated with KiMONo).* ***B)*** *and* ***C)*** *Illustrating the topological change in low* ***B)*** *and high* ***C)*** *dimensional datasets scaling values for the top 200 nodes with the highest betweenness centrality through increased missingness.*

With higher noise, the performance dropped below an F1-score of 0.2 (Supplementary Figure 6). This behaviour could also be observed for GALasso and knnSGLasso, although with generally lower F1-scores. Notably, knnSGLasso networks almost exclusively consisted of gene nodes, while all other methods had a proportional representation of all omics types. The implementation of kNN-imputation used here is not able to handle entire block-missing samples and, consequently, knnSGLasso removes a substantial amount of the features.

Similar behaviour could be observed when comparing the inferred networks on missing data to the stability-selection-based reference. Both SLasso and GLasso outperformed all other methods on low-dimensional data with SLasso reaching the highest F1-scores of 0.763±0.003 (10% missingness, a=0) to 0.655±0.007 (50% missingness, a=0.5), while CoCoLasso and HMLasso outperformed all methods on high-dimensional data reaching F1-scores of 0.773±0.001 (10% missingness, a=0) to 0.728±0.003 (50% missingness, a=0.5; Figure 4A). Adaptive weights (SALasso and GALasso) again had negligible effects and only improved performance markedly at high noise levels (a=1.5; Supplementary Figure 7).

HMLasso and CoCoLasso performed worst on low-dimensional data, reaching average F1-scores of 0.212±0.022, and 0.246±0.039 at 10% block-missingness, dropping to 0.063±0.033 and 0.073±0.031 respectively with 50% block-missingness and medium noise (a=0.5; Supplementary Figure 6). Comparing those methods to the stability-selection-based reference depicted a similar picture (Supplementary Figure 7).

knnSGLasso is the least affected by sample size reduction

All methods showed a decrease in concordance already at 10 % sample reduction (Figure 5). SLasso and knnSGLasso were the most stable method at single-omics level over progressive sample reduction (Figure 5). The F1-score of knnSGLasso was similar to SLasso on low-dimensional data at 10% sample reduction (knnSGLasso: 0.906±0.018, SLasso: 0.906±0.003). However, at a high

sample removal rate (50% sample reduction), SLasso (0.810±0.012) slightly outperformed knnSGLasso (0.796±0.020; Figure 5A). CoCoLasso and HMLasso reached similar performance as knnSGLasso on high-dimensional data when 10% sample reduction (CoCoLasso and HMLasso: 0.916±0.005, knnSGLasso: 0.917±0.002), but were consistently outperformed by knnSGLasso at higher sample reduction rates (CoCoLasso and HMLasso: 0.782±0.0059, knnSGLasso: 0.806±0.013; Figure 5A). Similar behaviour but more prominently could be observed when comparing performance based on the stability-selection reference network irrespective of the dimensionality of the data. knnSGLasso outperformed the rest of the methods followed by SLasso and GLasso. CoCoLasso and HMLasso were among the worst-performing methods (Figure 5A & B).

On multi-omics data, the results showed a similar pattern: knnSGLasso consistently outperformed all other methods on low-dimensional data (Figure 5B). On high-dimensional data, CoCoLasso and HMLasso outperformed knnSGLasso at 10% sample reduction (CoCoLasso and HMLasso: 0.959±0.003, knnSGLasso: 0.958±0.001), but were overtaken by knnSGLasso with higher reduction rates (CoCoLasso and HMLasso: 0.886±0.005, knnSGLasso: 0.893±0.006; Figure 5B). knnSGLasso strongly outperformed all other methods on low-dimensional data when compared to the stability-selection-based reference network (Figure 5B). On high-dimensional data, however, SLasso performed best, closely followed by GLasso. CoCoLasso and HMLasso performed worst for no to medium missingness rate, narrowly overtaking SALasso in presence of high missingness rate (CoCoLasso and HMLasso: 0.774±0.001 and 0.756±0.004; SALasso 0.776±0.001 and 0.752±0.004, for a=0.1 and a=0.5, respectively; Figure 5B).

HMLasso is the fastest approach in datasets with no to medium missingness

knnSGLasso, HMLasso, and CoCoLasso had a similar runtime of 82 - 85 sec on the low-dimensional dataset without missing information and gradually increased in runtime with rising missingness (Figure 5C). HMLasso was the fastest approach with an average runtime of 81.843±0.375 sec at low to

medium missingness levels. Only in scenarios with high missingness, HMLasso was outperformed by knnSGLasso, reaching an average runtime of 97.089±1.327 sec. knnSGLasso demonstrated a very consistent runtime across all missingness levels with an average runtime of 97.725±7.995 sec. CoCoLasso behaved similarly but was affected by the degree of missingness, reaching maximum average runtimes of 449.6±225.2 sec. GALasso and SALasso were the slowest, with average runtimes of 739±15.532 sec and 919.785±11.833 sec on the complete dataset, respectively, of which 241.191±2.975 sec were dedicated to imputation. Overall runtime gradually increased for both methods reaching an overall average runtime of 1148.871±245.077 sec and 1184.688±157.794, respectively, of which on average 507.674±58.126 sec was spent in imputation. For GALasso, adaptive weights calculations added another 275.363±54.180 sec on average to the overall runtime.

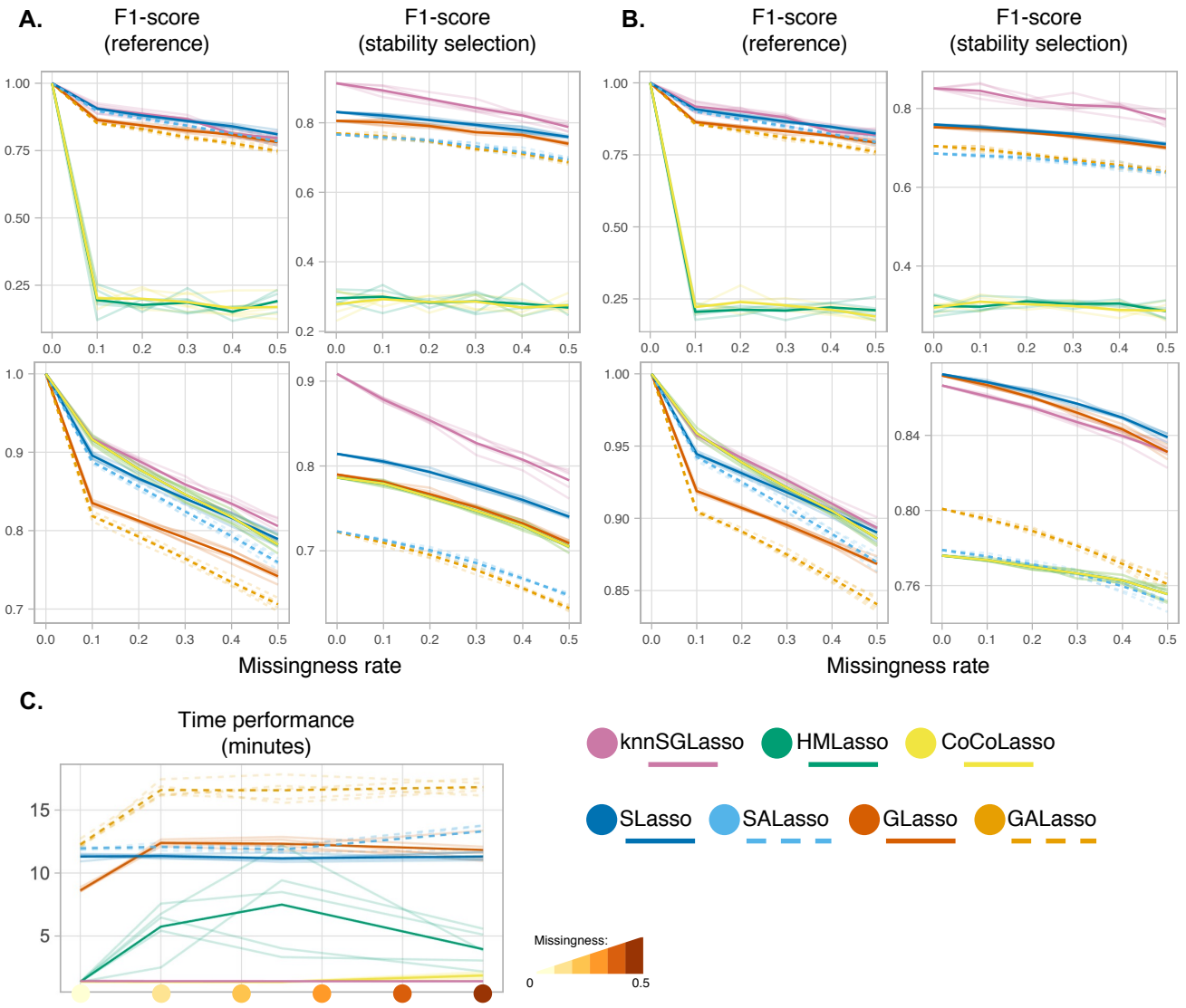

Figure 5. Sample size results across all experimental setups. *Transparent lines denote individual runs, while bold lines refer to the average performance. F1-scores for single-omics* ***A)*** *and multi-omics* ***B)*** *on low-dimensional data (top) and high-dimensional data (bottom).* ***C)*** *Runtime evaluation over multi-omics experiments using low-dimensional data.*

Discussion

Due to economic or technical restrictions, missingness of individual values or block-missingness of entire omics layers in a subset of samples is typical for high-throughput multi-omics experiments, rendering multi-omics network inference challenging. In this study, we benchmarked novel regression

approaches that can handle missing information across common missingness scenarios in single- and multi-omics experiments and integrated these approaches into KiMONo, a recent approach for network-guided multi-omics network inference.

We observed that the performance of the different methods was dependent on the dimensionality of the input data. However, some general trends could be identified: kNN-imputation combined with the standard KiMONo approach performed best in most cases except when block-missingness was present. For this particular case, the dimensionality of the dataset was a crucial factor, with SLasso and GLasso being suitable for low-dimensional data, while CoCoLasso and HMLasso were better suited for high-dimensional data. Both types of methods were also able to handle high degrees of block-missingness consistently well on low- and high-dimensional data, respectively, providing an advantage in real-world applications.

One reason why multiple-imputation-based methods provided good performances in presence of low-dimensional data might be due to their utility of the posterior predictive distribution of the missing data and harmonized feature selection over the feature estimates resulting in robust selections more so than point estimates of inverse covariance-based methods. While, the original paper of SALasso and GALasso claimed that adaptive weights improved the general performance [28], our results showed a marginal reduction. This discrepancy could be due to the lower imputation quality of the network-based multiple-imputation approach (ngMICE) we applied, propagating the imputation uncertainty into network inference. Increasing the number of multiple imputations could improve the performance, however, with a increase in runtime.

Multiple imputation in high-dimensional data is generally challenging since existing approaches do not scale to the number of covariates typically encountered in multi-omics studies. Here, dimensionality reduction methods for multiple imputation [49], latent factor models [50], or deep-learning-based approaches [51–53] might improve multiple imputation and therefore network inference quality. However, a benefit of such explicit approaches is their ability to use and adequately address prior imputed datasets often provided by larger consortia.

Implicit methods relying on inverse covariance matrix estimation performed poorly on low-dimensional data but performed generally well on high-dimensional data, even outperforming knnSGLasso and SLasso/SALasso in presence of block-missingness, indicating that large volumes of information are required to estimate the covariance structure properly with such methods.

We note that a true gold standard for evaluating the performance of network inference methods is missing. In its absence, we rely on networks inferred from complete, unperturbed data that nevertheless are likely to contain both false positive and false negative interactions which may affect the results. Hence, our reference networks are not suited for evaluating methods following different principles for inference as GENIE3 [54], or other nonlinear approaches such as ARACNe and their derivatives [7,8]. Nevertheless, the methods tested here can be used as direct substitutes for other Lasso-based network inference frameworks such as wgLasso [13], pLasso [54] or PoLoBag [55] with similar expected behaviour as demonstrated here. Also, other linear methods such as WGCNA [5] or Petal [56] could benefit from two-step approaches and consistent feature selection across multiple imputed datasets, similar to G(A)Lasso. A way of such approaches to handle multiple imputed datasets could be through the usage of Rubin's rule after Fisher's Z-transformation, which allows to pool the estimated correlation coefficients between omics features from multiple imputed datasets in a consistent manner [57,58].

In summary, we found explicit methods to be more robust than methods implicitly handling missingness in most scenarios except on high-dimensional data with block-missingness. While most methods were tolerant to high levels of missingness, they were strongly affected by noise. While HMLasso was the fastest tested method, knnSGLasso showed the best trade-off between performance and runtime and is thus our recommended approach for handling missingness in KiMONo except for high-dimensional data with block-missingness where we recommend HMLasso instead. While we see room for further method improvements, particularly with respect to multiple imputations of high-dimensional data and the robustness of inverse covariance methods, our results

show that robust multi-omics network inference in the presence of missingness is feasible with KiMONo and thus allows users to leverage available multi-omics data to their fullest extent.

Key Points:

- We extended KiMONo, a multi-omics network inference approach that is using prior network information, to handle data with missingness by integrating advanced Lasso-based inference techniques.
- knnSGLasso outperformed all other methods in the majority of scenarios except in presence of block-missingness where the kNN-imputation step tends to fail. In such cases, we recommend using SLasso on low- and HMLasso on high-dimensional data.
- Inverse covariance matrix estimation-based approaches were considerably sensible to the dimensionality of the input data, requiring high-dimensional data to reach high performance.

Contributions

JDH, MiL, MA, AG implemented the different lasso approaches. **JDH** implemented the benchmark framework**. JDH, BS** analysed the results**. BS, CO, MaL** conceived and supervised the study. **JDH, BS, CO, MaL** wrote the original manuscript. **MaL, CO, FJT, BS** reviewed and edited the manuscript. All authors read and approved the final manuscript.

Funding

This work has been supported by the German Centre of Lung Research (DZL). **BS** also acknowledges financial support from the Helmholtz International Lab “Causal Cell Dynamics”. **MiL** was supported by the Hanns Seidel Foundation. This study was also supported by the BMBF (German Federal Ministry of Education and Research) Project TRY-IBD Grant 01ZX1915B for **CO** and **AG**.

Conflict of Interest: **FJT** reports receiving consulting fees from Roche Diagnostics GmbH and Cellarity Inc., and an ownership interest in Cellarity, Inc. The remaining authors declare no competing interests.

References

1. Romero IG, Ruvinsky I, Gilad Y. Comparative studies of gene expression and the evolution of gene regulation. Nat. Rev. Genet. 2012; 13:505–516

2. Lee J, Hyeon DY, Hwang D. Single-cell multiomics: technologies and data analysis methods. Exp. Mol. Med. 2020; 52:1428–1442

3. Li Y, Ma L, Wu D, et al. Advances in bulk and single-cell multi-omics approaches for systems biology and precision medicine. Brief. Bioinform. 2021; 22:

4. Hawe JS, Theis FJ, Heinig M. Inferring Interaction Networks From Multi-Omics Data. Front. Genet. 2019; 10:535

5. Langfelder P, Horvath S. WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics 2008; 9:559

6. Song L, Langfelder P, Horvath S. Comparison of co-expression measures: mutual information, correlation, and model based indices. BMC Bioinformatics 2012; 13:328

7. Margolin AA, Nemenman I, Basso K, et al. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics 2006; 7 Suppl 1:S7

8. Lachmann A, Giorgi FM, Lopez G, et al. ARACNe-AP: gene network reverse engineering through adaptive partitioning inference of mutual information. Bioinformatics 2016; 32:2233–2235

9. Krumsiek J, Suhre K, Illig T, et al. Gaussian graphical modeling reconstructs pathway reactions from high-throughput metabolomics data. BMC Syst. Biol. 2011; 5:21

10. Schäfer J, Strimmer K. An empirical Bayes approach to inferring large-scale gene association networks. Bioinformatics 2005; 21:754–764

11. Petralia F, Wang P, Yang J, et al. Integrative random forest for gene regulatory network inference. Bioinformatics 2015; 31:i197–205

12. Sass S, Buettner F, Mueller NS, et al. A modular framework for gene set analysis integrating multilevel omics data. Nucleic Acids Res. 2013; 41:9622–9633

13. Li Y, Jackson SA. Gene Network Reconstruction by Integration of Prior Biological Knowledge.

G3 2015; 5:1075–1079

14. List M, Dehghani Amirabad A, Kostka D, et al. Large-scale inference of competing endogenous RNA networks with sparse partial correlation. Bioinformatics 2019; 35:i596–i604

15. Ogris C, Hu Y, Arloth J, et al. Versatile knowledge guided network inference method for prioritizing key regulatory factors in multi-omics data. Sci. Rep. 2021; 11:6806

16. Rubin DB. Multiple Imputation for Nonresponse in Surveys. 2004;

17. Donders ART, van der Heijden GJMG, Stijnen T, et al. Review: A gentle introduction to imputation of missing values. J. Clin. Epidemiol. 2006; 59:1087–1091

18. Wood AM, White IR, Royston P. How should variable selection be performed with multiply imputed data? Stat. Med. 2008; 27:3227–3246

19. Ganti R, Willett RM. Sparse Linear Regression With Missing Data. arXiv [stat.ML] 2015;

20. Ibrahim JG, Chen M-H, Kim S. Bayesian variable selection for the Cox regression model with missing covariates. Lifetime Data Anal. 2008; 14:496–520

21. Yang X, Belin TR, Boscardin WJ. Imputation and variable selection in linear regression models with missing covariates. Biometrics 2005; 61:498–506

22. Heymans MW, van Buuren S, Knol DL, et al. Variable selection under multiple imputation using the bootstrap in a prognostic study. BMC Med. Res. Methodol. 2007; 7:33

23. Liu Y, Wang Y, Feng Y, et al. VARIABLE SELECTION AND PREDICTION WITH INCOMPLETE HIGH-DIMENSIONAL DATA. Ann. Appl. Stat. 2016; 10:418–450

24. Wan Y, Datta S, Conklin DJ, et al. Variable selection models based on multiple imputation with an application for predicting median effective dose and maximum effect. J. Stat. Comput. Simul. 2015; 85:1902–1916

25. Chen Q, Wang S. Variable selection for multiply-imputed data with application to dioxin exposure study. Statistics in Medicine 2013; 32:3646–3659

26. Geronimi J, Saporta G. Variable selection for multiply-imputed data with penalized generalized estimating equations. Comput. Stat. Data Anal. 2017; 110:103–114

27. Marino M, Buxton OM, Li Y. Covariate Selection for Multilevel Models with Missing Data. Stat 2017; 6:31–46

28. Du J, Boss J, Han P, et al. Variable selection with multiply-imputed datasets: choosing between stacked and grouped methods. J. Comput. Graph. Stat. 2022; 1–35

29. Choi Y, Tibshirani R. An Investigation of Methods for Handling Missing Data with Penalized Regression. arXiv [stat.AP] 2013;

30. Johnson BA, Lin DY, Zeng D. Penalized Estimating Functions and Variable Selection in Semiparametric Regression Models. J. Am. Stat. Assoc. 2008; 103:672–680

31. Loh P-L, Wainwright MJ. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. Adv. Neural Inf. Process. Syst. 2011; 24:

32. Städler N, Bühlmann P. Missing values: sparse inverse covariance estimation and an extension to sparse regression. Stat. Comput. 2012; 22:219–235

33. Takada M, Fujisawa H, Nishikawa T. HMLasso: Lasso with High Missing Rate. arXiv [stat.ML] 2018;

34. Datta A, Zou H. CoCoLasso for high-dimensional error-in-variables regression. aos 2017; 45:2400–2426

35. Shen C-W, Chen Y-H. Model selection for generalized estimating equations accommodating dropout missingness. Biometrics 2012; 68:1046–1054

36. Sabbe N, Thas O, Ottoy J-P. EMLasso: logistic lasso with missing data. Stat. Med. 2013; 32:3143–3157

37. Yu G, Li Q, Shen D, et al. Optimal Sparse Linear Prediction for Block-missing Multi-modality Data without Imputation. J. Am. Stat. Assoc. 2020; 115:1406–1419

38. Xue F, Qu A. Integrating Multisource Block-Wise Missing Data in Model Selection. J. Am. Stat. Assoc. 2021; 116:1914–1927

39. Gentry AE, Kirkpatrick RM, Peterson RE, et al. Missingness Adapted Group Informed Clustered (MAGIC)-LASSO: A novel paradigm for prediction in data with widespread non-random

missingness. bioRxiv 2021; 2021.04.29.442057

40. Simon N, Friedman J, Hastie T, et al. A Sparse-Group Lasso. J. Comput. Graph. Stat. 2013; 22:231–245

41. Henao JD, Lauber M, Azevedo M, et al. Multi-Omics Regulatory Network Inference in the Presents of Missing Data. 2022;

42. Troyanskaya O, Cantor M, Sherlock G, et al. Missing value estimation methods for DNA microarrays. Bioinformatics 2001; 17:520–525

43. Escribe C, Lu T, Keller-Baruch J, et al. Block coordinate descent algorithm improves variable selection and estimation in error-in-variables regression. Genet. Epidemiol. 2021; 45:874–890

44. Weinstein JN, Collisson EA, Mills GB, et al. The Cancer Genome Atlas Pan-Cancer analysis project. Nat. Genet. 2013; 45:1113–1120

45. Gao J, Lindsay J, Watt S, et al. Abstract 5277: The cBioPortal for cancer genomics and its application in precision oncology. Cancer Res. 2016; 76:5277–5277

46. Oughtred R, Rust J, Chang C, et al. The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions. Protein Sci. 2021; 30:187–200

47. Ogris C, Guala D, Sonnhammer ELL. FunCoup 4: new species, data, and visualization. Nucleic Acids Res. 2018; 46:D601–D607

48. van Buuren S, Groothuis-Oudshoorn K. mice: Multivariate Imputation by Chained Equations in R. J. Stat. Softw. 2011; 45:1–67

49. Hodge DW, Safo SE, Long Q. Multiple imputation using dimension reduction techniques for high-dimensional data. arXiv [stat.ME] 2019;

50. Argelaguet R, Arnol D, Bredikhin D, et al. MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. Genome Biol. 2020; 21:111

51. Qiu YL, Zheng H, Gevaert O. Genomic data imputation with variational auto-encoders. Gigascience 2020; 9:

52. Gayoso A, Steier Z, Lopez R, et al. Joint probabilistic modeling of single-cell multi-omic data with

totalVI. Nat. Methods 2021; 18:272–282

53. Lotfollahi M, Litinetskaya A, Theis FJ. Multigrate: single-cell multi-omic data integration. bioRxiv 2022; 2022.03.16.484643

54. Huynh-Thu VA, Irrthum A, Wehenkel L, et al. Inferring regulatory networks from expression data using tree-based methods. PLoS One 2010; 5: