PIntMF: Supplementary Materials

March 8, 2021

1 Initialization of the algorithm

Often in NMF algorithms (Lee and Seung, 1999), the matrices are initialized by non-negative random values. Here, we evaluated four kinds of initialization for $\mathbf{W}$ or $\mathbf{H}$.

SVD (Singular Value Decomposition)

SVD is also a matrix factorization technique with constraints. The SVD is defined as:

$$SVD(\mathbf{X}^k) = \mathbf{USV}^T \tag{1}$$

Where $\mathbf{U}, \mathbf{S}$ and $\mathbf{V}$ are of dimensions $n \times n$, $n \times J_k$ and $J_k \times J_k$. Matrix $\mathbf{S}$ is a diagonal matrix. If r is the rank of matrix $\mathbf{X}^k$, therefore $\mathbf{S}$ has r non-zero entries.

SVD provides the best low-rank linear approximation of the original matrix $\mathbf{X}^k$ if we keep only $P \leq r$ singular values. The matrices $\mathbf{U}$ and $\mathbf{V}$ are also reduced to produce matrices $\mathbf{U}_P$ and $\mathbf{V}_P$, respectively. In our case, we initialize $\mathbf{H}^k$ with $\mathbf{V}$.

Hierarchical clustering

For this initialization, we perform a hierarchical clustering on each block and we keep the partitions in P clusters. Then, average profiles from the clusters are computed. For each block k:

1. Compute Hierarchical clustering with Ward's method
2. Cut the hierarchical clustering at P
3. Obtain P clusters denoted $\mathcal{C}_1, \ldots, \mathcal{C}_P$
4. $\mathbf{h}^k_{p\bullet} = \frac{1}{Card(\mathcal{C}_p)} \sum_{i \in \mathcal{C}_p} \mathbf{x}^k_{i\bullet}$

Random

We sample P profiles at random and $\mathbf{H}^k$ are initialized with these profiles.

SNF

For this initialization, we initialize $\mathbf{W}$ with the clustering in P clusters from the algorithm SNF (Similarity network fusion (Wang *et al.*, 2014)). $\mathcal{C}_p$ denotes the cluster p from SNF. Thus,

$$w_{ip} = \begin{cases} 1 & \text{if } i \in \mathcal{C}_p \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

This initialization has the advantage to take into account simultaneously the K blocks of the analysis.

1.1 Best initialization method

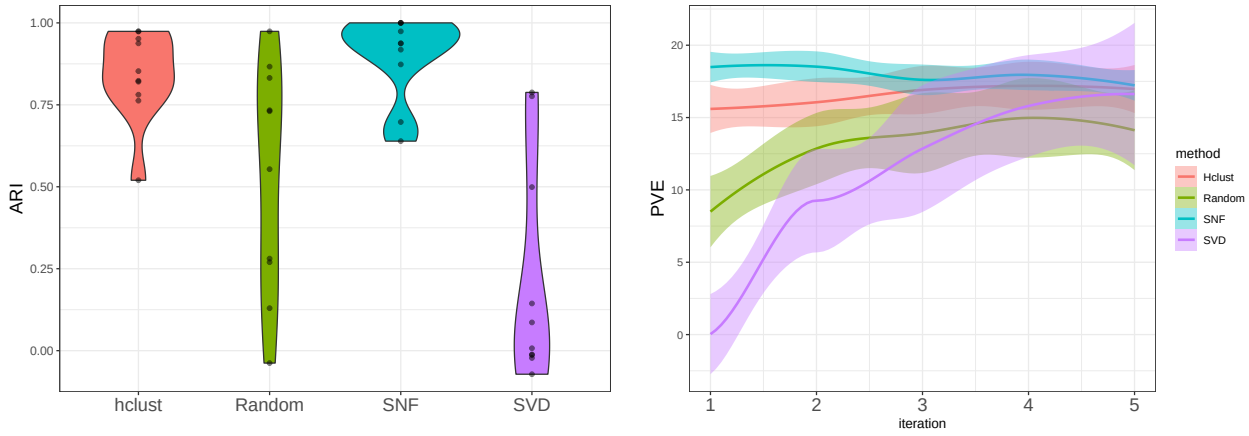
For non-negative matrix factorization methods, the first step consists in initializing either the matrix $\mathbf{W}$ or in our case the matrices $\mathbf{H}^k$. Several kinds of initialization were compared: Hclust, SVD, random, and SNF (see section 1 for more details). To evaluate the influence on the final results of the type of initialization, 20 datasets were simulated with R package CriMMix previously developed by our group Pierre-Jean *et al.*

(2019). Each dataset is composed of three blocks (under Gaussian, Binary, Beta-Like distributions with respectively 100, 50 and 500 variables in each block). Four unbalanced groups have been simulated with respectively 5, 10, 20 and 25 individuals.

First, ARIs between the clustering and the true clustering were computed for each type on initialization at the end of the algorithm (Fig. 1a). Second, we compute the PVE at each iteration to monitor the convergence of the algorithm (Fig. 1b).

The initialization with the SNF reaches higher values of both ARI and PVE. Indeed, ARI computed with SNF initialization is mainly close to 1 and PVE reaches fastly 20%, for the other types of initialization a larger number of iterations is necessary to reach this level of PVE. Hclust is the second best method to initialize the algorithm, however it does not use all the blocks jointly. SVD is clearly not stable as well as random initialization.

For all the following analyses, SNF initialization was used.



(a) ARI

(b) PVE

Figure 1: Performance of the 4 tested types of initialization (hierachical clustering, Random, SNF and SVD). (a) Evaluation of the final clustering with adjusted Rand Index (ARI), and (b) Percentage of variation explained (PVE)

2 Selection of the correct number of latent variables

2.1 The uncorrelated simulated dataset

For 25 simulations per benchmark, we ran the algorithm PIntMF for various numbers of latent variables (2 to 7). Then, we computed the different criteria (BIC, PVE and the cophenetic coefficient). We also added the RSS (Figures 2, 3, 4, 5).

On this dataset, the best criteria to choose the correct number of latent variables seem to be the cophenetic coefficient and the PVE.

For the benchmark 6 (with only two clusters), it seems that the criteria do not allow to choose the correct number of clusters. The model selection is difficult in the case where we simulated only two clusters in the data.

2.2 Cophenetic coefficient, PVE, MSE on realistic simulations

According to the three criteria, the best number of latent variables for PIntMF is 2. Indeed, this number minimizes the MSE criterion, adding more latent variables in the model does not really increase the PVE,

and the cophenetic coefficient decreases just after 2.

2.3 Cophenetic coefficient, PVE, MSE on real datasets

By looking at the MSE and PVE, we choose the best number of latent variables (P=5) for the Glioblastoma dataset (Fig. 7). The cophenetic coefficient is not stable after p=6.

By looking at the MSE, the best number of latent variable is 2 for the BXD dataset (Fig. 8). The PVE slightly increases from p=2 to 5 latent variables, therefore we can consider that adding new latent variables is not useful. For the cophenetic coefficient, the best is for P=2 and 3.

3 Fast computation for H matrix

We optimized the computation of the **H** matrix because it could be huge. Indeed, **H** is the matrix that represents the variable profiles. The number of variables could range from one hundred to one billion.

Therefore, for the computation of the $\mathbf{H}^k$ matrices, several packages `R` have been tried (`ncvreg` (Breheny and Huang, 2011), `glmnet` (Tibshirani, 1996), `biglasso` (Zeng and Breheny, 2017) and `quadrupen` (Grandvalet *et al.*, 2018)). All these packages proposed cross-validation (using 5 folds) to choose the optimal value of the lasso penalty.

The fastest method is clearly `glmnet`, therefore we used this package to solve $\mathbf{H}^k$ matrices 9).

4 Performance of PIntMF for variable selection

Mean and standard deviation of AUROC were computed for PIntMF and the other 5 methods (Table 1).

5 Details on simulations OmicsSIMLA

Expression The tool OmicsSIMLA creates a file containing individual information, such as family ID, individual ID, father ID, mother ID, sex, and affection status. Then, the file containing the gene expression data (i.e., read counts) has various types of genes: EE (equivalently expressed genes between cases and controls), DE (differentially expressed genes between cases and controls), eQTL_EE (equivalently expressed genes influenced by eQTL), eQTL_DE (differentially expressed genes influenced by eQTL), eQTM_EE (equivalently expressed genes influenced by eQTM), and eQTM_DE (differentially expressed genes influenced by eQTM). All columns labeled with DE correspond to the expression of the genes generated that are differentially expressed between case and controls. Here, 100 genes have been generated with DE and 3 genes are influenced by the eQTM.

Methylation The file containing methylation data is composed of the methylated read counts and total read depth for each CpG (separated by comma). We computed the percentage of methylation values from this table with an home-made R script. A file indicates which probes are DM (differentially methylated) and the eQTM status is also generated.

For the five methods, we transform the beta values to M-values by using this formula:

$$M_i = log2((\beta_i + \epsilon)/(1 - (\beta_i + \epsilon))$$

Note that, some beta values are exactly one or zero. To avoid getting infinite values for M, the same value $\epsilon = 0.001$ is added for each beta values before applying the formula.

Proteomics They used the mass-action kinetic action model (Guoshouteo *et al.*, 2015) to simulate proteomics data at a certain time point incorporating the gene expression data. No proteins are simulated as "true positive" variables to drive the clusters. Therefore, we do not evaluate the performance of the methods in terms of variable selection on this block.

Finally, the dataset is composed of 100 individuals and three omics blocks with 12102 genes with expression data, 2248 CpG probes, and 12103 proteins respectively.

Benchmark	Dataset	CIMLR	iClusterPlus	iNMF	MoCluster	PIntMF	SGCCA
Benchmark1	Beta-like	0.93(0.01)	0.84(0.04)	0.29(0.01)	**0.97(0.01)**	0.96(0.00)	0.95(0.06)
	Binary	0.93(0.06)	0.45(0.03)	0.99(0.01)	**1.00(0.01)**	0.98(0.02)	0.95(0.06)
	Gaussian	0.99(0.02)	**1.00(0.01)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.98(0.05)
Benchmark2	Beta-like	0.92(0.02)	0.83(0.05)	0.29(0.01)	**0.96(0.02)**	**0.96 (0.00)**	0.94(0.006)
	Binary	0.92(0.05)	0.46(0.03)	**0.99(0.01)**	**0.99(0.01)**	0.98(0.02)	0.93(0.07)
	Gaussian	0.97(0.03)	0.96(0.04)	0.99(0.01)	**1.00(0.00)**	**1.00(0.00)**	0.96(0.06)
Benchmark3	Beta-like	0.94(0.01)	0.85(0.04)	0.29(0.01)	**0.98(0.01)**	0.96(0.00)	0.94(0.04)
	Binary	0.91(0.04)	0.52(0.05)	0.98(0.02)	**0.99(0.01)**	**0.99(0.02)**	0.91(0.07)
	Gaussian	**1.00(0.01)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.97(0.06)
Benchmark4	Beta-like	0.87(0.01)	0.69(0.05)	0.32(0.01)	0.87(0.01)	**0.88(0.01)**	0.82(0.07)
	Binary	0.93(0.04)	0.48(0.03)	0.98(0.01)	**0.99(0.01)**	**0.99(0.02)**	0.89(0.09)
	Gaussian	0.99(0.01)	0.99(0.01)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.92(0.09)
Benchmark5	Beta-like	0.94(0.01)	0.87(0.04)	0.29(0.01)	**0.98(0.00)**	0.96(0.00)	0.97(0.00)
	Binary	0.95(0.02)	0.45(0.02)	0.99(0.01)	**1.00(0.01)**	0.99(0.01)	0.99(0.01)
	Gaussian	**1.00(0.01)**	0.99(0.02)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**
Benchmark6	Beta-like	**1.00(0.00)**	0.85(0.21)	0.25(0.00)	0.98(0.00)	**1.00(0.00)**	0.97(0.01)
	Binary	**1.00(0.00)**	0.49(0.03)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.92(0.03)
	Gaussian	**1.00(0.01)**	0.97(0.06)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.99(0.04)
Benchmark7	Beta-like	**1.00(0.00)**	0.95(0.07)	0.25(0.00)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**
	Binary	**1.00(0.00)**	0.49(0.02)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.98(0.02)
	Gaussian	**1.00(0.00)**	**1.00(0.02)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.01)**
Benchmark8	Beta-like	**1.00(0.00)**	0.98(0.04)	0.26(0.00)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**
	Binary	**1.00(0.00)**	0.47(0.03)	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.99(0.03)
	Gaussian	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	**1.00(0.00)**	0.99(0.03)

Table 1: Mean (standard deviation) of AUROC of intNMF, SGCCA, MoCluster, iClusterPlus, CIMLR and PIntMF methods on simulated data

6 Jackknife

Jackknife was performed to evaluate the stability of variable selection. To perform this technique, we run the model PIntMF on the data without one individual at each step. Therefore, we obtain n datasets containing $n-1$ individuals on which we apply the method.

The stability of the selected variables for Binary, Gaussian, methylation and expression datasets seems to be robust (Figures 10a and 10b). For proteins and for beta-like data, the bootstrap reveals that some selected variables are not stable and we could remove false-positives by adding this step.

7 Additional results on BXD dataset

7.1 Clustering

8 Additional results on Glioblastoma dataset

8.1 Comparison of selected variables

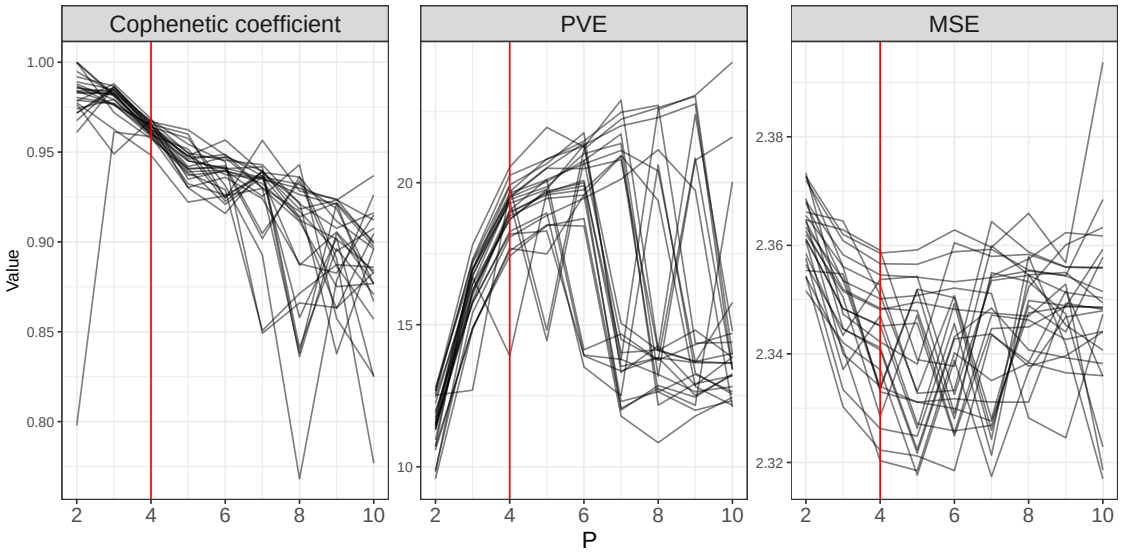
We explore the selected variables on each dataset (Copy number, expression, and methylation). We notice that most selected variables are specific to one component. For instance, for the copy number dataset and the 3rd component of the model, 136 variables are specific to this component. We conclude that these variables are specific to the corresponding cluster and could have an impact on survival. Some variables are also shared between several components (Fig. 11) and could reveal common biological processes.

Data Set	Methods	F-score	ARI
BXD	iClusterPlus	0.97	0.88
	MoCluster	1.00	1.00
	intNMF	0.98	0.94
	SGCCA	0.95	0.82
	CIMLR	1.00	1.00
	PIntMF	1.00	1.00

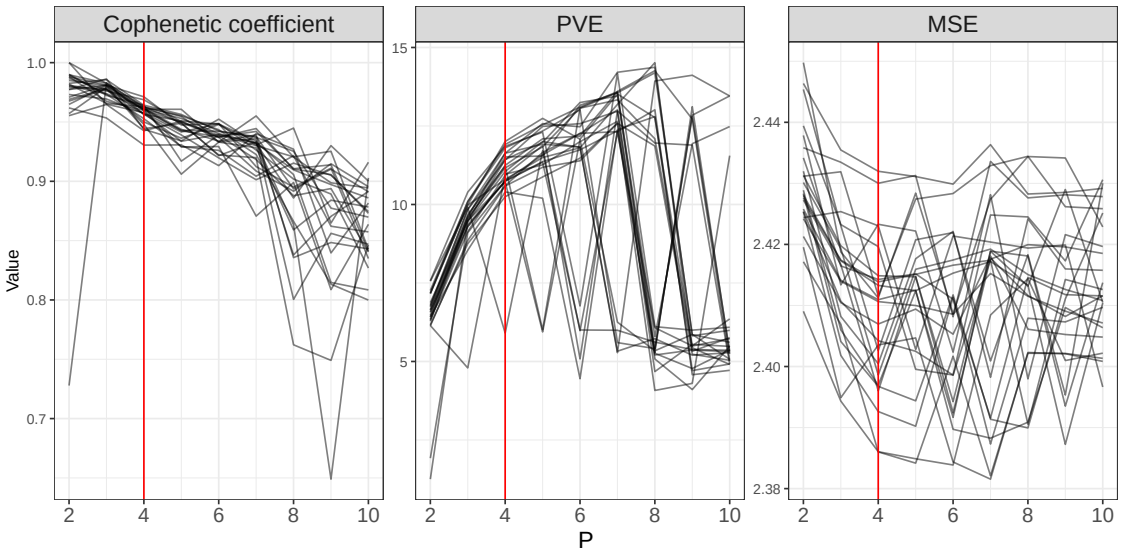
Table 2: Performance evaluation of clustering using ARI and F-measure on BXD data.

8.2 Enrichment

We performed a pathway enrichment analysis with the R package `ClusterProfiler` (Yu *et al.*, 2012) for each component with the union of the selected genes across the three datasets. As the selected genes, some pathways are unique to the components (Fig. 12), this means that some pathways could be linked to survival. All pathways for each component are in table 3.

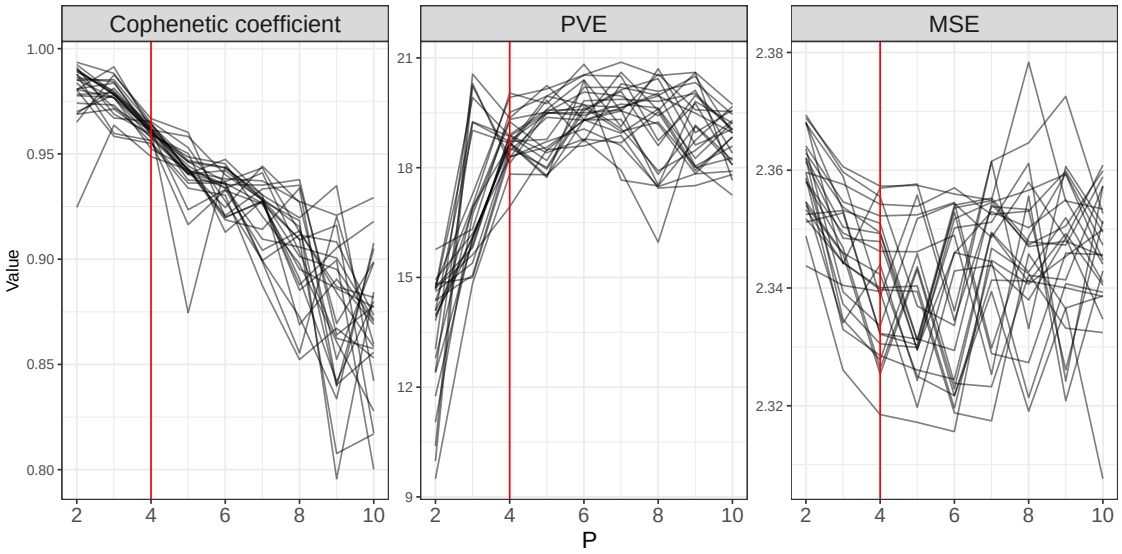


(a) Benchmark 1

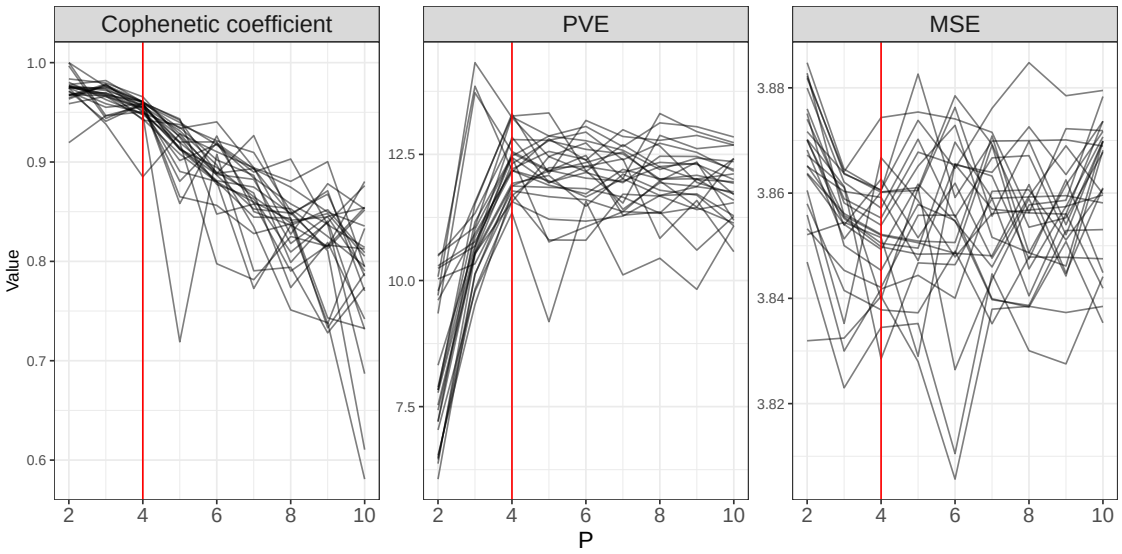


(b) Benchmark 2

Figure 2: Benchmarks 1 and 2: Cophenetic coefficient, PVE, MSE for uncorrelated simulations. The vertical red line represents the simulated number of clusters.

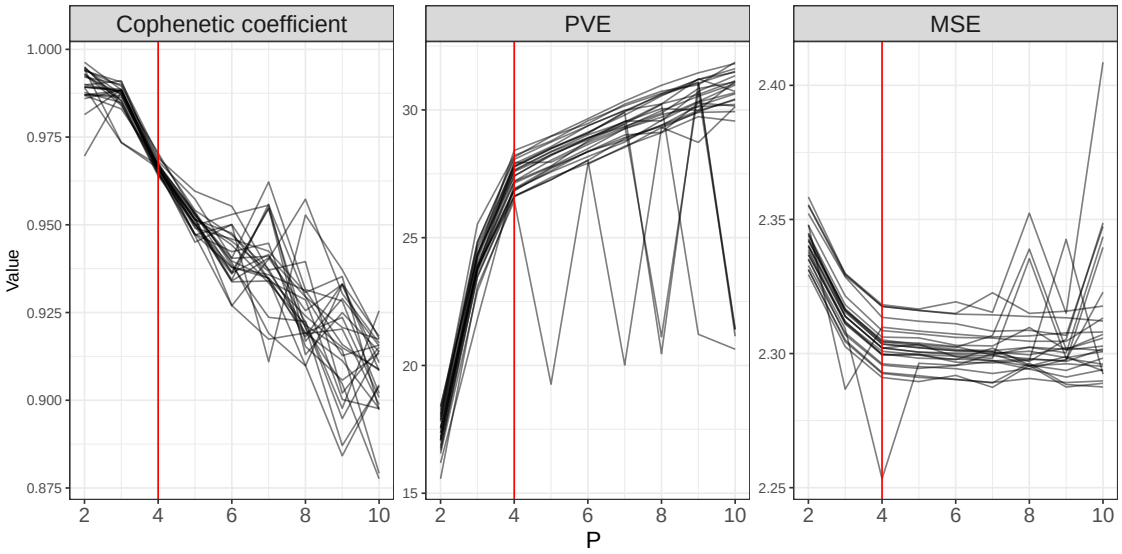


(a) Benchmark 3

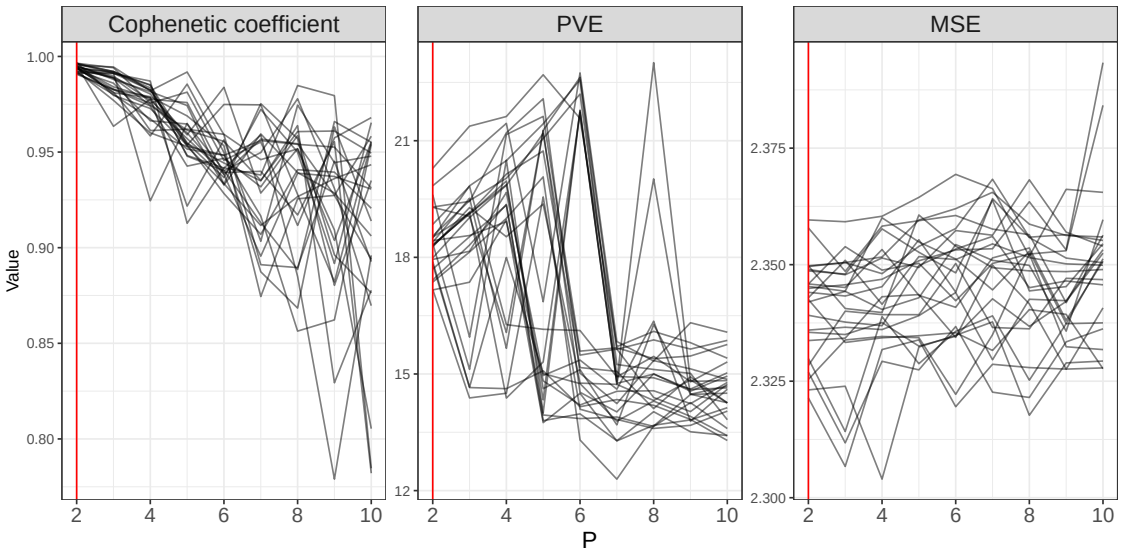


(b) Benchmark 4

Figure 3: Benchmarks 3 and 4: Cophenetic coefficient, PVE, MSE for uncorrelated simulations. The vertical red line represents the simulated number of clusters.

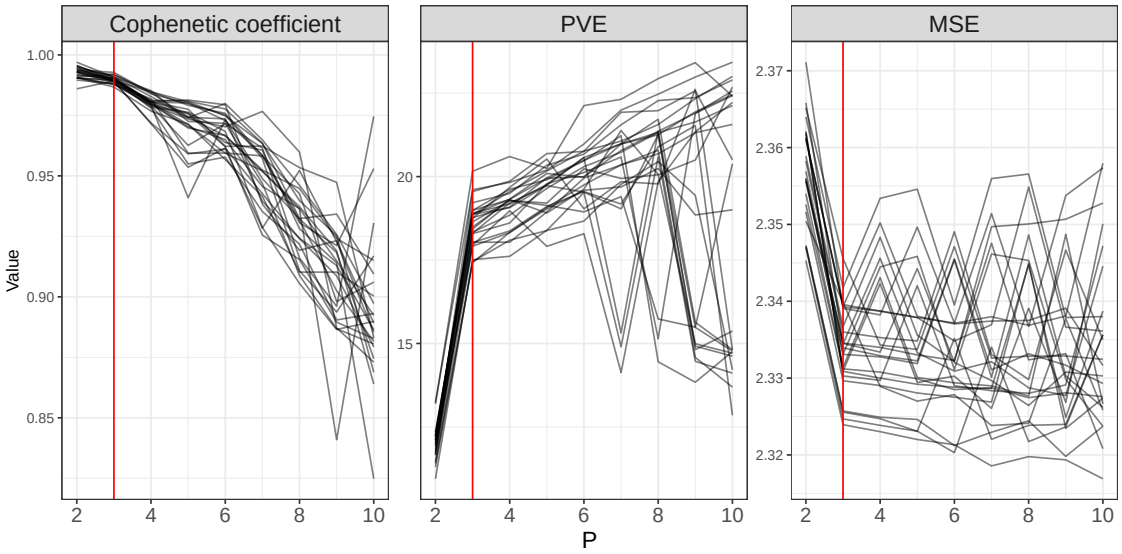


(a) Benchmark 5

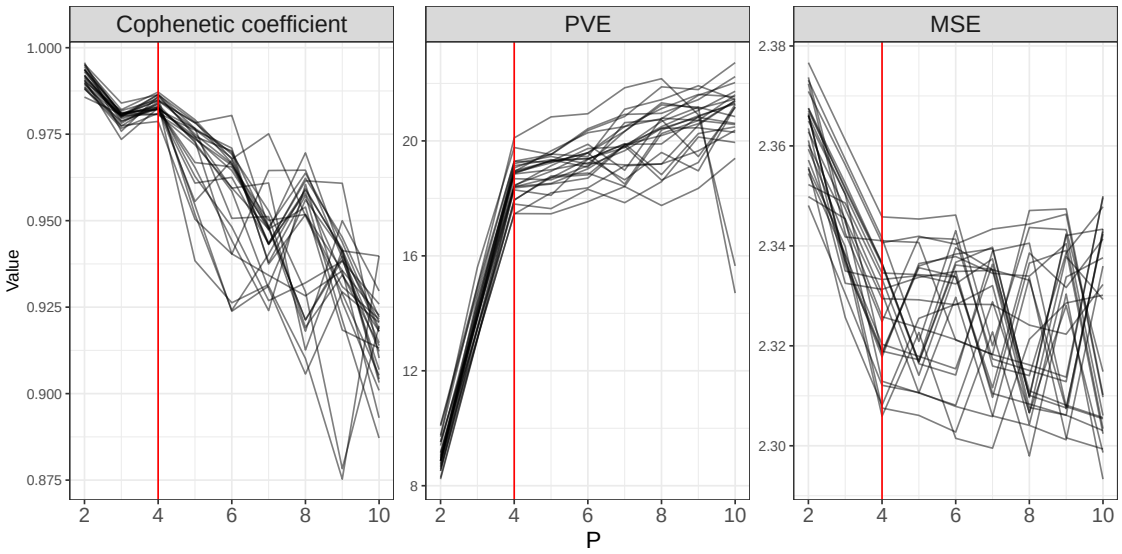


(b) Benchmark 6

Figure 4: Benchmarks 5 and 6: Cophenetic coefficient, PVE, MSE for uncorrelated simulations. The vertical red line represents the simulated number of clusters.



(a) Benchmark7



(b) Benchmark 8

Figure 5: Benchmarks 7 and 8: Cophenetic coefficient, PVE, MSE for uncorrelated simulations. The vertical red line represents the simulated number of clusters.

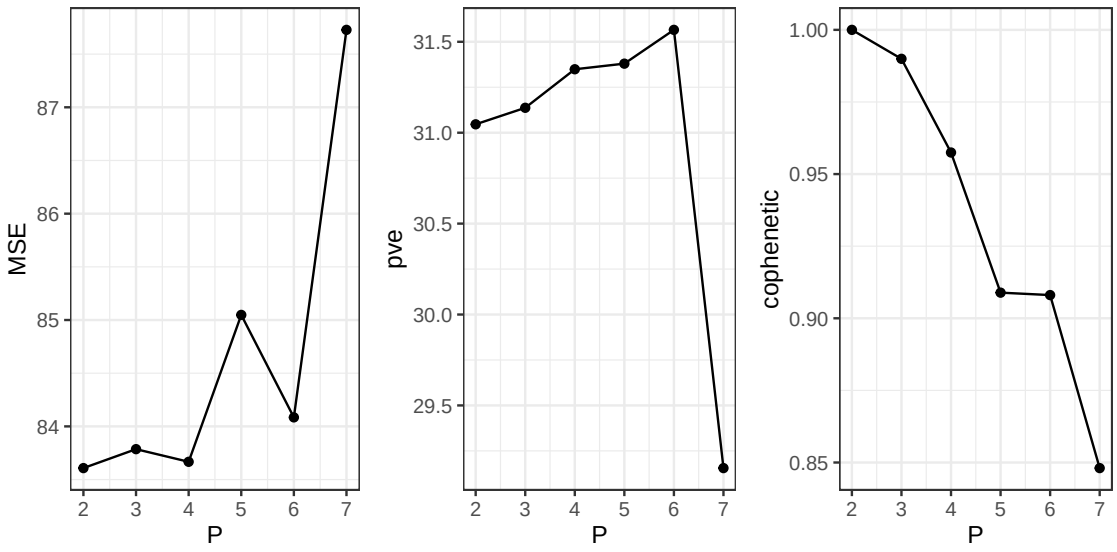


Figure 6: Cophenetic coefficient, PVE, MSE for correlated simulations

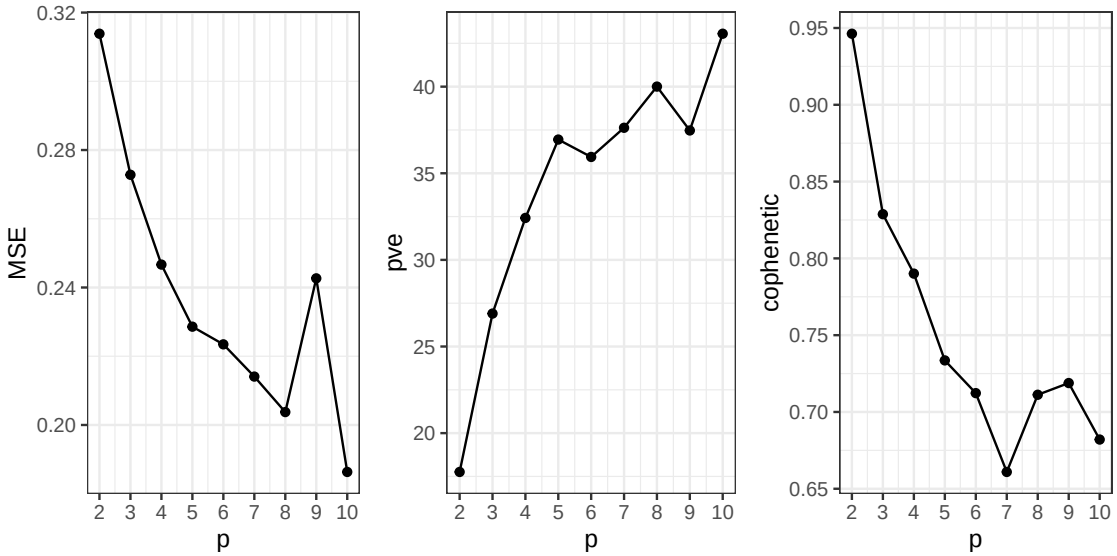


Figure 7: Cophenetic coefficient, PVE, MSE for GBM data

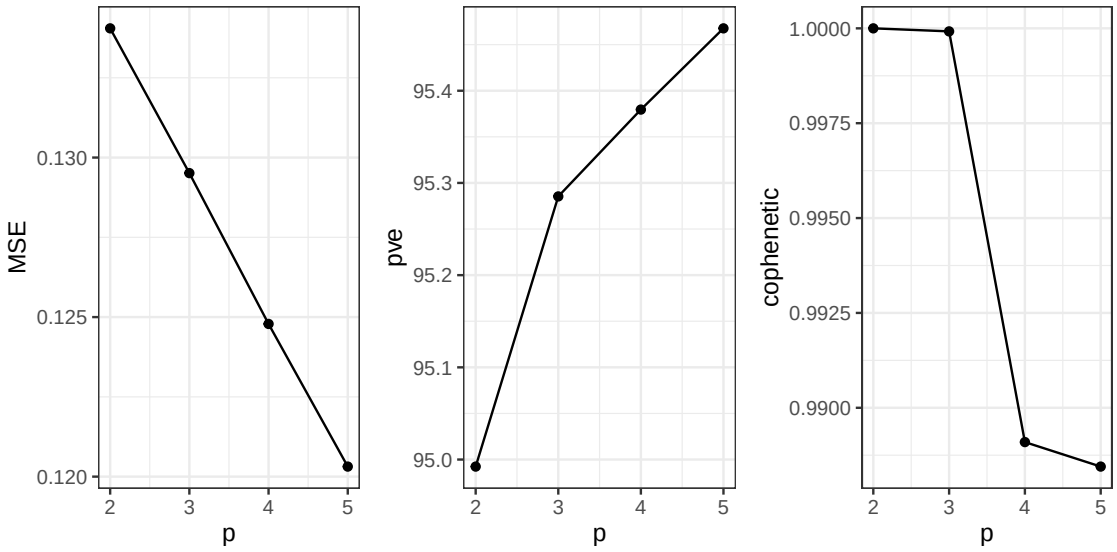


Figure 8: Cophenetic coefficient, PVE, MSE for BXD data

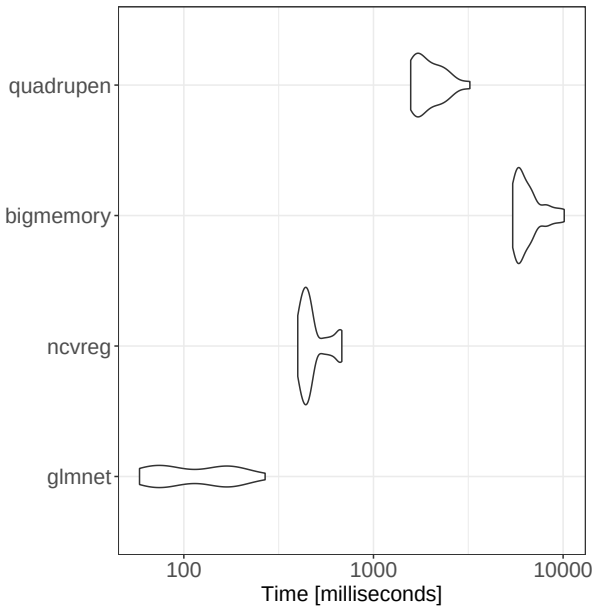
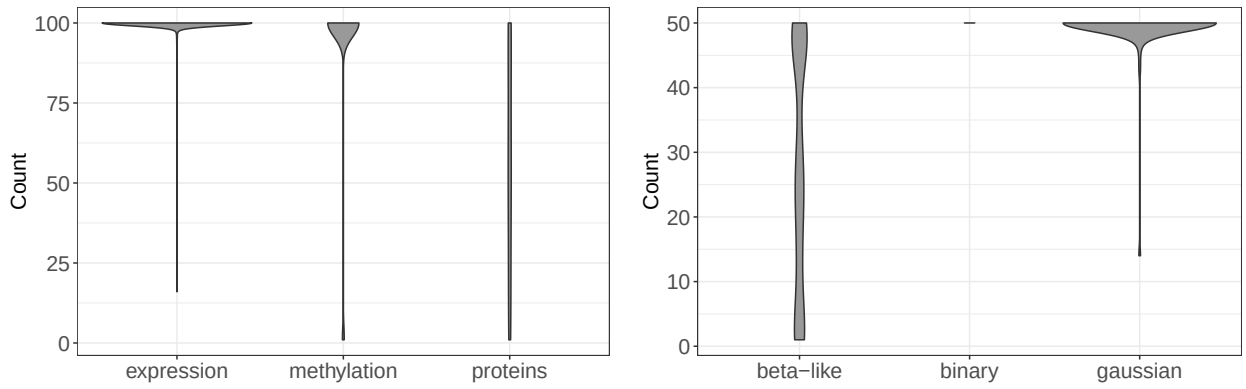


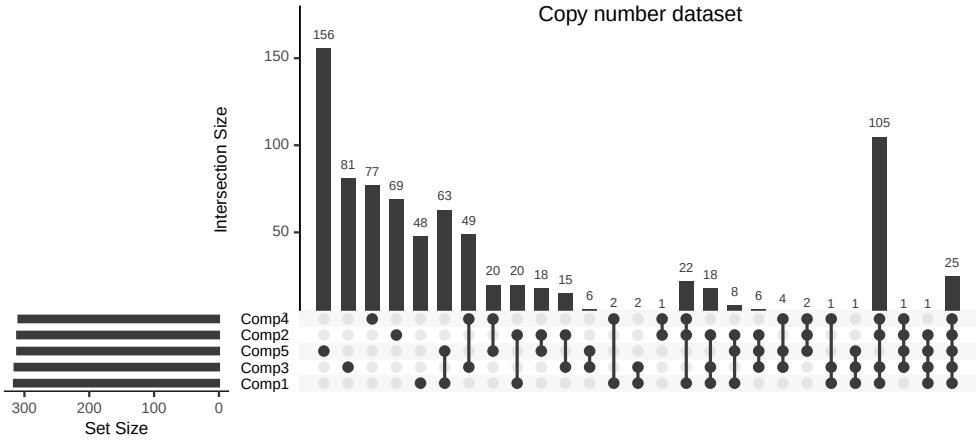
Figure 9: Violin-plot of time computing of **H** matrices for packages glmnet, quadrupen, bigmemory, and ncvreg.



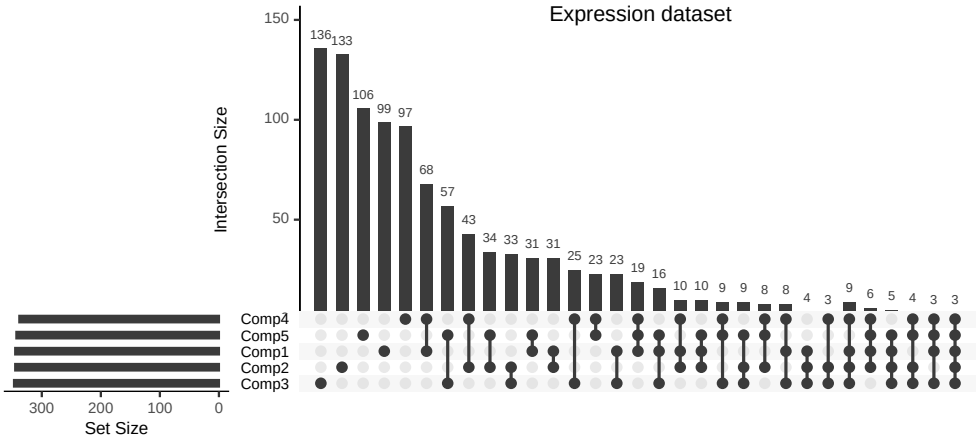
(a) Correlated simulated datasets

(b) Non-correlated simulated datasets

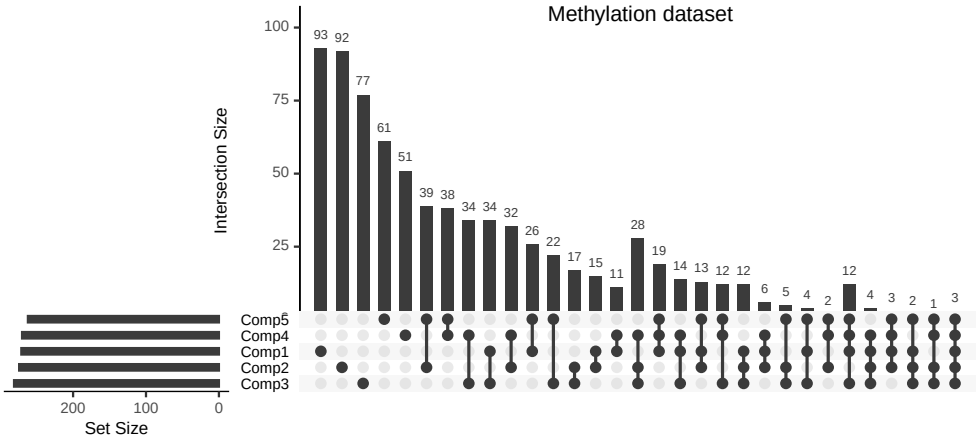
Figure 10: Jackknife on simulated datasets. At each run, one sample is removed, we count the number of times where each variable is selected by the model (coefficient not equal to zero).



(a) Copy Number variables



(b) Expression variables



(c) Methylation variables

Figure 11: GBM: Comparison of selected variables across the latent variables from PIntMF for each dataset.

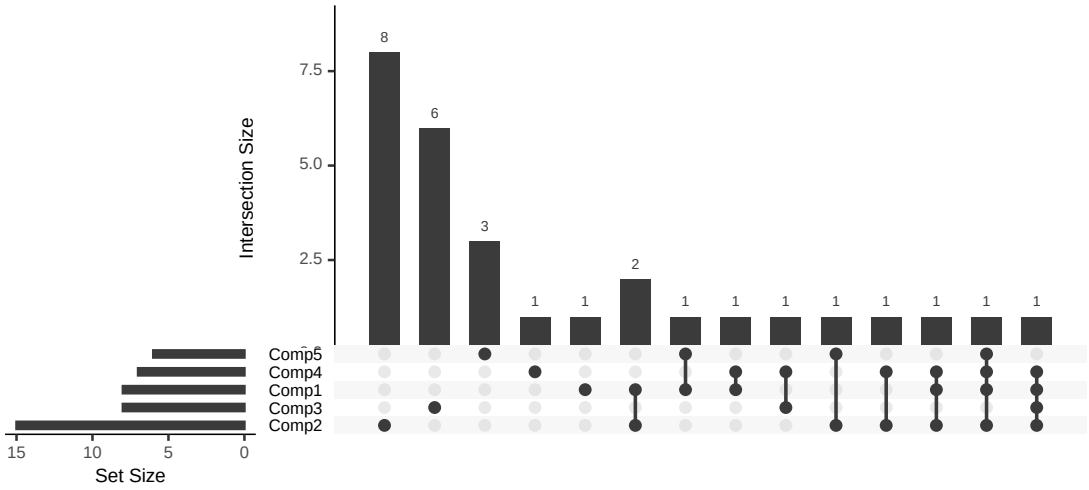


Figure 12: GBM: Comparison of enriched pathways across latent variables.

	Description	p_value	Comp
1	Proteoglycans in cancer	0.00	1
2	Focal adhesion	0.00	1
3	Transcriptional misregulation in cancer	0.00	1
4	Leishmaniasis	0.00	1
5	Chagas disease	0.00	1
6	Amoebiasis	0.00	1
7	Legionellosis	0.00	1
8	Rheumatoid arthritis	0.00	1
9	Focal adhesion	0.00	2
10	AGE-RAGE signaling pathway in diabetic complications	0.00	2
11	Amoebiasis	0.00	2
12	PI3K-Akt signaling pathway	0.00	2
13	Pertussis	0.00	2
14	Cell adhesion molecules	0.00	2
15	Legionellosis	0.00	2
16	Transcriptional misregulation in cancer	0.00	2
17	Rheumatoid arthritis	0.00	2
18	Staphylococcus aureus infection	0.00	2
19	Hematopoietic cell lineage	0.00	2
20	PPAR signaling pathway	0.00	2
21	TNF signaling pathway	0.00	2
22	ECM-receptor interaction	0.00	2
23	NF-kappa B signaling pathway	0.00	2
24	p53 signaling pathway	0.00	3
25	Cell cycle	0.00	3
26	HIF-1 signaling pathway	0.00	3
27	Bile secretion	0.00	3
28	DNA replication	0.00	3
29	Fc gamma R-mediated phagocytosis	0.00	3
30	Mineral absorption	0.00	3
31	Rheumatoid arthritis	0.00	3
32	Cell adhesion molecules	0.00	4
33	Proteoglycans in cancer	0.00	4
34	Rheumatoid arthritis	0.00	4
35	HIF-1 signaling pathway	0.00	4
36	Focal adhesion	0.00	4
37	Ras signaling pathway	0.00	4
38	Transcriptional misregulation in cancer	0.00	4
39	Phospholipase D signaling pathway	0.00	5
40	AGE-RAGE signaling pathway in diabetic complications	0.00	5
41	Chagas disease	0.00	5
42	Focal adhesion	0.00	5
43	Complement and coagulation cascades	0.00	5
44	Glutamatergic synapse	0.00	5

Table 3: Enriched pathway for each component from PIntMF

References

Breheny, P. and Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, **5**(1), 232–253.

Grandvalet, Y., Chiquet, J., and Ambroise, C. (2018). Sparsity by worst-case quadratic penalties. Technical report, arXiv preprint.

Guoshouteo, C. V., Ghosh, D., Kim, S., and Choi, H. (2015). A mass-action-based model for gene expression regulation in dynamic systems. *Integrating Omics Data*, page 362.

Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, **401**(6755), 788.

Pierre-Jean, M., Deleuze, J.-F., Le Floch, E., and Mauger, F. (2019). Clustering and variable selection evaluation of 13 unsupervised methods for multi-omics data integration. *Briefings in bioinformatics*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Wang, B., Mezlini, A. M., Demir, F., Fiume, M., Tu, Z., Brudno, M., Haibe-Kains, B., and Goldenberg, A. (2014). Similarity network fusion for aggregating data types on a genomic scale. *Nature methods*, **11**(3), 333.

Yu, G., Wang, L.-G., Han, Y., and He, Q.-Y. (2012). clusterprofiler: an r package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*, **16**(5), 284–287.

Zeng, Y. and Breheny, P. (2017). The biglasso package: A memory- and computation-efficient solver for lasso model fitting with big data in r. *ArXiv e-prints*.