



OPEN

Exploratory analysis of metabolic changes using mass spectrometry data and graph embeddings

Edwin Alvarez-Mamani^{1,2}, Florian Buettner^{3,4,5}, Cesar A. Beltran-Castanon¹ & Alfredo J. Ibanez^{2,6}✉

Mass spectrometry (MS)-based metabolomics analysis is a powerful tool, but it comes with its own set of challenges. The MS workflow involves multiple steps before its interpretation in what is denominated data mining. Data mining consists of a two-step process. First, the MS data is ordered, arranged, and presented for filtering before being analyzed. Second, the filtered and reduced data are analyzed using statistics to remove further variability. This holds true particularly for MS-based untargeted metabolomics studies, which focused on understanding fold changes in metabolic networks. Since the task of filtering and identifying changes from a large dataset is challenging, automated techniques for mining untargeted MS-based metabolomic data are needed. The traditional statistics-based approach tends to overfilter raw data, which may result in the removal of relevant data and lead to the identification of fewer metabolomic changes. This limitation of the traditional approach underscores the need for a new method. In this work, we present a novel deep learning approach using node embeddings (powered by GNNs), edge embeddings, and anomaly detection algorithm to analyze the data generated by mass spectrometry (MS)-based metabolomics called GEMNA (Graph Embedding-based Metabolomics Network Analysis), for example for an untargeted volatile study on Mentos candy, the data clusters produced by GEMNA were better than the ones used traditional tools, i.e., GEMNA has *silhouette score* = 0.409, vs. the traditional approach has *silhouette score* = −0.004.

Keywords Mass spectrometry, Metabolomic networks, Graph neural networks, Graph embeddings

Mass spectrometry (MS)-based metabolomics analysis is frequently used due to broad analyte coverage, high sensitivity, high selectivity, and high performance¹. The MS-based metabolomic analytical workflow involves multiple steps before its interpretation in what is called data mining. The requirement for data mining is that the raw data generated by MS techniques requires normalization and noise filtration to obtain more precise results for data interpretation. Furthermore, when the MS instrument is coupled to other orthogonal analytical techniques such as *Gas chromatography-mass spectrometry* (GC-MS) and *Liquid chromatography-mass spectrometry* (LC-MS), more complex processes such as peak identification and peak alignment are required.

In more detail, the data mining process aims to deconvolute the relevant data, i.e., the data associated with the property of interest (i.e., phenotype) from the rest of the data (i.e., noise based are products of electrical noise, chemical contamination, or any other artifact). Thus, data mining consists of a two-step process. During the first step, the MS data is ordered, arranged, and presented for filtering (data reduction) before being analyzed. Subsequently, in the second step, the filtered and reduced data are analyzed using descriptive and inferential statistics to remove further variability. This two-step approach facilitates data interpretation^{2,3}. This holds true particularly for MS-based untargeted metabolomics studies, which focused on understanding fold changes in metabolic networks by monitoring the changes in as many individual metabolites as possible. These changes are studied by comparing at least two types of samples, one denominated baseline and the others in an altered state. The idea of changes in the metabolic network between metabolites due to a stress or mutation induction was developed by^{4,5}, where the authors proposed monitoring changes in the metabolite network. Since the task of filtering and identifying changes from a large dataset to determine its quality is unrealistic, automated techniques for mining untargeted MS-based metabolomic data are needed. The traditional approach based on

¹Engineering Department, Pontificia Universidad Catolica del Peru, Lima, Peru. ²Institute for Omics Sciences and Applied Biotechnology, Pontificia Universidad Catolica del Peru, Lima, Peru. ³Goethe University, Frankfurt, Frankfurt am Main, Germany. ⁴German Cancer Consortium (DKTK), Frankfurt am Main, Germany. ⁵German Cancer Research Center (DKFZ), Frankfurt am Main, Germany. ⁶Science Department, Pontificia Universidad Catolica del Peru, Lima, Peru. ✉email: aibanez@pucp.edu.pe

statistics (ANOVA, t-Test, coefficient of variation, etc.)^{6,7} filter raw data very strongly, and there is the possibility of eliminating relevant data, as a result, identify less metabolomic changes.

In a recent review article⁸, we described the use of graph representation learning on biomedical data (i.e., protein-protein interaction networks, and metabolite-metabolite networks). For instance, *Shallow embeddings*-based and *Autoencoders*-based models focus on simpler tasks such as solving link prediction, in terms of biology tasks to find a drug-target prediction. Within *Shallow embeddings*-based there are DeepWalk⁹, Node2vec¹⁰, and Struc2vec¹¹; *Autoencoders*-based: SDNE¹², DNGR¹³, etc. More complex problems, such as graph matching, in biology terms to compare two metabolic phenotypes require more advanced models such as *Graph neural network*-based: Variational Graph Autoencoder VGAE¹⁴, Deep Graph Infomax DGI¹⁵, Adversarially Regularized Variational Graph Auto-Encoder ARGVA¹⁶ and Linear Graph Variational Autoencoder LGVAE¹⁷. The advantage of GNN-based models over the more straightforward graph embedding models is that they are more robust because GNNs can share parameters between nodes (through message passing), work directly on the graphs, and take advantage of the node features and the graph structure¹⁸.

In this work, we present a novel deep learning approach using node embeddings (powered by GNNs), edge embeddings, and anomaly detection algorithm to analyze the data generated by mass spectrometry (MS)-based metabolomics called GEMNA, i.e., Graph Embedding-based Metabolomics Network Analysis (Supplementary Fig. S1 and S2). Although embeddings have been used previously in mass spectrometry, this is the first time they have been used for MS-based data filtration, and later identify changes in the filtered metabolic networks. Receiving as input the MS data obtained either by an (i) flow injection MS or chromatography coupled-MS system, GEMNA identifies the ‘real’ signals by using embedding filtration couple with a GNN model; and generating as output the filtered MS-based signal list and a dashboard with graphs showing the changes between metabolites among two or more samples.

Our proposal promises to help non-specialized MS researchers in chemistry and biology better understand/visualize their MS results for decision-making and data interpretation. Moreover, in the case of an unknown sample or first-time measurement, these steps together represent approximately 90% of the time in an analytical laboratory. Hence, embedding models could accelerate this process, providing life scientists and health researchers with a new toolbox.

Experiments and results

In this section, we evaluate our proposal, on three real datasets containing metabolomic studies and one synthetic dataset. The source code was divided in backend and frontend. The backend was implemented in Django rest framework, with PyTorch Geometric, PyOD libraries, and it is available at https://github.com/win7/GEMNA_Backend.git. Meanwhile, the frontend was implemented in Vue.js with Nuxt framework, and it is available at https://github.com/win7/GEMNA_Frontend.git. The experiments were run on an AMD Ryzen Threadripper PRO 5955WX computer with 32 cores, 256 GB of RAM, 2x NVIDIA RTX A6000 with 48 GB of VRAM. However, the experiments could be run on a computer with 16 GB of RAM and 8 GB of VRAM. For example, our Mentos dataset took on average 8.45 min to be analyzed using the AMD Ryzen system, while it took on average 12.66 min on the other.

Dataset

The datasets used to perform the experiments are described below. Additionally, further biological details of the real raw data are shown in Table 1.

Synthetic dataset

We generated synthetic data using a multivariate normal distribution (Gaussian distribution), the distribution depends on the mean and the covariance matrix. The mean was in the range of 0 to 100 000 and the covariance matrix values were based on positive semi-definite matrices. Two datasets were created, each one representing a phenotype. The dataset size was 1 000 samples with 200 features. After that, for the experiments, a number of features were selected to generate groups representing biological and analytical repetitions. Then each phenotype may have 2, 3, 4 or 5 biological repetitions and 2, 3, 4 or 5 analytical repetitions, where an analytical repetition is a feature.

Dataset	Metabolites	Phenotypes	Biological rep.	Analytical rep.
Mutant	6 245	WT	5	40, 40, 40, 40, 40
		PFK1	2	40, 39
		ZWF1	3	40, 40, 40
Leaf	943	Control	3	3, 3, 3
		Treatment	3	3, 3, 3
Mentos	1 782	Orange	2	3, 3
		Red	2	3, 3
		Yellow	2	3, 3

Table 1. Raw data, biological details.

Mutant dataset

The dataset was generated by Buettner et al.⁶. In their publication, the data was acquired using an ESI-MS metabolomics (non-targeted) mass spectrometry-based to study *Saccharomyces cerevisiae*, auxotrophic, BY4742 strains. The ZWF1 mutant strain lacks one of the reactions from glycolysis into the PPP and the major NADPH-producing step; meanwhile, the PFK1 mutant strain lacks one of the *Phosphofructokinase* enzymes, hence making the glycolysis pathway less efficient. According to the publication⁶, cellular extracts of these phenotypes were analyzed on a flow injection tandem (quadrupole time-of-flight) mass spectrometer (Agilent 6520) operating in negative mode scanning from 50 to 1000 *m/z* at a frequency of 1.4 full spectra per second.

Leaf dataset

The dataset was generated by González-Teuber et al.⁷. The authors used an ESI-MS metabolomics experiment (non-targeted) to evaluate the effects of warming on plant (*Aristolochia chilensis*) performance (growth, leaf area and chlorophyll) and to estimate the overall changes in leaf metabolites between the ambient control and when treated with the warming simulation. *Aristolochia chilensis*’ leaf samples were obtained from both WT and heat-stress plants in the field at Praderas de lo Aguirre, *Región Metropolitana, Santiago, Chile* (*N* = 10 plants per treatment) by Dr. González-Teuber’s team. The Dr. González-Teuber’s team collected the plant material following the Pontifical Catholic University of Chile, Chilean, and international guidelines and legislation. According to the publication⁷, leaf samples were shock-frozen in liquid nitrogen immediately after dissection for transportation, lyophilized, and subsequently stored at −80 °C. Stock solutions were prepared by grinding 40 *mg* of leaves under cold conditions. The leaf powder was dissolved in 1 *mL* of MeOH, vortexed for a period of 5 *min*, and then filtered with a 0.20 μ *M* syringe disk filter. Stock solutions were stored at −20 °C until measurement. A dilution solvent, comprising a mixture of MeOH and ddH₂O containing 0.5% formic acid (FA; 1:1), was used to dilute stock solutions at a ratio of 1:200. The samples were directly injected into an Orbitrap Q-Exactive HF mass spectrometer (MS; Thermo Fisher Scientific, United States) at a flow rate of 20 μ *L/min*. The MS data were acquired at a resolution of 60 000 in positive mode data-dependent acquisition (DDA) with a scan range of 50 – 750 *m/z*. Raw data from this experiment were first extracted using an in-house data processing tool built in MatLab (vR2022a). Tentative identification was performed using the Metlin¹⁹ and the mzCloud (<https://www.mzcloud.org/>) repository.

Mentos dataset

The samples were collected by placing 3 *g* of one flavor (strawberry, orange, or lemon) of Mentos candy in a 20 *mL* headspace vial (Thermo Scientific, USA). A 50/30 μ *m* DVB/CAR/PDMS coated SPME fiber (Supelco-Aldrich, Bellefonte, PA, USA) was thermally cleaned and desorbed between samples. The fibers were conditioned for 30 *min* at 250 °C. Then, the fibers were introduced inside the headspace vial and incubated at 40 °C for 30 *min* with agitation. The SPME fiber was placed into the injector port at 250 °C for 3 *min*. Helium carrier gas (99.999%) was set at a constant 1.2 *mL/min* flow. The oven temperature was programmed from 40 °C (held for 3 *min*) to 250 °C at 10 °C/*min* (held for 10 *min*). A DB-5 column (30 *m* × 250 μ *m*, 0.25 μ *m* film thickness; Agilent) was used. The Thermo TSQ 9610 was used as a detector, with the MS transfer line temperature set at 250 °C and the ion source (electron impact, 70 *kV*) maintained at 250 °C. The TSQ operated in full-scan acquisition mode with a mass range of 45 – 1100 *m/z*. The raw data, a comprehensive record of our experiments, were meticulously processed with MS-DIAL, ensuring the accuracy and reliability of our results.

Experiments setup

According to the pipelines, the experiments involve several steps, one of the most relevant steps is the networking filtering through of the node and edge embeddings. Table 2 shows the possible configurations, each configuration was run 5 times to determine common subnetworks and the runtimes.

Features	Values
Node embedding	VGAE, LVGAE, ARGVA, DGI
Positional encoding	Laplacian PE, Random Walk PE
Dimensions	2, 3, 4, 8, 16, 32, 64, 128
Optimizer	Adam with $lr = 10^{-2}$, $weight\ decay = 10^{-4}$
Early stopping	<i>Patience</i> = 10
Epochs	100
Edge embedding	Weighted-L2
Outliers detection	COPOD with (<i>contamination</i> = 0.1)
Dataset	Synthetic, Mutant, Leaf, Mentos
Network variations	<i>None</i> (0:0), <i>one-to-one</i> (1:1), <i>many-to-many</i> (m:m)

Table 2. Experiments setup. Laplacian eigenvector, Random Walk were applied only on VGAE, LVGAE, ARGVA, DGI

Results

This section is divided into two parts, in the first part we show the influence of the number of biological repeats and analytical repeats on the task of finding a subnetwork. In the second part, we show the results for each proposed pipeline, including the runtimes to generate node embeddings, and edge embeddings.

Influence of raw data

Using our algorithm on our synthetic data set illustrates that the efficiency of network filtering depends on the number of biological repetitions and analytical repetitions for each sample type (phenotype). Fig. 1 shows the number of common edges (i.e., pattern) is influenced by the number of biological repetitions (e.g., 2, 3, 4, or 5) and their respective analytical repetitions (e.g., 2, 3, 4, or 5). While a low number of analytical repetitions show a better coverage (i.e., sensitivity), a higher number of analytical repetitions increases the variability, and more common edges are filtrated, i.e., the common subnetworks (i.e., phenotype pattern) contain fewer common edges (i.e., robustness). This result is similar to the one observed in DNA sequencing. The more measurements one has (depth), the more it is possible to discover single nucleotide polymorphisms. After the experiments with the synthetic dataset, it was concluded that the minimal data set must consist of 2 biological repetitions (each of them with two analytical measurements) to generate partial correlations with coefficients of -1 , 1 , or *null*. However, for generating correlation coefficients with normal distribution, minimum requirements for 6 data points are needed for one phenotype, Fig. 8(b). Ideally, these 6 data points for one phenotype consist of 2 biological repetitions, each with three analytical measurements. Meanwhile, the greedy (filtering) version finds all common edges, including abnormal edges (outliers).

Pipeline results

In this section, we display and describe the results obtained by each pipeline of our proposal.

Network generation Since the signal filtering will happen using embeddings, it is important that the original networks will be in the same latent space for an optimum comparison. Thus, adding variations to the partial correlation networks, such as linking one common node among biological repeats or linking each common node among biological repeats, forces all embeddings will be present in the same dimensional space. The details of the Synthetic, Leaf, and Mentos networks are shown in Tables 3, 4, and 5, for each biological feature or class (i.e., phenotype) and with their respective network variations (*none*, *one-to-one*, *many-to-many*). For each network, the edge features are partial correlation coefficients.

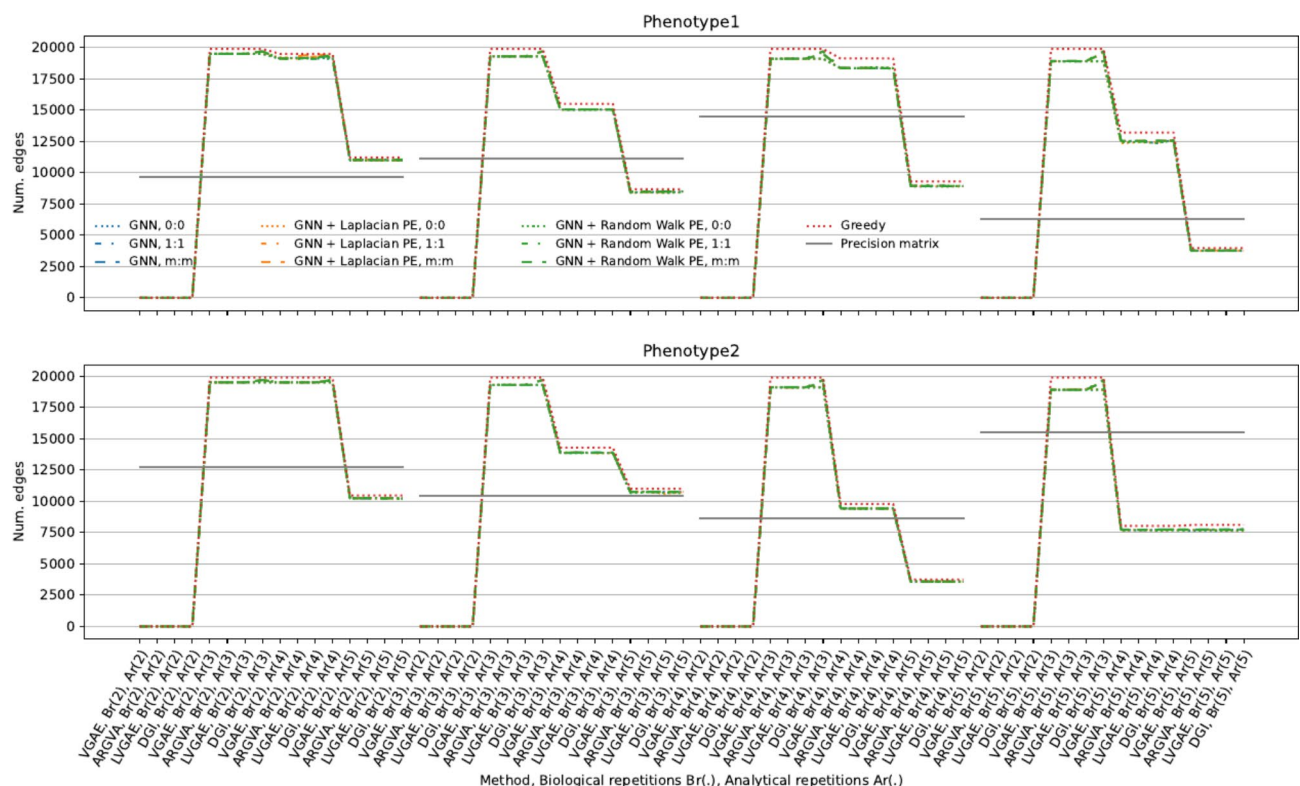


Fig. 1. Performance of biological repetitions (Br) and analytical repetitions (Ar). The red dotted line represents the number of common edges found by a greedy algorithm, which does not eliminate edges considered as noise (outlier). The solid grey line represents the number of common edges based on the inverse of the covariance matrix (Precision matrix).

Phenotype	Variation	Num. nodes	Num. edges	Density	Node feat. (dim.)
Phenotype1	0:0	200	19 900	1.0	Signal intensity (3)
		200	19 900	1.0	Signal intensity (3)
	1:1	400	39 801	0.49	Signal intensity (3)
	m:m	400	40 000	0.50	Signal intensity (3)
Phenotype2	0:0	200	19 900	1.0	Signal intensity (3)
		200	19 900	1.0	Signal intensity (3)
	1:1	400	39 801	0.49	Signal intensity (3)
	m:m	400	40 000	0.50	Signal intensity (3)

Table 3. Synthetic networks details.

Phenotype	Variation	Num. nodes	Num. edges	Density	Node feat. (dim.)
Control	0:0	890	255 685	0.65	Signal intensity (3)
		826	227 941	0.67	Signal intensity (3)
		837	237 457	0.68	Signal intensity (3)
	1:1	2 553	721 085	0.22	Signal intensity (3)
	m:m	2 553	722 709	0.22	Signal intensity (3)
Warming-treated	0:0	877	249 858	0.65	Signal intensity (3)
		825	228 861	0.67	Signal intensity (3)
		851	235 812	0.65	Signal intensity (3)
	1:1	2 553	714 533	0.22	Signal intensity (3)
	m:m	2 553	716 144	0.22	Signal intensity (3)

Table 4. Leaf networks details.

Phenotype	Variation	Num. nodes	Num. edges	Density	Node feat. (dim.)
Orange	0:0	1 779	1 098 769	0.69	Signal intensity (3)
		1 771	1 064 854	0.68	Signal intensity (3)
	1:1	3 550	2 163 624	0.34	Signal intensity (3)
	m:m	3 550	2 165 392	0.34	Signal intensity (3)
Red	0:0	1 777	1 066 283	0.68	Signal intensity (3)
		1 772	1 084 732	0.69	Signal intensity (3)
	1:1	3 549	2 151 016	0.34	Signal intensity (3)
	m:m	3 549	2 152 784	0.34	Signal intensity (3)
Yellow	0:0	1 780	1 060 719	0.67	Signal intensity (3)
		1 771	1 053 119	0.67	Signal intensity (3)
	1:1	3 551	2 113 839	0.34	Signal intensity (3)
	m:m	3 551	2 115 608	0.34	Signal intensity (3)

Table 5. Mentos networks details.

Network filtering Traditionally, filtering MS data is based on the CV% of a signal or the peak shape. Here, filtering is focused on obtaining a common subnetwork from the metabolic correlation network (corpus networks) of a phenotype. The filtering process depends essentially on the generation of node embeddings, edge embeddings, and anomaly detection. The Supplementary Fig. S3 displays this process of embeddings manipulation.

To evaluate the over-smoothing issue in the generation of the node embeddings, we used the Mean Average Distance (MAD) metric²⁰, in Fig. 3, we show the behavior of Laplacian PE, and RandonWalk PE, a value close to 0, represents that the node embeddings are indistinguishable. In the same way, Fig. 2 shows the number of common edges that could be filtered, according to the features in the Table 2. Thus, from Fig. 2, we can observe that, the *many-to-many* network variation find more common edges. Moreover, adding Laplacian PE helps to find a more significant number of common edges. Regarding the dimension of embeddings, they are visualized that the results are similar, however, the runtimes to generate the node embeddings and especially to generate the edge embeddings grows according to the dimension of embeddings, where the curve looks like an exponential growth. The plots of the runtimes for the generation of the node embeddings are shown in Fig. 4(a) and for the case of edge embeddings in Fig. 4(b). To select the best model we performed a ranking (see Table 6) with the normalized data of Fig. 2 with *MinMax scaler*, as of result the LVGAE and ARGVA models find more

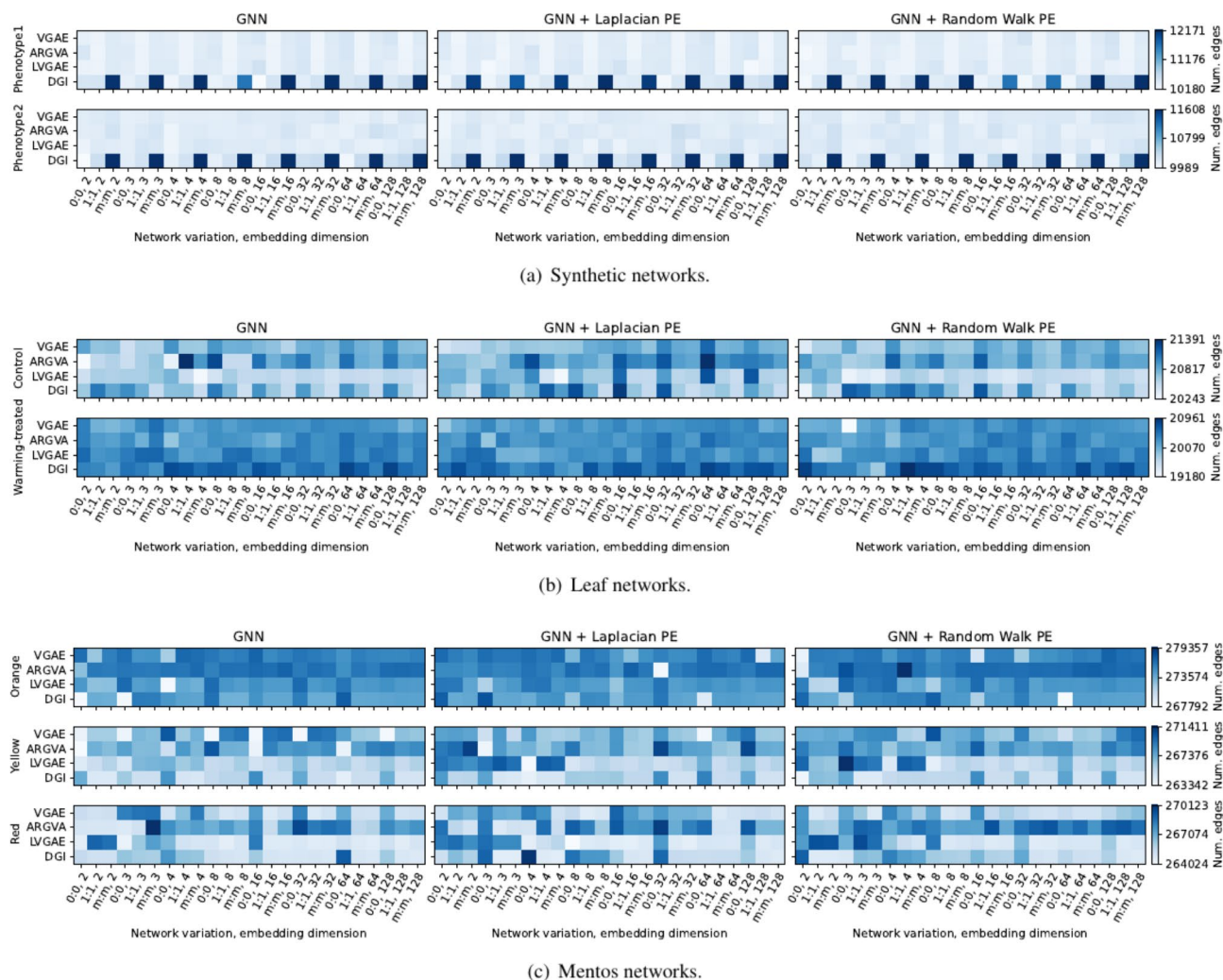


Fig. 2. Number of common edges in filtering networks.

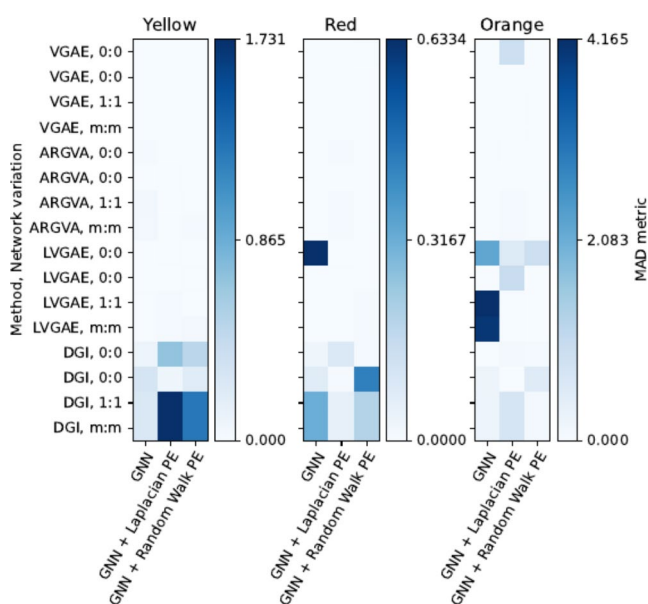


Fig. 3. MAD values on Mentos dataset.

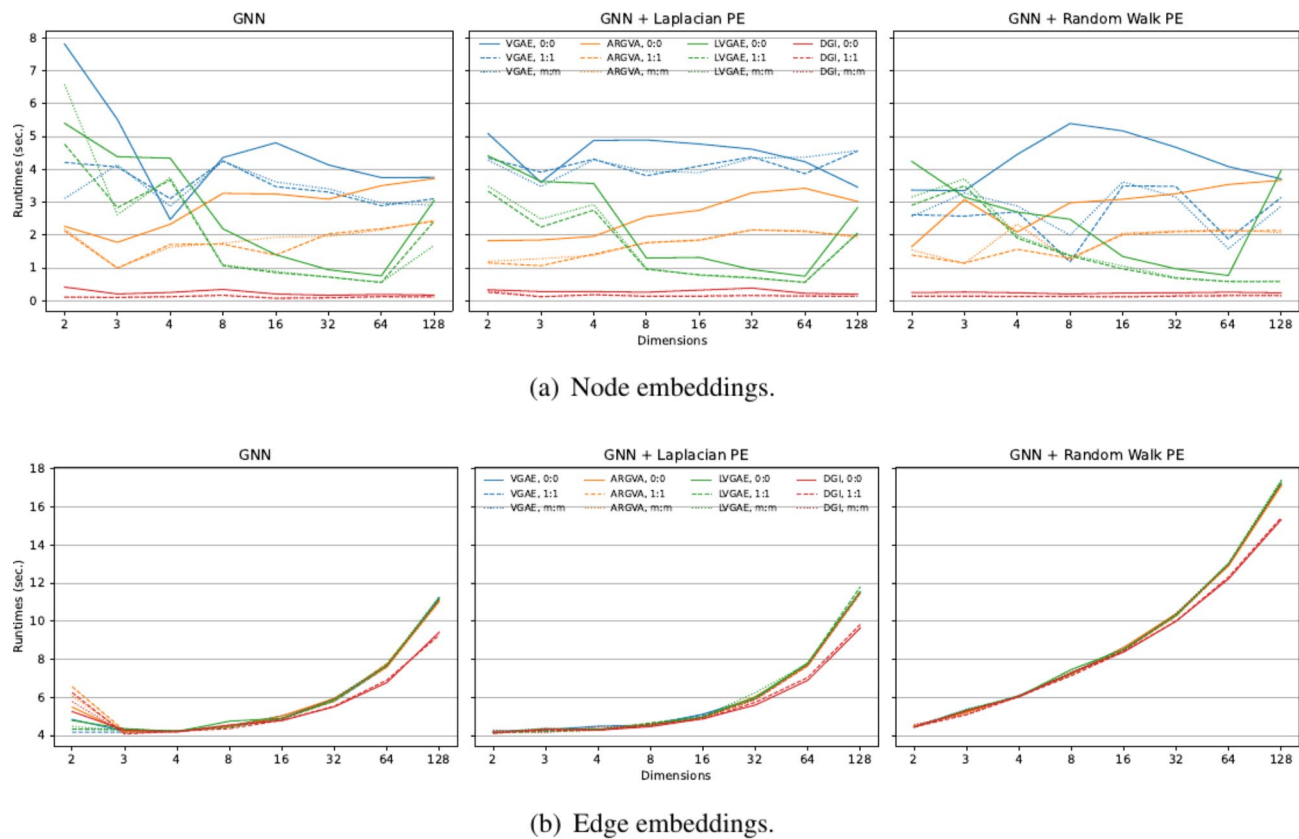


Fig. 4. Runtimes to generate embeddings on Synthetic networks.

Num.	GNN	GNN + Laplacian PE	GNN + Random Walk PE
1	ARGVA	DGI	ARGVA
2	DGI	ARGVA	DGI
3	VGAE	LVGAE	LVGAE
4	LVGAE	VGAE	VGAE

Table 6. Ranking of models to find common subnetwork.

edges in common. By default, our proposal approximately keeps 90% of common edges (it depends on anomaly detection algorithm parameters). Therefore, we expect the GNN models to be close to this percentage, and it keeps a maximum number of common edges.

In summary, to select the best performing configuration (experiments setup), we consider the configuration that filters a larger number of edges (see Fig. 2) in a shorter runtime (see Fig. 4). Based on the results, the best choices are the LVGAE and ARGVA models, with a network variation of *many-to-many*, and an embedding dimension of 3.

Similarity analysis This final step of the GEMNA workflow is associated to the visualization of the data. The objective of the similarity analysis is to identify the changes between a pair of metabolites in the partial correlation network. Note that one of the networks must be a control network. In Fig. 5 the significant correlation changes (metabolomic changes) for Mutant networks are visualized. The software can also generate heatmaps to visualize changes in terms of the intensity ratio between two metabolites. In Fig. 6 changes in intensity of selected MS signals exhibiting significant differences (One-way ANOVA) between ambient control (C) and warming-treated (W) *Aristolochia chilensis* plants. The Fig. 7 displays that GEMNA identifies more metabolites than standard approaches based on CV%. We performed a systematic quantitative comparison on the Mentos dataset. Here, we quantify the quality of clusters using the Silhouette score, and Intra-cluster distance. GEMNA outperforms traditional methods including, signal intensity, ANOVA, and CV% by a large margin, as reported in Table 7.

Method

The methodology used in network generation, network filtering and similarity analysis (metabolomic changes) between two or more metabolic networks is presented in this section.

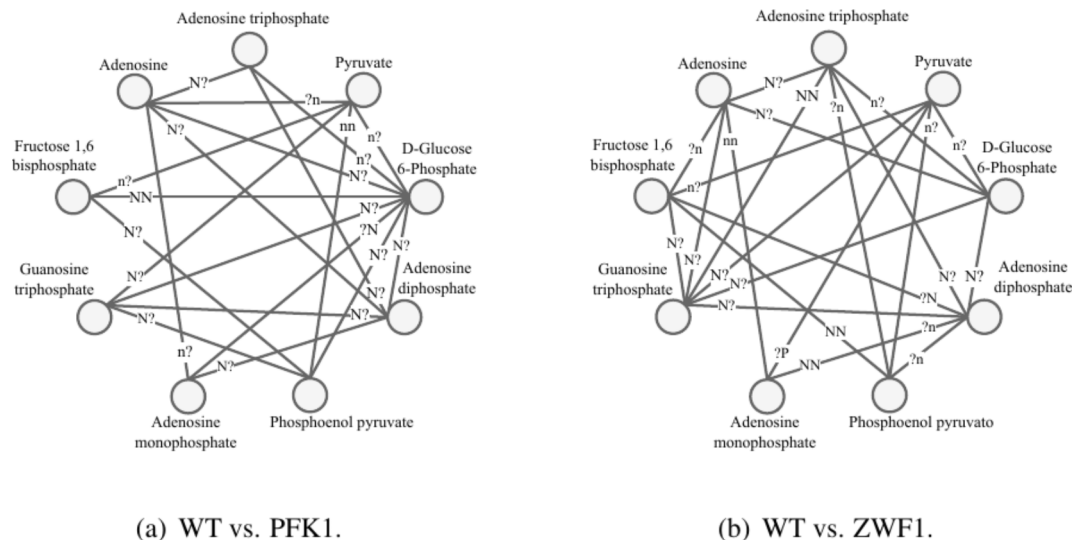


Fig. 5. Similarity analysis on Mutant network.

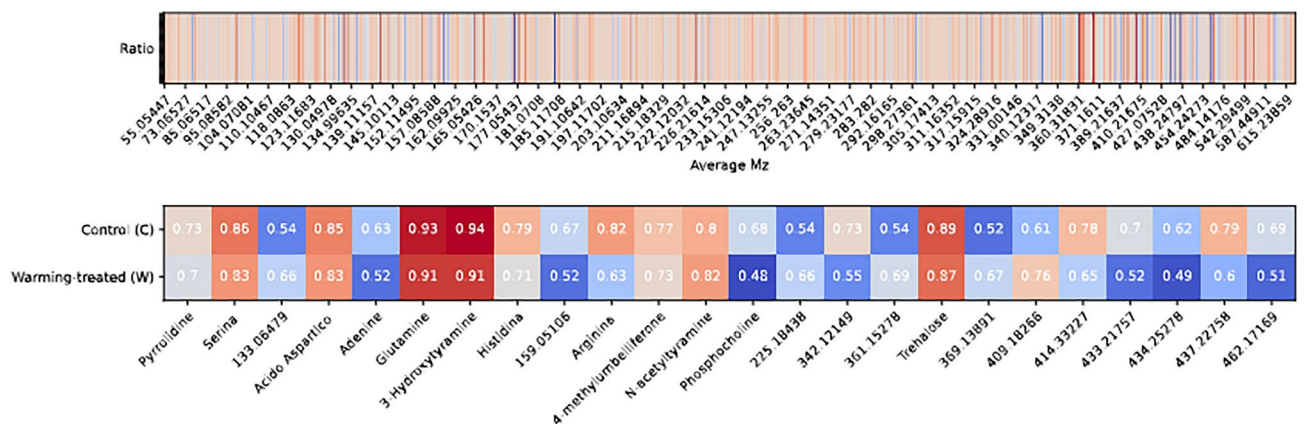


Fig. 6. Heatmap on Leaf filter data.

First, we will describe the format of the mass spectrometry data (raw data), and the requirements needed. The first 4 columns of the tables are metadata, while the rest of them are signal intensity (measurements), divided into biological and analytical repetitions for each phenotype, as shown in Fig. 8(a). We expect the data to have at least three analytical repetitions for each biological repetition (see Fig. 8b). If there are only two analytical repetitions, a third column can be added using the bootstrapping technique. Our proposal is described below.

Network generation

The raw data generated after a metabolomics study undergoes preprocessing, where the intensities of the identified compounds are aligned. Then from this, an additional preprocessing is applied to build the metabolomics networks. The steps of this pipeline 1 are described below.

- Remove not required variables that will not be used in the network filtering, and similarity analysis steps. The variables we keep are: *Alignment ID*, *Retention time* (if given), *Isotopic molecular ion (m/z)*, *Metabolite name* (if given) and signal intensity. Figure 8a shows the result.
- Transform measurements by $\log_{10}$ (only if necessary).
- Calculate the partial correlations on the measurements of analytical repetitions.
- Build networks, the edge between a pair of nodes (metabolites) (v_i, v_j) exists if, $|correlation(v_i, v_j)| \geq 0.5$, this condition involves moderate, strong or very strong correlations between those two metabolites influence for the rest of metabolites. The output of this pipeline 1 is a set of partial correlation networks for each phenotype. Figure 9a shows pipeline 1 with the network generation from raw data. Additionally, we add variations on the original networks (see Fig. 9(b)). The variation called *one-to-one* (1:1) consists of joining by an edge to a single node that belongs to the networks of the set of networks. Likewise, the variation called *many-to-many* (m:m) consists of joining by an edge all the nodes with the same identifier within the set of

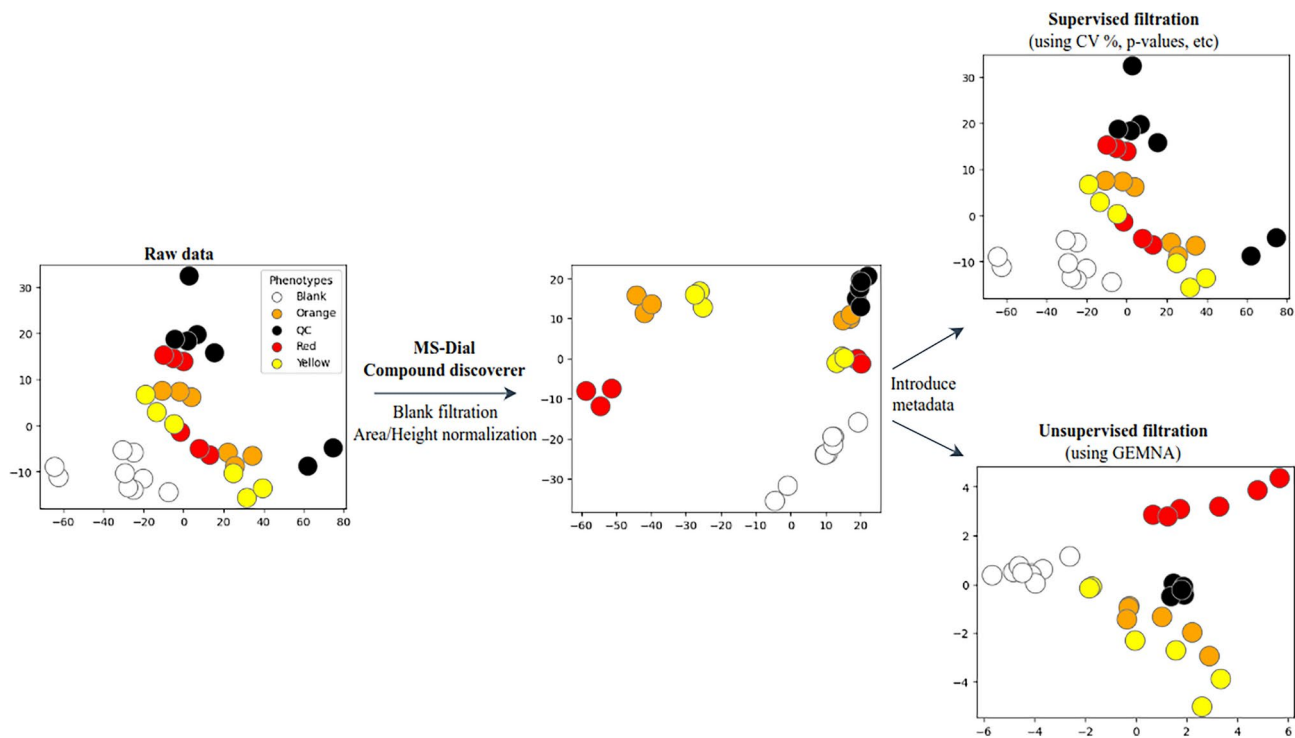


Fig. 7. PCA on Mentos filter data workflow.

Methods	Filtration (nodes remaining)				Intra-cluster distance			Silhouette score	Nodes lost
	MS-Dial features	+ ANOVA (> 0.05%)	+ CV % (< 40%)	+ GNN	Red	Orange	Yellow		
Baseline	1782	-	-	-	46.32	36.33	27.10	0.242	0
ANOVA	1782	622	-	-	3.96	5.99	9.59	0.154	1160
ANOVA + CV%	1782	622	215	-	17.01	23.88	32.60	- 0.004	1567
GEMNA	1782	-	-	301	2.60	1.96	3.69	0.409	1481

Table 7. Comparison between GEMNA vs. Statistics approach.

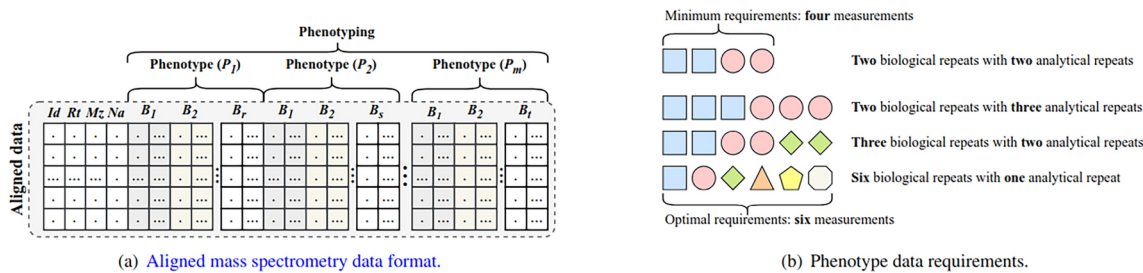


Fig. 8. Mass spectrometry data details. (a) *Id* is Alignment ID, *Rt* is Retention time, *Mz* is Average Mz, *Na* is Metabolite name, *B*₁, *B*₂, ..., *B*_{*} are Biological repetitions. In addition, each biological repetition has two analytical repetitions at least.

networks, also consecutively. These ideas were found from^{11,21} respectively. Adding these variations to the original set of networks helps to bring the node embeddings of each biological repeat to closer latent spaces and, therefore, to find anomalies more properly.

Network filtering

It receives as input two or more metabolomic networks of a similar phenotype $C = \{B_1, B_2, \dots, B_q\}$, with some network variations, as shown in Fig. 9b. Pipeline 2 includes the following steps.

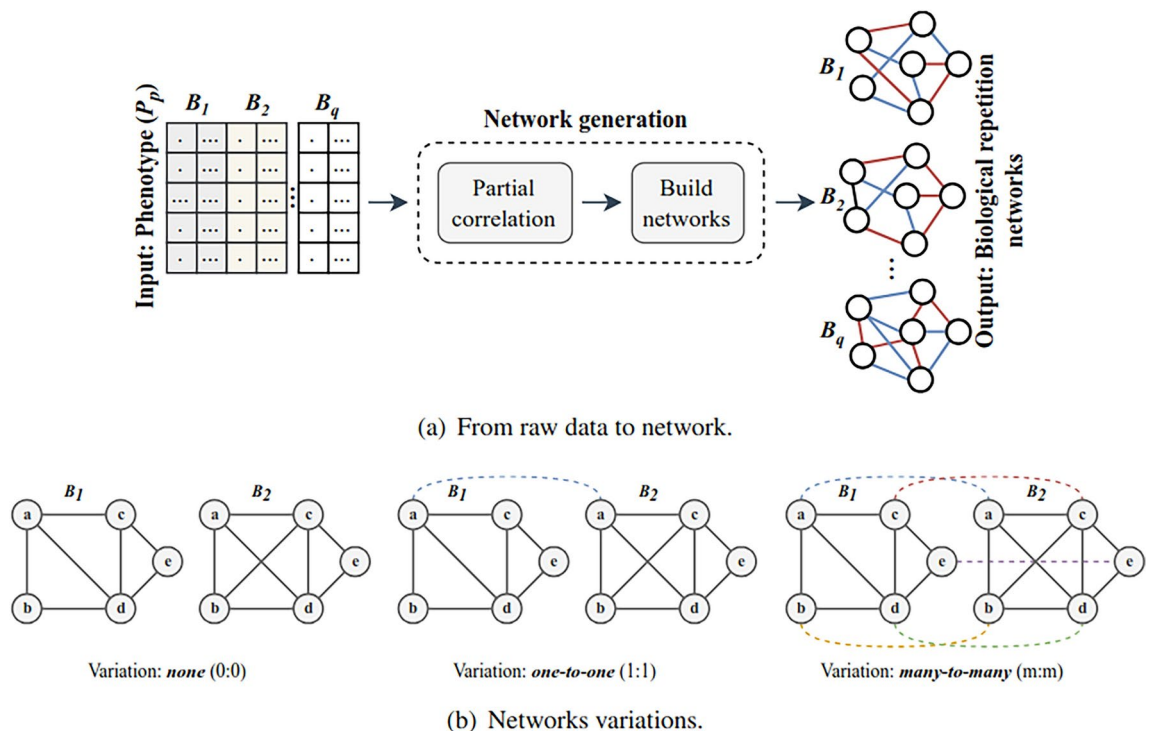


Fig. 9. Pipeline 1: Network generation. Note: **none** means, network without variation.

- i Generate node embeddings $Z_k, \forall G_k \in C$, using VGAE¹⁴, DGI¹⁵, ARGVA¹⁶ or LGVAE¹⁷ models. The architecture of VGAE, ARGVA, LGVAE contains encoders and decoders. The encoder of VGAE contains two Graph Convolutional Network (GCN)²² layers, the encoder of LGVAE replaces GCN layers with linear models, the encoder of ARGVA include an adversarial training scheme to regularize the latent space by GCN. On the other hand, DGI has four components: corruption function, encoder based on maximizing local mutual information, readout function, and discriminator. Thus, the decoder is trained to reconstruct the original graph structure. All of them were proposed to unsupervised tasks. Because of, the non-existence of positional information of the nodes; in the experiments we concatenate the node features with two positional encodings (PE), Random Walk PE²³ and Laplacian PE²⁴ to generate the node embeddings and leverage the GNNs. The positional encodings are a strategy that supports the positional and structural representation of GNNs.
- ii Calculate edge embeddings $\hat{Z}_k, \forall G_k \in C$, using Weighted-L2 operator, defined as: $|z_i - z_j|^2$.
- iii Concatenate (only for 0:0 network variation) and detect outliers (anomalous edges), according to edge embeddings. Therefore, $\hat{Z}_{concat} = \hat{Z}_1 \oplus \hat{Z}_2 \oplus \dots \oplus \hat{Z}_q$ and $inliers(\hat{Z}_{concat}) = \hat{Z}_{concat} - outliers(\hat{Z}_{concat})$, using Copula-Based Outlier Detection (COPOD) algorithm²⁵. Formally, a copula is the cumulative distribution function (CDF) of a vector with uniform marginals on the interval $[0, 1]$. COPOD, includes three steps, 1) calculate the CDFs of the dataset, 2) use the empirical CDFs to generate the copula function, 3) use the empirical copula to approximate the tail probability. The output of this pipeline is a filtered network, i.e., a common subnetwork for a biological network of one phenotype (sample type). Note: This common phenotype can be built by focusing on sensitivity (optimizing for the maximum number of nodes) or robustness (optimizing for the most stable interactions between nodes), for both cases using the common edges obtained. Pipeline 2 is shown in Fig. 10.

Similarity analysis

It receives as input two filtered metabolomic networks F_1, F_2 obtained in pipeline 2 and consists of the following steps, as shown in Fig. 11(a).

- i Operations on networks via union of F_1 and F_2 into a single network R_o .
- ii Determine the change in correlation (metabolomic change) over the network R_o , based on a pair of filtered networks. To determine whether two correlations are significantly different; first, transform the correlations into fisher Z-scores; second, calculate the standard error of the difference of the Z-scores, then calculate the ratio between the difference and the standard error; finally, compare this ratio to a standard normal distribution. The p-values for each change of correlations are obtained after the calculation²⁶. The output of this pipeline is a directed network R_o with labelled edges with their own correlation changes (metabolomic changes). The network R_o can be divided into two subnetworks, the first subnetwork R'_o with significantly different correlations (p-value < 0.05 or label contains the "?" mark) and the second subnetwork R''_o with non-significantly different correlations (p-value ≥ 0.05 and labels with identical letters). The labeling of each

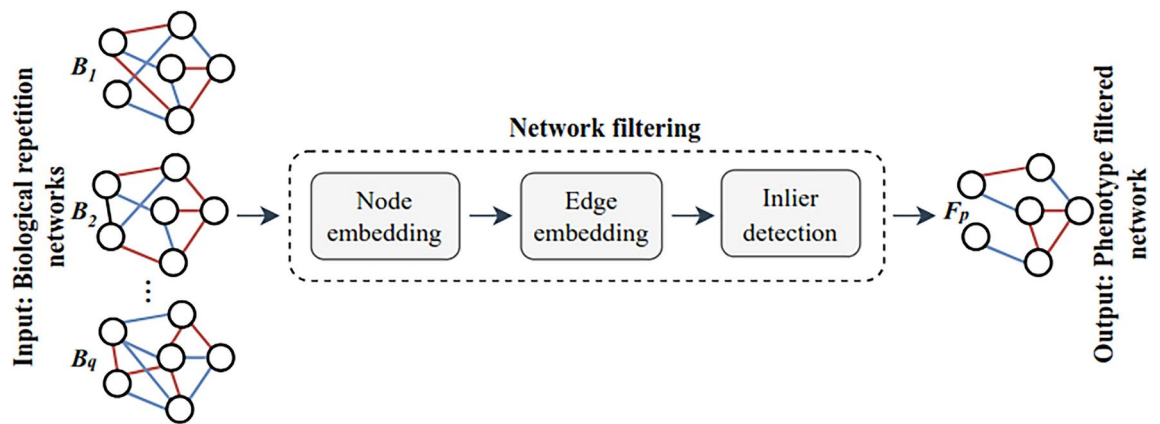
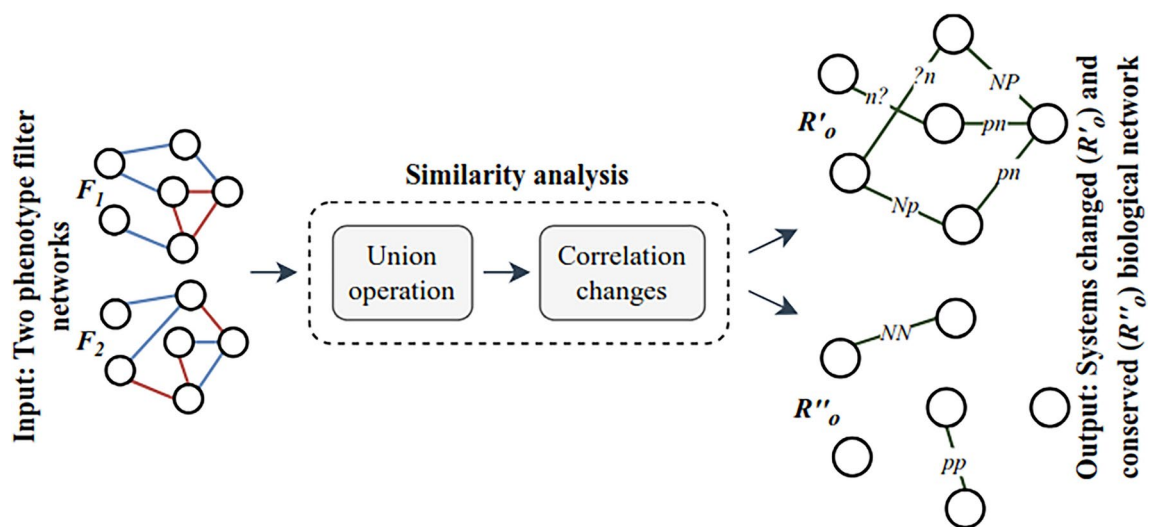
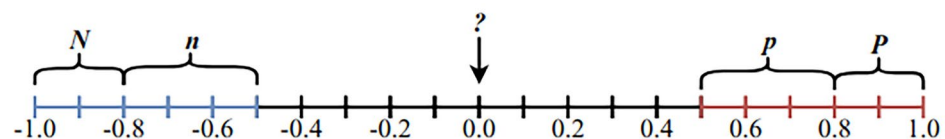


Fig. 10. Pipeline 2: Network filtering. Note: The edges in red color have positive correlation and the edges in blue color have negative correlation.



(a) Similarity analysis.



(b) Correlation scale and labels.

Fig. 11. Pipeline 3: Similarity analysis. Note: the “?” mark means that there is no correlation.

edge $(v_i, v_j) \in R_o$ is obtained according to a threshold (th) and the weights $w_{1,i,j} \in F_1$ and $w_{2,i,j} \in F_2$. We consider a $th = 0.8$, then, the high negative correlation (N) is in $[-1.0, -th]$, the moderate negative correlation (n) is in $[-th, -0.5]$. In contrast, the moderate positive correlation (p) is in $[0.5, th]$, the high positive correlation (P) is in $[th, 1.0]$. For example, given an edge (v_1, v_2) with weights $w_{1,1,2} = 0.6$ and $w_{2,1,2} = -0.9$, then, according to the scale in Fig. 11(b), the label for the edge is pN ($v_1 \xrightarrow{pN} v_2$).

In summary, all steps are visualized in Fig. 12, from the network generation of the biological repetitions with their respective analytical repetitions, to obtaining the biological network systems with their respective metabolomic changes. The toy example in Fig. 12 has three phenotypes, P_1, P_2, P_3 with P_2 considered as the control. These phenotypes pass through the network generation (pipeline 1), after that, they are filtered (pipeline 2) and finally, similarity analysis (pipeline 3) to identify significant changes in the biological networks, it is performed for each

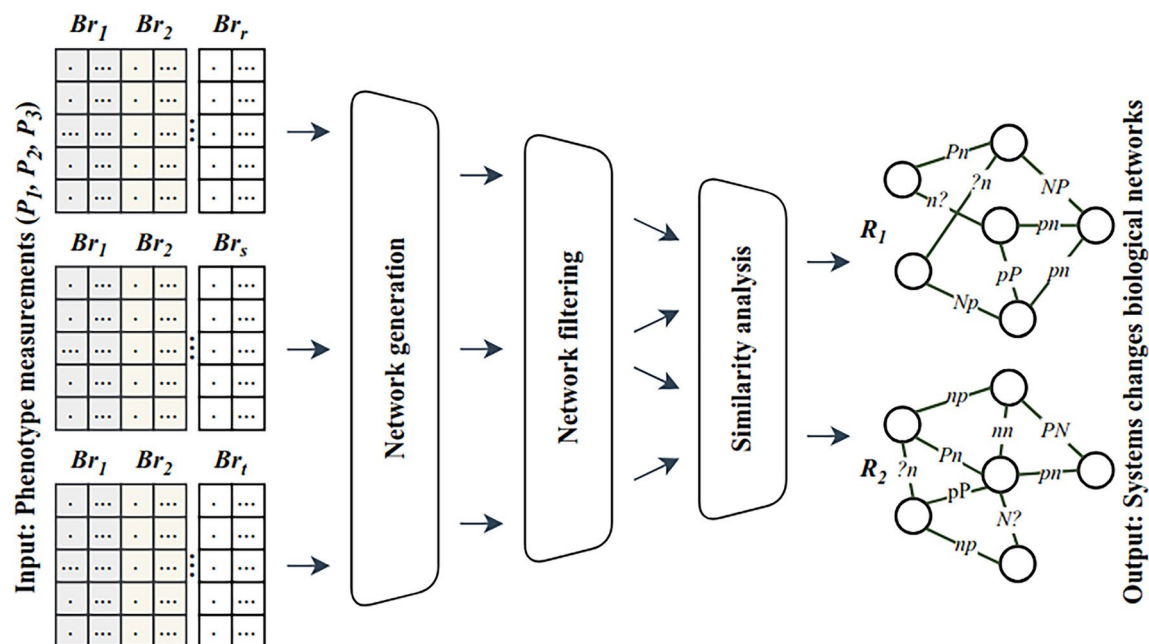


Fig. 12. Our method with a toy example. Note: $Br_1, Br_2, \dots, Br_s$ are **B**iological **r**epetitions.

phenotype regarding the control, in this case P_1 with P_2 and P_3 with P_2 . The entire pipeline was implemented in a web application called GEMNA (Supplementary Fig. S1 and S2).

Discussion

Due to the large datasets produced in traditional mass spectrometry-based experiments, specialized software is always required to identify better and interpret the results. The current metabolomics software uses normalization tools to eliminate confounding factors, particularly the so-called batch effects. Normalizing the peaks is usually performed using a regression curve (e.g., LOESS) or Random forest (SERRF) algorithm to reduce these co-founding factors. These normalization techniques allow for the reduction of batch variations or amplitude drifts. However, this is not always enough. More advanced software uses coefficient of variations between analytical repeats or peak shape profiles. This strategy assumes that MS noise is a mixture of “real (unique)” signals and “not interesting” signals. The latter combines redundant signals (i.e., fragments or clusters of the unique signals), contaminants, and electronic noise. Therefore, since contaminants and electronic noise are not strongly correlated to “real (unique)” signals, these signals could be filtered using a combination of embedding algorithms.

In addition, GEMNA minimizes MS noise by removing those signals whose interaction with the rest (correlation network) has the most variability. Hence, we are improving the output data quality by zooming in on the signals with the most robust interactions. Furthermore, GEMNA also facilitates data interpretation for non-MS experts by introducing a graphic user interface with simple data visualization tools. Hence, once the most efficient parameters have been identified for building the embeddings (pipeline 1), and subsequently, the network was filtered from the signals with the highest variability (pipeline 2), the software allows the researcher to visualize the data under different formats.

For example, in the similarity analysis for different phenotypes of *S. cerevisiae* measured with ESI-MS⁶, the differences between the metabolic networks formed by the WT (control) and PFK1 properties can easily be observed in Fig. 5(a). Here, it is noted that for the WT phenotype, there was a strong negative correlation (N) between the metabolites *Fructose 1; 6-bisphosphate* and *Phosphoenol pyruvate* as it would be expected. This correlation significantly differs from the one observed in the PFK1 system, where one PFK enzyme is removed, while it remains slightly changed in the ZWF1 system, i.e., the algorithm shows that the strong negative correlation stays, but the magnitude of the correlation is (statistically) significantly different. On the Leaf dataset measured using an ESI-MS instrument⁷, the Fig. 6 shows ratio of change between warming-treated and WT plants. The zoom version of the figure highlights the signals with the highest variation, among them are selected amino acids as reported in⁷. In summary, the software is an open-source web application, which is compatible with any mass spectrometry experiments and can contribute non-specialists to better filtrate their data from MS noise and visualize the results in a friendly user environment.

Another advantage of GEMNA is its ability to improve clustering-based analysis, such as PCA. Figure 7, and Table 7 showcases the results of GEMNA (*silhouette score* = 0.409) vs. the traditional approach (*silhouette score* = −0.004) for noise filtration after peak normalization in a headspace analysis of Mentos using a GC-MS instrument. Moreover, after the network filtering pipeline over Mentos, GEMNA on average obtains 301 nodes (metabolites), while the statistical approach only obtains 215. As in many untargeted metabolomics systems, PCA is difficult to interpret due to the presence of confounding factors, in particular, the

so-called batch effects and MS noise. Here, the graph neural network (graph embedding AI) can automatically choose relevant signals and produce a better PCA to the one obtained by manually choosing the signals of interest after removing “noisy peaks” based on their CV% and peak shape.

Conclusion

We propose GEMNA, a novel open-source tool for metabolic study based on node embeddings, edge embeddings, and anomaly detection. GEMNA allows filtering raw data and finds significant metabolomic changes between two or more metabolic networks. Experiments report that in network filtering, on average 90% of edges in common are kept, the others are removed by the anomaly detection algorithm. The runtime is related to the dimension of the embeddings, i.e., a smaller dimension needs a shorter runtime.

In conclusion, the most competitive models were LVGAE and ARGVA, the best network variation was *many-to-many*, and the dimension of the embeddings was 3. In addition, the minimum requirement for the good performance of these models is to have two or more phenotypes, which one of them is the control. Furthermore, each phenotype (and control) consists of at least two biological repetitions, and each with at least 3 measurements.

Data availability

The MS files for the Mutant and Leaf (*Aristolochia chilensis*) datasets can be found in their original articles ^(6,7). The MS files for the Mentos dataset can be downloaded from the following link: <https://drive.google.com/drive/folders/11jFi1AxMmjHFPqETv9VLPyEg9nVTZVna>

Received: 30 August 2024; Accepted: 22 November 2024

Published online: 28 November 2024

References

- Liebal, U. W., Phan, A. N., Sudhakar, M., Raman, K. & Blank, L. M. Machine learning applications for mass spectrometry-based metabolomics. *Metabolites* **10**, 243 (2020).
- Fan, S. et al. Systematic error removal using random forest for normalizing large-scale untargeted lipidomics data. *Anal. Chem.* **91**, 3590–3596 (2019).
- Bahado-Singh, R. O. et al. Artificial intelligence and the analysis of multi-platform metabolomics data for the detection of intrauterine growth restriction. *PLoS ONE* **14**, e0214121 (2019).
- Sauer, U. & Zamboni, N. From biomarkers to integrated network responses. *Nat. Biotechnol.* **26**, 1090–1092 (2008).
- Patti, G. J., Yanes, O. & Siuzdak, G. Metabolomics: the apogee of the omics trilogy. *Nat. Rev. Mol. Cell Biol.* **13**, 263–269 (2012).
- Buettner, F. et al. Non-targeted metabolomic approach reveals two distinct types of metabolic responses to telomerase dysfunction in *S. cerevisiae*. *Metabolomics* **13**, 1–10 (2017).
- González-Teuber, M., Palma-Onetto, V., Aguirre, C., Ibáñez, A. J. & Mithöfer, A. Climate change-related warming-induced shifts in leaf chemical traits favor nutrition of the specialist herbivore *battus polydamas archidamas*. *Front. Ecol. Evol.* **11**, 1152489 (2023).
- Alvarez-Mamani, E., Dechant, R., Beltran-Castañón, C. A. & Ibáñez, A. J. Graph embedding on mass spectrometry- and sequencing-based biomedical data. *BMC Bioinformatics* **25**, 1 (2024).
- Perozzi, B., Al-Rfou, R. & Skiena, S. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 701–710 (2014).
- Grover, A. & Leskovec, J. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 855–864 (2016).
- Ribeiro, L. F., Saverese, P. H. & Figueiredo, D. R. struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 385–394 (2017).
- Wang, D., Cui, P. & Zhu, W. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 1225–1234 (2016).
- Cao, S., Lu, W. & Xu, Q. Deep neural networks for learning graph representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30 (2016).
- Kipf, T. N. & Welling, M. Variational graph auto-encoders. Preprint at [arXiv:1611.07308](https://arxiv.org/abs/1611.07308) (2016).
- Velčković, P. et al. Deep graph infomax. arXiv preprint [arXiv:1809.10341](https://arxiv.org/abs/1809.10341) (2018).
- Pan, S. et al. Adversarially regularized graph autoencoder for graph embedding. Preprint at [arXiv:1802.04407](https://arxiv.org/abs/1802.04407) (2018).
- Salha, G., Hennequin, R. & Vazirgiannis, M. Simple and effective graph autoencoders with one-hop linear models. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part I*, 319–334 (Springer, 2021).
- Hamilton, W. L. Graph representation learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* **14**, 1–159 (2020).
- Smith, C. A. et al. Metlin: a metabolite mass spectral database. *Ther. Drug Monit.* **27**, 747–751 (2005).
- Chen, D. et al. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. *Proc. AAAI Conf. Artif. Intell.* **34**, 3438–3445 (2020).
- Kazemi, S. M. *Dynamic Graph Neural Networks* 323–349 (Springer Nature Singapore, 2022).
- Kipf, T. N. & Welling, M. Semi-supervised classification with graph convolutional networks. Preprint at [arXiv:1609.02907](https://arxiv.org/abs/1609.02907) (2016).
- Dwivedi, V. P., Luu, A. T., Laurent, T., Bengio, Y. & Bresson, X. Graph neural networks with learnable structural and positional representations. Preprint at [arXiv:2110.07875](https://arxiv.org/abs/2110.07875) (2021).
- Dwivedi, V. P. et al. Benchmarking graph neural networks. *J. Mach. Learn. Res.* **24**, 1–48 (2023).
- Li, Z., Zhao, Y., Botta, N., Ionescu, C. & Hu, X. Copod: copula-based outlier detection. In *2020 IEEE International Conference on Data Mining (ICDM)* (ed. Li, Z.) 1118–1123 (IEEE, 2020).
- Snedecor, G. W. G. W. & Cochran, W. G. W. *Statistical methods / [by] George W. Snedecor, William G. Cochran*. 7th Edn. (Iowa State University Press, Ames, Iowa, 1980).

Acknowledgements

E.A.M. doctoral studies are funded by *Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica* (CONCYTEC), and *Fondo Nacional de Desarrollo Científico, Tecnológico y de Innovación Tecnológica* (FONDECYT), under contract No. 174-2020-FONDECYT “Doctoral Programs in Peruvian Universities”. A.J.I. thank to *The Max Planck Partner Group* (Max Planck Institute for Chemical Ecology-Jena), and the *Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica* (CONCYTEC-Prociencia funding call E041-2024-01; N°

PE501086715-2024-PROCIENCIA) for their financial support. We also thank Prof. Dr. Marcia González-Teuber (Pontificia Universidad Católica de Chile) for the *Aristolochia chilensis* leaf dataset, and PD Dr. Reinhard Dechant (Calico) & Dr. Madina Mansurova (PUCP) for their valuable recommendations and review of this work.

Author contributions

E.A.M. and A.J.I. designed the experiments, E.A.M. ran the experiments, E.A.M., F.B., C.A.B.C., and A.J.I. wrote the main manuscript text, and E.A.M. prepared figures and tables. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1038/s41598-024-80955-5>.

Correspondence and requests for materials should be addressed to A.J.I.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024