IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Kernel Joint Non-negative Matrix Factorization for Genomic Data

DIEGO SALAZAR[1], JUAN RIOS[1], SARA ACEROS[1], OSCAR FLOREZ-VARGAS[2], AND CARLOS VALENCIA.[1]

[1]School of Industrial Engineering, University of Los Andes, Bogota, 111711, Colombia.
[2]Division of Cancer Epidemiology and Genetics, National Cancer Institute, Bethesda, MD, 20892, USA.

Corresponding author: Carlos Valencia (e-mail: cf.valencia@uniandes.edu.co)

ABSTRACT The multi-modal or multi-view integration of data has generated a wide range of applicability in pattern extraction, clustering, and data interpretation. Recently, variants of the Non-negative Matrix Factorization (NMF) such as joint NMF (jNMF) have allowed not only to integrate data from different sources but also have facilitated the incorporation of prior knowledge such as the interactions between variables from different sources. However, in both NMF and jNMF the factorization is carried out as a linear system, which does not identify non-linear patterns that are present in most real-world data. Therefore, we propose a new variant of jNMF called Kernel jNMF. This new method incorporates the factorization of the original matrices into a high-dimensional space. Applying our method on synthetic data and biological cancer data, we found that the method performed better in clustering and interpretation compared to the jNMF methods.

INDEX TERMS Data integration, kernel, joint matrix factorization, cancer.

I. INTRODUCTION

INTEGRATION data from different sources has become an area of intense research. In health applications, for example, high-throughput omics technologies provide a wealth of information related to different types of molecular entities (e.g., DNAs, RNAs, proteins) about cells and organisms. The integration of this vast information for multiple individuals, such as those stored in The Cancer Genome Atlas (TCGA) [1] and The Cancer Cell Line Encyclopedia (CCLE) [2] projects, allows the identification of associations between different sources, and to find groups of related molecules across different layers of information. In addition, the use of machine learning methods to integrate these data opens a diverse field of possibilities to improve the discovery of patterns embedded in the original data [3], [4].

Among the strategies for multi-view integration of data, three main groups stand out (early, intermediate, and late integration), which are based on dimension reduction and ensemble methods. In early integration, the matrices are concatenated to make feature selection or decomposed in principal components (PC) to create new variables that can be used as input in some machine learning models. In intermediate integration, the data are initially processed individually, for example, using kernel functions to extract non-linear patterns of the data and use these patterns as input in an ensemble model. Finally, late integration consists of integrating ensemble models, where individual models are generated for each input and then the results are integrated into a final model by voting or averaging strategies [5].

As in conventional machine learning, multi-view data integration can be classified as supervised or unsupervised methods.

Among the unsupervised methods based on the Non-negative Matrix Factorization (NMF) technique, the join NMF (jNMF) is the benchmark for intermediate integration [6]. While in the NMF a matrix $\mathbf{X}$ is linearly factorized into two low-rank matrices (i.e., a base matrix $\mathbf{W}$ and a coefficient matrix $\mathbf{H}$), in the jNMF this process is done simultaneously for different input matrices $\mathbf{X}_1, \ldots, \mathbf{X}_M$. The resulting factorized matrices correspond to a common base matrix $\mathbf{W}$ and as many coefficient matrices as the number of input matrices. The advantage of jNMF over NMF is that it allows finding the common centroids for all samples in the base matrix, whereas the clusters and co-clusters assignments in the coefficient matrix favor the interpretation of embedded patterns [7]. Some extensions of jNMF may help interpretation of clusters such as integrative Orthogonal NMF (iONMF) that uses an orthogonal-regularized penalty

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

to avoid non-overlapping features within clusters providing interpretable models [8].

Despite the versatility and wide range of applications of these methods, both NMF and jNMF are linear models, where the data observed are decomposed in linear combinations of columns of $\mathbf{W}$ and rows of $\mathbf{H}$. However, in many cases, it is expected that the relations between basis vectors are non-linear given the nature of the data [9], [10]. A possible approach to find these relations is to map the observations (x_i) into a higher dimensional space where more meaningful associations may be found. To perform this step, it is possible to use a mapping function ($\phi(.)$) into a high-dimensional space that is not explicitly known but is provided with a scalar product expressed as a kernel function ($K(x_i,\ x_j) = \phi^T(x_i)\phi(x_j)$). In this new space, we can apply the NMF method and achieve a better pattern separation. This approach was used in NMF [11] by comparing two NMF based on kernel functions: the Polynomial NMF (PNMF) and the Gaussian nmNMF (GNMF), where the classification task accuracy performed better in GNMF than PNMF.

In this article, we propose an extension of the jNMF framework for multi-view data integration based on kernels. This implementation, called Kernel joint NMF (KjNMF), was tested on simulated data and observational data from the TCGA project. In both types of data, we incorporated sparsity and prior knowledge constraints to improve cluster identification. In addition, several methods of normalization of the $\mathbf{W}$ and $\mathbf{H}_I$ matrices were tested to improve clustering. We found that, compared with the standard jNMF method and iONMF methods, KjNMF performs better by increasing the cluster quality and, thus, leading to a more accurate interpretation and identification of real clusters.

In this research, we use the following notations:

1) Input non-negative matrices are denoted by bold capital letters, e.g., $\mathbf{X}$. As several matrices contain data on the same samples from different sources (M), the subindex I is added to indicate the particular source $\mathbf{X}_I$ where $I = 1, \ldots, M$. Each matrix has dimensions of n samples per p_i variables.
2) Low-rank matrices are also defined by bold capital letters ($\mathbf{A}$, $\mathbf{W}$ or $\mathbf{H}$), but their dimensions will depend on the rank (k) that is determined beforehand.
3) In the NMF methods, $\mathbf{W}$ is the basis matrix, and $\mathbf{H}$ is the coefficient matrix. In the convex-NMF methods, $\mathbf{W}$ is a linear combination of columns of $\mathbf{X}$ (x_i) and rows of $\mathbf{A}$ (a_{ji}), a matrix of weights.
4) $\mathbf{\Theta}_I$ and $\mathbf{R}_{IJ}$ are sparse matrices that relate the association that exist between variables of the input $\mathbf{X}_I$ matrices. $\mathbf{\Theta}_I$ relates the variables of $\mathbf{X}_I$, whereas $\mathbf{R}_{IJ}$ relates the variables of $\mathbf{X}_I$ and $\mathbf{X}_J$.
5) A kernel is a function that, for all pairs of points x_i and $x_j \in \mathbf{X}$, satisfies $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ where ϕ is a mapping function from $\mathbf{X}$ to an feature space F, i.e., $\phi : x \longmapsto \phi(x) \in F$.
6) The ϕ function maps a column of $\mathbf{X}_I$ to the feature space. So the mapping of the complete matrix is defined by $\Phi(\mathbf{X}_I)$.
7) The hyperparameters are defined as lowercase Greek letters (λ, γ, and ω).
8) ρ corresponds to the cophenetic coefficient and AUC to the Area Under the Curve.

II. RELATED WORK

NMF has proved to be useful on a wide range of applications in pattern identification and classification. However, there are still very complex structures of real-world data that may need methods beyond the standard linear Euclidean formulation to be discovered. The use of non-linear approaches can increase the performance of NMF results [10]. The natural alternative to include non-linearity in the factorization is to map the observations to a higher dimensional space or "feature space", where the clusters or patterns are better defined and, therefore, the method of separation is feasible. The common way is to formulate the NMF optimization problem in terms of the dot product ($\mathbf{X}^T\mathbf{X}$) to incorporate kernel functions, such as Gaussian or polynomial, to do the mapping. A convex-NMF has been proposed in [9], which postulates $\mathbf{W}$ as a weighted of columns in $\mathbf{X}$, i.e., $w_i = \sum_{j=1}^{m} A_{ji}x_i$. Because of this, convex-NMF is established in the form $\|\mathbf{X} - \mathbf{XAH}\|^2$, and the update rules depend on $\mathbf{X}^T\mathbf{X}$.

In recent years, several attempts to kernelize NMF have generated variants that showed high efficiency in clustering and pattern recognition. A method using the semi-NMF strategy was proposed to incorporate the kernel function [12]. This method showed that the discrimination of captured information worked better than the standard NMF method. Other approaches, such as online Kernel NMF (OKNMF), are based on the convex-NMF method with a set of constraints to determine the number of the basis of the feature space [13]. In general, this method performed well and was computationally efficient. A flexible kernel NMF (KNMF) method has also been proposed [11], which presents a general formulation of Gaussian kernel function implementation in NMF.

To allow the integration of data that measures the same set of objects from different approaches or views, there are some processes of data integration that enable NMF. The jNMF method, for instance, has been proposed to standardize the use of NMF into multi-view data [6]. This procedure adapts the two low-rank non-negative matrices obtained in NMF: the basis matrix $\mathbf{W}$ and the coefficient matrix $\mathbf{H}$, into a common basis matrix $\mathbf{W}$ for the multiple views, but maintains the coefficient matrix $\mathbf{H}_I$ for each type of view. The goal is to detect a common pattern for all individuals while maintaining the specific patterns of each view. In this framework, the use of constraints in the form of prior knowledge can increase the performance of clustering by relating features through the within-relationship matrix $\mathbf{\Theta}_I$ and the between-relationship matrix $\mathbf{R}_{IJ}$ [6]. $\mathbf{\Theta}_I$ constraint identifies if there is an existing relationship between feature i and feature j in the same view matrix. $\mathbf{R}_{IJ}$ constraints identifies if there is an existing relationship between feature i in matrix I and feature j in

matrix J. This prior information is included in the framework as constraints in the loss function.

In addition, given that these methods may present a non-unique solution, it has been recommended to normalize the matrices obtained during the convergence process. As stated in [14], the normalization over the matrices could affect the efficiency in the NMF cluster allocation process, because normalizing the matrices alleviates the uncertainties of pattern detection by reducing variations in the data or the need for some initial conditions.

III. METHODS OF JOINT FACTORIZATION OF NON-NEGATIVE MATRICES: JNMF AND KJNMF

A. JOINT NON-NEGATIVE MATRIX FACTORIZATION

The jNMF method integrates data which consist of n individuals from whom p_i variables have been measured from each different source, forming a matrix $\mathbf{X}_I$ for $I = 1, \ldots, M$. This method was adapted by extending the concept of single matrix factorization ($\mathbf{X}$) to a multiple types of matrices ($\mathbf{X}_I$) [6]. Therefore, jNMF factorizes problem (1) the input matrices $\mathbf{X}_I \in \mathbb{R}^{n \times p_i}$ into a low-rank matrices: a common basis matrix $\mathbf{W} \in \mathbb{R}^{n \times k}$ and coefficient matrices $\mathbf{H}_I \in \mathbb{R}^{k \times p_i}$ for each type of data source available, where k is the range of factorization that usually is much less than each p_i. Matrices $\mathbf{H}_I$ and $\mathbf{W}$ correspond to the solution of the minimization problem:

$$\sum_{I=1}^{M} \|\mathbf{X}_I - \mathbf{W}\mathbf{H}_I\|_F^2. \tag{1}$$

with respect to $\mathbf{W}$ and each $\mathbf{H}_I$. Furthermore, prior knowledge can be incorporated in the jNMF problem as constraints by using the matrices $\mathbf{R}_{IJ} \in \mathbb{R}^{pi \times p_j}$ and $\mathbf{\Theta} \in \mathbb{R}^{p_i \times p_i}$ which contains the relations between and within the variables of the different data sources. This not only helps to reduce the instability of the estimation due to the non-uniqueness of the NMF solution but also to the formation of meaningful clusters. In the case of omic data, for instance, $\mathbf{R}_{IJ}$ could express if a specific gene from $\mathbf{X}_I$ is related to a specific miRNA from $\mathbf{X}_J$, i.e., variables from different sources are being related. Similarly, the $\mathbf{\Theta}_I$ matrix contains information about possible relations between variables of the same data source, for example, protein-protein interaction and metabolic pathways relate gene interactions. The superscript (t), i.e., $\mathbf{\Theta}_I^{(t)}$ can be used when several type $\mathbf{\Theta}$ restrictions are used on the same matrix $\mathbf{X}_I$. In both types of restrictions, the matrices are binary coded, taking a value of 1 when there is a relationship and a value of 0 otherwise. These matrices are included in the jNMF optimization problem as constraints as well as regularization terms to favors the correct clustering. The estimation results as in (2).

$$\begin{aligned} \mathbf{W}, \mathbf{H}_{\mathbf{I},\ \mathbf{I}=\mathbf{1},\mathbf{2},\ldots,\mathbf{M}} F(W; H_I) = & \sum_I \|\mathbf{X}_I - \mathbf{W}\mathbf{H}_I\|_F^2 + \gamma_1 \|\mathbf{W}\|_F^2 + \gamma_2 (\sum_I \sum_j \|h_j\|_1^2) \\ & - \lambda_1 \sum_I \sum_t Tr(\mathbf{H}_I \Theta_I^{(t)} \mathbf{H}_I^T) - \lambda_2 \sum_{I \neq J} Tr(\mathbf{H}_I \mathbf{R}_{IJ} \mathbf{H}_I^T) \\ & subject\ to\ \mathbf{W} \geq 0, \mathbf{H_I} \geq 0 \quad . \end{aligned} \tag{2}$$

Another variant of jNMF corresponds to the incorporation of orthogonality constraints which forces the basis vectors to be orthogonal and prevents overlapping of features within the clusters [8]. This generates more interpretable models by better discriminating the features. The orthogonality of the columns of H_I is controlling by the incorporation of the hyperparameter α. The objective function (3) is minimized to obtain W and H_I matrices.

$$\sum_{I=1}^{M} (\|\mathbf{X}_I - \mathbf{W}\mathbf{H}_I\|_F^2 + \alpha \|\mathbf{H}_I^T \mathbf{H}_I - I)_F^2 \ . \tag{3}$$

B. KERNEL JOINT NON-NEGATIVE MATRIX FACTORIZATION (KJNMF)

Given the non-linear structure of real-world data, the kernel-based methods make use of a function ϕ, that maps the original observations (columns) of $\mathbf{X} = [x_1, \ldots, x_p]$ to a higher dimensional space or feature space (F) to obtain the mapping $\Phi(\mathbf{X}) = [\phi(x_1), \ldots, \phi(x_p)]$. The goal of this mapping is to find a way to linearize non-linear relationships in the feature space. The feature space is a vector space that works with finite or infinite dimensions, which must have the properties of being separable and complete. Its inner product exists for any pair of points ($\phi(x_i)$ and $\phi(x_j)$). When the mapping function (ϕ) is applied to $\mathbf{X}$, the coordinates of the observations in this feature space are in general unknown. However, it is possible to calculate their pairwise inner product which is a metric, that defines the distance between pairs of points. In other words, this process creates a distance between observations in F where they seek relationships that can be represented linearly. If the optimization problem is defined using the inner product of the observations in the feature space, then a kernel function is easily applicable without the need to specify the ϕ function.

To modify the objective function of (2) in terms of the inner product of the input matrices $\mathbf{X}_I$, we consider the approach based on the convex-NMF method [9], which proposes that the centroids obtained in $\mathbf{W}$ are a convex combination of the observations (columns), i.e., $\mathbf{W} = \mathbf{X}\mathbf{A}$. This strategy has several advantages: (i) it is possible to impose non-negativity and sparsity constraints on $\mathbf{A}_I$; (ii) it generates a sparse and interpretable solution since it allows to identify how the clusters would be composed; (iii) the matrix $\mathbf{X}$ can have negative values; and (iv) the objective function remains in terms of the kernel function [9], [15]. In our case, we formulated the basis matrix $\mathbf{W}$ as $\Phi(\mathbf{X}_I)\mathbf{A}_I$. It is important to note that in the original jNMF problem there is a single $\mathbf{W}$ that allows the integration of the samples based on the information in each input matrix. This can be understood as follows: given the information provided by the input matrices, the low-rank matrix $\mathbf{W}$ contains the centroids of the samples that gather all this information. For this reason, we introduced a fourth term that allows us to approximate a single $\mathbf{W}^*$, where all products $\Phi(\mathbf{X}_I)\mathbf{A}_I$ are similar between them. This approach was used in the jNMF implementation of [16]. In this implementation the low-rank matrix $\mathbf{W}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

can be different for each factorization, then the researchers normalized $\mathbf{W}$ and calculated a distance measure between each of the $\mathbf{W}$ obtained, i.e., $\sum_{I=1,I\neq J}^{M}\lambda\|\mathbf{W}_I-\mathbf{W}_J\|_F^2$ where λ is a parameter to force the similarity between these matrices. Therefore, we proposed the fourth term in (4) with the parameter ω to control this matrix similarity.

In order to control control on sparsity and scale of low-rank matrices [6], we introduced two additional terms: a fifth term to control the scale of $\mathbf{A}_I$ and a sixth term to induce the matrices $\mathbf{H}_I$ to be sparse. These terms control cluster formation and pattern detection. The parameter γ_1 controls the degree of growth of $\mathbf{A}_I$ and γ_2 controls the required sparsity as we shown in (4).

A graphical structure of the KjNMF method is shown in Fig. 1 and the formulation of the objective function is as in (4).

$$\begin{aligned}\mathbf{A_I},\mathbf{H_I},\mathbf{I=1,2,\ldots,M}F(A_I;H_I)=\sum_I\|\phi(\mathbf{X}_I)-\phi(\mathbf{X}_I)\mathbf{A}_I\mathbf{H}_I\|_F^2\\-\lambda_1\sum_I\sum_t Tr(\mathbf{H}_I\Theta_I^{(t)}\mathbf{H}_I^T)-\lambda_2\sum_{I\neq J}Tr(\mathbf{H}_I\mathbf{R}_{IJ}\mathbf{H}_I^T)\\+\omega\sum_{I\neq J}\|\phi(\mathbf{X}_I)\mathbf{A}_I-\phi(\mathbf{X}_J)\mathbf{A}_J\|_F^2+\gamma_1\sum_I\|\mathbf{A}_I\|_F^2+\gamma_2(\sum_I\sum_j\|h_j\|_1^2)\\ subject\ to\ \ \mathbf{A_I}\geq 0,\mathbf{H_I}\geq 0\quad .\end{aligned}\tag{4}$$

IV. FACTORIZATION ALGORITHM

The common method to solve the optimization problem in jNMF and NMF methods is the multiplicative update rule (MUR). Although diverse variants have been generated to reduce computational resources, MUR keeps being a competitive algorithm with good performance [6], [17]. Our algorithm works in a similar way to the multiplicative rule of jNMF or NMF. Thus, since the optimization problem is not convex, we initially fix the matrix $\mathbf{A}_I$ and update the matrix $\mathbf{H}_I$, and then we fix the latter ($\mathbf{H}_I$) and update the former ($\mathbf{A}_I$). To find the multiplicative rules in our KjNMF context, we derived the objective function defined in (4) and use an approach based on [11] to generate MUR in this kernel problem. As we have mentioned, the base matrix in KjNMF is a linear combination of the mapped data using the ϕ function, so our objective function can be expressed as (5).

$$\begin{aligned}F(A_1,\ldots,A_M;H_1,\ldots,H_M)=\sum_{I=1}^{M}Tr(K_I-K_IA_IH_I-H_I^TA_I^TK_I+H_I^TA_I^TK_IA_IH_I)\\+\gamma_1\sum_{I=1}^{M}Tr(A_I^TA_I)+\gamma_2\sum_{I=1}^{M}Tr(H_I^Te_{jxj}H_I)\\-\lambda_1\sum_{I=1}^{M}\sum_t Tr(H_I\Theta_I^{(t)}H_I^T)-\lambda_2\sum_{I=1,I\neq J}^{M}Tr(H_IR_{IJ}H_J^T)\\+\omega\sum_{I=1}^{M}Tr(A_I^TK_IA_I-A_I^TK_{IJ}A_J-A_J^TK_{JI}A_I+A_J^TK_JA_J).\end{aligned}\tag{5}$$

where $K_I=\Phi(X_I)^T\Phi(X_I)$ is the kernel matrix within data in matrix $\mathbf{X}_I$, and $K_{IJ}=\Phi(X_I)^T\Phi(X_J)$ is the cross-kernel matrices $\mathbf{X}_I$ and $\mathbf{X}_J$. It is important to mention that the kernels are created on the columns and, therefore, a kernel of the type K_I will have dimensions of $p_I\times p_I$.

The Lagrange function L associated with the minimization problem in (4) is $L(\mathbf{A}_I,\mathbf{H}_I)=F+\sum_I Tr(\Psi_I\mathbf{A}_I^T)+\sum_I Tr(\Phi_I\mathbf{H}_I^T)$, where $\Psi_I=[\psi_{ij}^I]$ and $\Phi_I=[\phi_{ij}^I]$ are the parameters from the non-negativity constrains. The partial derivatives of L with respect to $\mathbf{A}_I$ and $\mathbf{H}_I$ are respectively:

$$\begin{aligned}\frac{\partial L}{\partial A_I}=-2K_IH_I^T+2K_IA_IH_IH_I^T+\gamma_1Tr(A_I^TA_I)\\+2\omega(N-1)K_IA_I-2\omega\sum_{J\neq I}K_{IJ}A_J+\Psi_I.\end{aligned}\tag{6}$$

$$\begin{aligned}\frac{\partial L}{\partial H_I}=-2A_I^TK_I+2A_I^TK_IA_IH_I+\gamma_2 2e_{jxj}H_I\\-\lambda_1\sum_t H_I(\Theta_I^{(t)}+(\Theta_I^{(t)})^T)-\lambda_2\sum_{I\neq J}H_JR_{IJ}^T+\Phi_I.\end{aligned}\tag{7}$$

From the Karush-Kuhn-Tucker conditions $\phi_{ij}^I\,(\mathbf{A}_I)_{ij}=0$ and $\psi_{ij}^I\,(\mathbf{H}_I)_{ij}=0$, we calculated the update rule for $(\mathbf{A}_I)_{ij}$ and $(\mathbf{H}_I)_{ij}$:

$$(A_I)_{ij}^{t+1}\leftarrow(A_I)_{ij}^t\frac{[K_IH_I^T+\omega\sum_{J\neq I}K_{IJ}A_J]_{ij}}{[K_IA_IH_IH_I^T+\gamma_1A_I+\omega(N-1)K_IA_I]_{ij}}.\tag{8}$$

$$(H_I)_{ij}^{t+1}\leftarrow(H_I)_{ij}^t\frac{[A_I^TK_I+\frac{\lambda_1}{2}\sum_t H_I(\Theta^{(t)}+(\Theta_I^{(t)})^T)+\frac{\lambda_2}{2}\sum_{I\neq J}H_JR_{IJ}^T]_{ij}}{[A_I^TK_IA_IH_I+\gamma_2e_{j\times j}H_I]_{ij}}.\tag{9}$$

For a complete convergence analysis of the multiplicative algorithm check Supplementary Section S1.

Although NMF methods are non-deterministic polynomial-time (NP-problem), and it is difficult to find a global optimum, it is feasible to identify a local optimum [18]. Therefore, we define a stopping criterion that evaluates the relative difference between two consecutive iterations of the objective function (F) assessed at iteration t, and $t+1$. The algorithm stops when reach a threshold τ, i.e., $F_t-F_{t+1}/F_0-F_{t+1}\leq\tau$; in our case, the stopping threshold was set to 10^{-6}.

The multiplicative rules were implemented in an iterative process as described in Algorithm 1.

V. HYPERPARAMETER SELECTION

For both jNMF and KjNMF, the best set of hyperparameters (λ_1, λ_2, γ_1,γ_2, ω, and k) are selected according to metrics that evaluate the performance of these methods. These metrics were chosen to measure the stability of the clusters when the algorithm is running (Section V-A) and when the clusters are evaluated with a test dataset (Section V-B). In addition, for biological data, it is expected that the clusters will have an interpretation according to the disease or biological process under study (Section V-C). The scale for these metrics is between 0 and 1, with 1 being the best-expected result. For the iONMF method, we evaluated the range (k) and the orthogonality hyperparameter (α). We used the same set of metrics to select the best set of hyperparameters.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

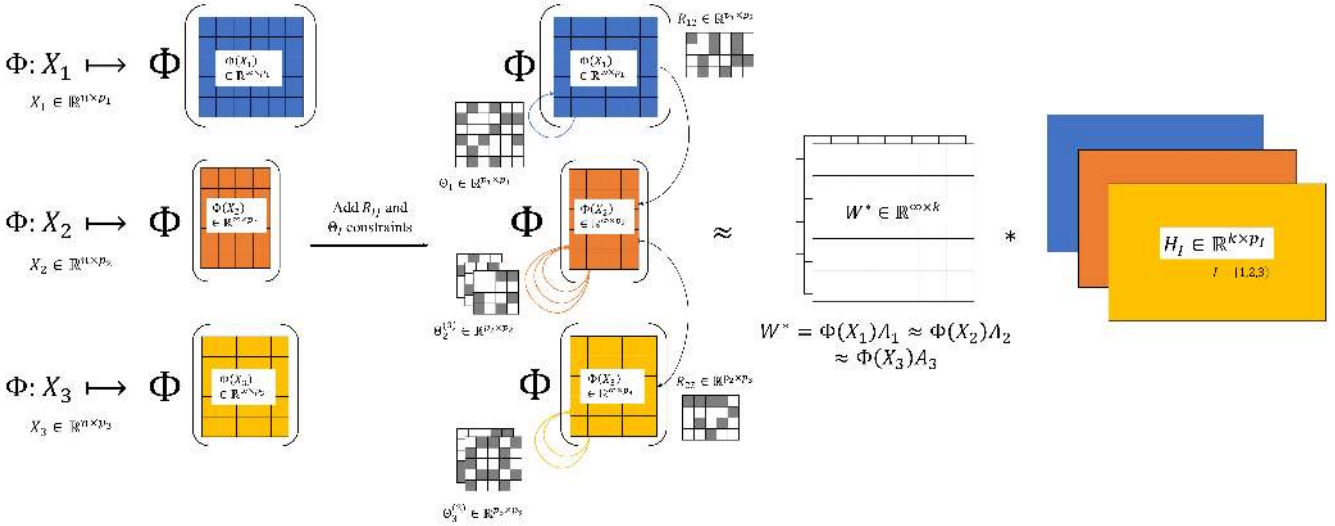


FIGURE 1: Hypothetical case of multi-view integration using KjNMF. Three input matrices are mapped using the function ϕ to a feature space. Additionally, $\mathbf{\Theta}_I$ and $\mathbf{R}_{IJ}$ constraints are added and the joint factorization is performed on the feature space. The resulting low-rank matrices are $\mathbf{A}_I$ and $\mathbf{H}_I$ for each input matrix. The product $\phi(\mathbf{X}_I)\mathbf{A}_I$ is penalized to be similar among the different inputs.

Algorithm 1 Gaussian Join Non-Negative Matrix Factorization (KjNMF)

Require: Kernel matrices $\{\mathbf{K}_{IJ} = e^{-(\frac{\|\mathbf{x}_{Ii}-\mathbf{x}_{Jj}\|^2}{2\sigma})}\}_{i,j=1}^{p_i,p_j}$ for all M input matrices; hyperparameters (λ_1, λ_2, γ_1,γ_2, ω, k); normalization method (NM); $\mathbf{\Theta}_I^{(t)}$ and $\mathbf{R}_{IJ}$ constraints.

Ensure: Low-dimension matrices: $\mathbf{A}_1,\ldots,\mathbf{A}_M$ and $\mathbf{H}_1,\ldots,\mathbf{H}_M$

for $r \leftarrow 1$ **to** R **do**
  Initialize $\mathbf{A}_1,\ldots,\mathbf{A}_M$ *and* $\mathbf{H}_1,\ldots,\mathbf{H}_M$
  Normalize initialized matrices using NM
  Calculate
  $F(\mathbf{A}_1,\ldots,\mathbf{A}_M,\mathbf{H}_1,\ldots,\mathbf{H}_M)_1$
  for $t \leftarrow 1$ **to** t_{max} **do**
    Fix $\mathbf{H}_1,\ldots,\mathbf{H}_M$, *update each matrix* $\mathbf{A}_I$ *as*
    $(A_I)_{ij}^{t+1} \leftarrow (A_I)_{ij}^{t} \frac{[K_I H_I^T + \omega \sum_{J \neq I} K_{IJ} A_J]_{ij}}{[K_I A_I H_I H_I^T + \gamma_1 A_I + \omega(N-1) K_I A_I]_{ij}}$
    Fix $\mathbf{A}_1,\ldots,\mathbf{A}_M$, *update each matrix* $\mathbf{H}_I$ *as*
    $(H_I)_{ij}^{t+1} \leftarrow (H_I)_{ij}^{t} \frac{[A_I^T K_I + \frac{\lambda_1}{2} \sum_t H_I(\Theta^{(t)} + (\Theta_I^{(t)})^T) + \frac{\lambda_2}{2} \sum_{I \neq J} H_J R_{IJ}^T]_{ij}}{[A_I^T K_I A_I H_I + \gamma_2 e_{j \times j} H_I]_{ij}}$
    Normalize $\mathbf{A}_1,\ldots,\mathbf{A}_M$ *and* $\mathbf{H}_1,\ldots,\mathbf{H}_M$ *as* NM
    Calculate
    $F(\mathbf{A}_1,\ldots,\mathbf{A}_M,\mathbf{H}_1,\ldots,\mathbf{H}_M)_{t+1}$
    if $\frac{F_t - F_{t+1}}{F_1 - F_{t+1}} \leq \tau$ **then**
      break
    else
      $t = t + 1$
    end if
  end for
end for

A. CLUSTER STABILITY

The cluster stability measure depends on the connectivity and the consensus matrices. There are as the connectivity matrices as many types of data sources M in the model. For each type of data source, the connectivity matrix $C_I^{(n_I \times n_I)}$ is a squared matrix which indicates if a pair of samples p_1 and p_2 of the same type of data source I belongs to the same cluster, i.e., $C_I[p_1][p_2] = 1$, 0 otherwise [19]. Then, there is the consensus matrix $\hat{C}^{n_i \times n_i}$ that measures the proportion of times that the samples p_1 and p_2 are allocated at the same clusters. The closer $\hat{C}[p_1][p_2]$ is to 0 or to 1, the more stable the allocation process are for this two samples. Associated to the consensus matrix, the Cophenetic coefficient (ρ) can be defined. Accordingly to [20], ρ is calculated as:

$$\rho(\hat{C}_I) = \frac{1}{n_I^2} \sum_i^{n_I} \sum_j^{n_I} 4[\hat{C}_I[t_i][t_j] - \tfrac{1}{2}]^2 \;\; \forall \; M = 1,\ldots I. \tag{10}$$

The larger ρ the better because the structure of the clusters across iterations remain similar [20].

B. AUC ON TESTING DATA

We calculated Area Under Curve (AUC) to investigate the accuracy of identifying embedded patterns by comparing training and testing coefficient matrices. We used a similar approach to the one stated in [6], i.e., to extend the jNMF method to the prediction form. In this procedure, we split the data into train and test datasets. Firstly, we trained the model to find the optimal base and coefficient matrices ($(\Phi(X_I)\mathbf{A}_I)^{Train}$ and $\mathbf{H}_I^{Train}$). Secondly, we set the base matrix ($(\Phi(X_I)\mathbf{A}_I)^{Train}$) obtained from the train and update the test coefficient matrices ($\mathbf{H}_I^{Test}$) until reach the threshold (Section IV). Thirdly, we guarantee perfect matching for the rows of $\mathbf{H}_I^{Train}$ and $\mathbf{H}_I^{test}$ by using the strategy proposed by [6], i.e., we used the Hungarian algorithm which

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

selects what row of $\mathbf{H}_I^{Test}$ corresponds to one row of $\mathbf{H}_I^{Train}$. Finally, the AUC was calculated for each data source. In this way, we expected to find the same compartmentalization of the variables independently of the origin of the samples.

C. BIOLOGICAL RELEVANCE MEASURE

A gene enrichment analysis was performed using the clusterProfiler v.3.14.3 package [21]. This analysis was done for each cluster to determine associations with ontological groups, metabolic, and signaling pathways. In addition, we used OmicsNet [22] and Ingenuity Pathway Analysis [23] to interpret clusters of different molecules (co-clusters).

We used this information to calculate the number of molecules used on average across all omics, the number of genes included in an enriched term on average across all terms (GeneRatio average), and the ratio of enriched terms that were detected in the clusters. As a final metric, we calculated an average between ρ, AUC, GeneRatio average, and the ratio of captured terms as a metric for selecting suitable hyperparameter sets. We call this metric a "performance score", which is acceptable when near to one.

VI. CLUSTER MEMBERSHIP ASSIGNMENT

As mentioned in [9] and [16], the values in the rows of the coefficient matrices ($\mathbf{H}_I \in \mathbb{R}^{k \times v_i}$) could be interpreted as the degree of belonging to the respective cluster. In NMF, the clustering membership assignment for a variable v is as easy as finding the row with the maximum value of the v_{th} column on the matrix $\mathbf{H}_I$. But this assignment rule can assign different variables to a cluster k, and their assignment values can vary, so we decided to implement a level of certainty of the membership assignment to a cluster among the comparable degrees of the associated samples.

With that in mind, the implemented process defines whether the p_{th} variable belongs to a cluster k_{th}. In the first step, we assigned the clusters as usual; finding the row with the maximum value of each column in the matrix $\mathbf{H}_I$ (green-colored cells in Fig. 2). In the second step, we checked if the value in $\mathbf{H}_I$ of those samples already assigned to a cluster k_{th} is at least greater than the first quartile of the value of belonging to the cluster k_{th} of all the variables. Fig. 2 exemplified these assignments. Briefly, in this example, three variables (v_1, v_2 and v_4) were assigned to the cluster k_3 in the first assignment. By calculating the third quartile of this vector [0.57,0.96,0.92], which was 0.94, we deallocated the variable v_1 since it has a lower degree of belonging to the cluster k_3 than those that were assigned to that cluster. In the second assignment, we apply the same strategy for the second maximum degrees of each column, but taking the values greater than the third quartile plus 1.5 times the interquartile range. This was decided because a biomolecule can participate in several biological processes, so it would be very restrictive not to allow this inclusion. It is important to denote that with this new rule for the membership assignment process, some variables could end up deallocated.

First assignment

H_I	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
k_1	0.49	0.05	0.65	0.03	0.51	0.78	0.26	0.26
k_2	0.33	0.73	0.15	0.05	0.64	0.03	0.38	0.57
k_3	0.57	0.96	0.63	0.92	0.10	0.45	0.36	0.25

Second assignment

H_I	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
k_1	0.49	0.05	0.65	0.03	0.51	0.78	0.26	0.26
k_2	0.33	0.73	0.15	0.05	0.64	0.03	0.38	0.57
k_3	0.57	0.96	0.63	0.92	0.10	0.45	0.36	0.25

Final clusters: k_1:{v_6, v_1, v_5}, k_2:{v_5, v_8, v_2}, k_3:{v_2, v_4, v_3}

FIGURE 2: Assignment rule for variables to each cluster. Two assignations are made using the maximum and second maximum per column, deciding whether it belongs to a cluster when within these values exceeds a threshold defined by the quartiles (Q). In the first assignment is used the $Q3$, in the second assignment is used $Q3 + 1.5 * IQR$. The final clusters contain the variables obtained from the two assignments.

VII. NORMALIZATION OF $\mathbf{A}_I$ AND $\mathbf{H}_I$

The problem of non-uniqueness of the resulting matrices in NMF methods leads to misinterpretations of the clusters or patterns. This may be due to noise in the data or the initial conditions of the algorithm. Several strategies in the normalization of $\mathbf{W}$ or $\mathbf{H}_I$ during the computation of the multiplicative rules have been proposed. By exploring how the normalization of the low-rank matrices affects the NMF performance, it has been found that the normalization choice could be used to regulate the non-uniqueness of the NMF increasing the performance on clustering [14]. The process of normalization consists of using mapping functions ($f(.)$) on the columns or rows of the low-rank matrices to obtain diagonal matrices that multiply each value in the low-rank matrix. By using this approach it was found that the maximum norm ($\|.\|_\infty$) had the best performance than traditional normalization methods [14].

We implement different normalization mapping functions to analyze the variation on the clustering allocation process due to the normalization method. In [14], the normalization process divides each column of $\mathbf{W}$ by the value of a mapping function ($f(.)$) that is applied for every column of $\mathbf{W}$; in the case of KjNMF, this process is applied to $\mathbf{A}_I$.

Accordingly to [14] every normalization method should be applied as $\mathbf{W}' = \mathbf{W}\mathbf{D}^{-1}$ and $\mathbf{H}' = \mathbf{D}\mathbf{H}$, where $\mathbf{D}$ is a diagonal matrix of dimension k (the range). The k_{th} value of the diagonal of $\mathbf{D}$ is the result of mapping function (e.g., $\|.\|_\infty$) applied to the k_{th} column of $\mathbf{W}$ (w_k). For example, the $\mathbf{D}$ matrix for $\mathbf{W}$ is:

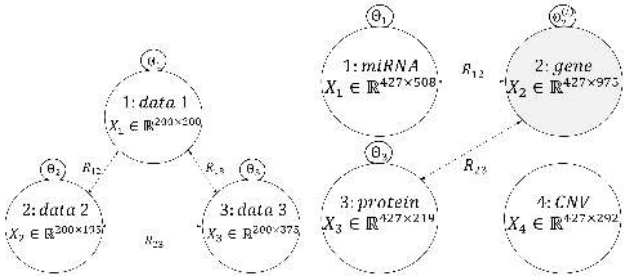
$$D = \begin{vmatrix} f(w_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & f(w_k) \end{vmatrix} \quad (11)$$

There are different ways to find this $\mathbf{D}$ matrix. In [14], for instance, it was used the columns of the $\mathbf{W}$ matrix to be mapped into various norms to find the values of $\mathbf{D}$. The product $\mathbf{WD}^{-1}$ means that each value of the diagonal of $\mathbf{D}$ divides each value of the k_{th} column of $\mathbf{W}$, whereas in the product $\mathbf{DH}$ is multiplied by the k_{th} row of $\mathbf{H}$. We investigated other alternative cases of normalization based on the original jNMF by [6] and the research of [14]:

1) Adapted Zhang method [6]: they concatenated all the $\mathbf{H}_I$ matrices along the common rows (k) and applied the mapping function $f(\mathbf{H}^{concat}_{I_{k_i}}) = 1/sum(\mathbf{H}^{concat}_{k_i})$ to calculate the k_{th} element of the diagonal of $\mathbf{D}$. We adapt this normalization method to the KjNMF as there are multiple $\mathbf{A}_I$. Then we implemented the normalization as $\mathbf{A}'_I = \mathbf{A}_I\mathbf{D}^{-1}$ and $\mathbf{H}'_I = \mathbf{DH}_I$.
2) Modified Zhang method [6]: we modified the previous normalization method by calculating an individual $\mathbf{D}_I$ using the same mapping function over each row of $\mathbf{H}_I$. Then we implemented the normalization as $\mathbf{A}'_I = \mathbf{A}_I\mathbf{D}_I^{-1}$ and $\mathbf{H}'_I = \mathbf{D}_I\mathbf{H}_I$.
3) Maximum value per column of $\mathbf{A}_I$: [14] used the mapping function $f(\mathbf{W}_k) = max(\mathbf{W}_k)$ to calculate the k_{th} element of the diagonal of $\mathbf{D}$. For KjNMF, we applied the mapping function over each $\mathbf{A}_I$. Then we implemented the normalization as $\mathbf{A}'_I = \mathbf{A}_I\mathbf{D}_I^{-1}$ and $\mathbf{H}'_I = \mathbf{D}_I\mathbf{H}_I$.
4) l_2 norm per column of $\mathbf{A}_I$: we implemented a normalization based on the l_2 norm. The mapping function applied is $f(\mathbf{A}_{I_k}) = \|\mathbf{A}_{I_k}\|_2$. The implemented normalization was $\mathbf{A}'_I = \mathbf{A}_I\mathbf{D}_I^{-1}$ and $\mathbf{H}'_I = \mathbf{D}_I\mathbf{H}_I$.
5) l_1 norm per column of $\mathbf{A}_I$: [14] emphasizes in the use of the l_1 norm because it has a probabilistic interpretation. We implemented $f(\mathbf{A}_{I_k}) = \|\mathbf{A}_{I_k}\|_2$ as the mapping function. The implemented normalization was $\mathbf{A}'_I = \mathbf{A}_I\mathbf{D}_I^{-1}$ and $\mathbf{H}'_I = \mathbf{D}_I\mathbf{H}_I$.

VIII. EXPERIMENTS

We present two evaluation scenarios for the proposed algorithm (KjNMF) and linear methods (jNMF and iONMF). The first case uses synthetic data defined by isotropic Gaussian distributions, meaning that, it can be considered as spheres immersed in others. The second case belongs to the bioinformatics field, where we tested the implementation on cancer omic data since these type of data contain a non-linear structure and several patients with different characteristics. In addition, using biological databases to create prior knowledge constraints will allow a higher potential to find biomarkers or biological pathways that are associated with a particular disease. Fig. 3 shows a graphical representation of these relations in both synthetic data and TCGA datasets. In each case, the algorithm was iterated in 3-fold cross-validation, in which the ρ metric was calculated and the AUC in testing data were averaged for each repetition. Numeric results for synthetic and biological datasets are in Supplementary File 1 and File 2, respectively. The convergence time for MUR was approximately less than a one-half hour using a 2.20GHz Intel Corei7 processor with 16GB RAM.



(a) (b)

FIGURE 3: Graph representation of the input matrices for the two cases evaluated. (a) Synthetic data created from an isotropic Gaussian distribution. Constraints $\mathbf{R}_{IJ}$ and $\mathbf{\Theta}_I$ generated as sparse arrays. (b) Omic profiles for proteins, genes, miRNA, and copy number variation (CNV) from the TCGA low-grade glioma dataset (TCGA-LGG). Constraints created according to the information of several databases for physical interactions, associations, and molecular relations.

A. SYNTHETIC DATA

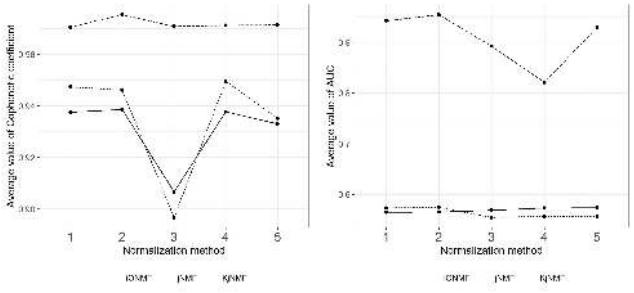
We created three data sources with an equal number of samples n and a different number of variables p (200, 195, and 375). We employed the package Scikit-learn v0.23.2 to generate a mixture of isotropic Gaussian datasets, which create a classification dataset of nested concentric multi-dimensional spheres. For each X_I matrix, two isotropic Gaussian datasets were generated, one with mean 0 and variance 2 and the other with mean 4 and variance 1. In both cases, 2 classes were defined, i.e., each observation was assigned to a cluster (Supplementary Fig. F1). The constraints $\mathbf{R}_{IJ}$ and $\mathbf{\Theta}_I$ were created using a random sparse matrix of 0 and 1.

The hyperparameters were evaluated in a random grid where λ_1, λ_2, γ_1, γ_2 and ω were defined in a range of $[1^{-8}, 1]$ for the normalization methods 1, 2 and 3, and in a range of $[1^{-10}, 1^{-7}]$ for the normalization methods 4 and 5 (SectionVII). The hyperparameter values for the normalization methods 4 and 5 were chosen because they use l_2 and l_1 norm, respectively. Therefore, penalizing with large values generates low-rank matrices with values at or near zero, which leads to numerical problems. A total of 300 scenarios were run for all the normalization methods. The range (k) was evaluated between 5 and 100. For KjNMF, σ, the hyperparameter of the kernel, was evaluated between 1 and 5.

Fig. 4 shows the behavior of the AUC and ρ in the different types of normalization methods for iONMF, jNMF, and KjNMF. We found a similar behavior between all types of normalization for these methods. In both metrics, KjNMF performed better than linear methods, where the AUC was

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

above 0.9 for KjNMF in most cases, whereas it did not exceed 0.6 for jNMF and iONMF (Supplementary Fig. F2).



(a) (b)

FIGURE 4: Evaluation of (a) ρ and (b) AUC in each type of normalization method for synthetic data.

In all three algorithms the results of ρ were very close to each other (ranging between 0.8 and 1.0). In addition, in the fifth normalization method, we observed that the behavior of ρ is very stable in both algorithms, although KjNMF always presents better performance than jNMF and iONMF methods. These results show that in all cases the clusters are stable (ρ with high values), i.e., clusters are similar in each run. However, in the case of KjNMF, the clusters are similar to each other ($AUC \approx 1$) independent of the sample (train and test), whereas jNMF or iONMF ($AUC \approx 0.6$) the clusters formed will be different for each sample.

B. BIOLOGICAL DATA

We used datasets from TCGA that contains measurements of gene expression at the level of both transcription (mRNA) and translation (protein), RNA-based gene regulation (miRNAs), and genetic structural variation (CNV) from cancer patients. Since the number of genes was very high ($\sim 20K$), we selected the genes that have been previously associated with specific cancer by using the MalaCards database (Supplementary Section S2). In addition, we obtained prior knowledge of biological molecule associations or interactions to create $\mathbf{R}_{IJ}$ and $\mathbf{\Theta}_I$ constraints by using publicly available biological databases (composition of prior knowledge constraints in Supplementary Section S3). The databases and packages used were:

- BioGRID v3.5 [24]: gene-gene associations.
- STRINGdb v9.1 [25]: protein-protein associations.
- KEGG graphite v1.32.0 [26]: gene-gene associations in metabolic pathways.
- limma v3.42.2 [27]: co-expressed genes.
- miRNet v2.0 [**?**]: miRNA-gene associations.
- CancerNet [28]: miRNA-miRNA associations.

We compared the performance of the jNMF, iONMF, and KjNMF algorithms across three TCGA cancer types: BReast CAncer (BRCA), Low-Grade Glioma (LGG), and LUng ADenocarcinoma (LUAD). The data were obtained using the TCGA-Assembler tool [29]. These cancer types were selected not only because of the dimensionality of the data and heterogeneity of the disease but also because of its epidemiology: where breast and lung cancer are the predominant neoplasms in adults, whereas glioma the most common solid tumor in children. In addition, we evaluated LGG as a specific study case for clustering interpretation.

1) Performance of KjNMF on different datasets

We evaluated the performance of the KjNMF using biological datasets. The $\mathbf{R}_{IJ}$ and $\mathbf{\Theta}_I$ constraints were generated for each of the three TCGA cancer types. The hyperparameters γ_1, γ_2, and ω were evaluated respectively as $[1^{-6}, 1^{-4}, 1^{-3}]$ for the normalization methods 1, 2, and 3; and as $[1^{-10}, 1^{-9}, 1^{-7}]$ for the normalization methods 4 and 5. For λ_1 and λ_2, the values were fitted among $[1^{-10}, 1^{-2}]$. For the iONMF method, the α hyperparameter were evaluated in the range $[1^{-10}, 1]$.

Considering that different sets of hyperparameters can be evaluated for all the algorithms, we found that the kernel implementation is superior to the linear methods. Therefore, we evaluated normalization methods to identify the range (k) with good performance of stability ($\rho \approx 1$ and $AUC \approx 1$) and biological interpretation (refer to Table 1). The normalization method 4 for KjNMF presented good scores of these metrics for the three types of cancer evaluated. The ρ evidenced cluster stability as this value is an average of the cluster structure of each MUR repeat (greater than 0.90). Concerning the AUC, when an AUC is close to 1 it means that the train $\mathbf{H}_I$ matrix is very similar to the test $\mathbf{H}_I$, then the assignation of variables within the different clusters is similar. This can be understood because they come from a common type of cancer and certain molecules will maintain their relationship independent of the origin of the sample. Finally, the enrichment ratio (the quantity of clusters enrichment over the range) was above 0.60 in the majority of cases (refer to Table 1 and Supplementary Fig. F3 and F4). The other four normalization methods had a good performance but they were not as consistent as method 4.

2) Effect of prior knowledge and similarity constraints in KjNMF

We evaluated whether prior knowledge constraints have any effect on cluster formation. We found that metrics such as the gene ratio average in KjNMF was 0.45, whereas jNMF was below 0.25 ($p - value < 0.001$, unpaired two-samples t-student test). This suggests that the jNMF method is more unstable due to the lack of terms that allows the incorporation of prior knowledge (Supplementary Fig. F5). On the other hand, we tested for the omission of the hyperparameter ω which controls the similarity between the products $\Phi(\mathbf{X}_I)\mathbf{A}_I$. We found a negative effect on the AUC because it dropped to less than 0.6 ($p - value < 0.001$, unpaired two-samples t-student test). These results indicate that the KjNMF method is still able to cluster the features using the non-linear data but loses the ability to use predictive cluster identification in a test dataset. For this reason, it is recommended to include these constraints.

TABLE 1: Metric values for different types of cancer data using the five normalization methods for the factorization algorithm KjNMF and jNMF. The Biological Terms detected is the ratio between the number of enrichment terms detected over the total enrichment terms.

TCGA	K	Method	Normalization 1				Normalization 2				Normalization 3				Normalization 4				Normalization 5			
			ρ	AUC	Gene ratio	Bio.Terms detected	ρ	AUC	Gene ratio	Bio.Terms detected	ρ	AUC	Gene ratio	Bio.Terms detected	ρ	AUC	Gene ratio	Bio.Terms detected	ρ	AUC	Gene ratio	Bio.Terms detected
LUAD $\mathbf{X}_I \in \mathbb{R}^{310\times p_i}$ p_i : 1102 miRNA, 2340 gene, 1062 CNV and 240 proteins	30	KjNMF	0.956	0.583	0.185	0.182	0.948	0.576	0.204	0.455	0.938	0.551	0.218	0.273	0.952	0.864	0.439	0.545	0.946	0.549	0.216	0.182
		jNMF	0.817	0.59	0.088	0.818	0.838	0.605	0.098	0.636	0.783	0.664	0.177	1	0.846	0.678	0.11	0.636	0.825	0.662	0.118	0.364
		iONMF	0.897	0.351	0.115	0.455	0.9	0.355	0.107	0.182	0.838	0.336	0.38	1	0.885	0.34	0.137	0.909	0.857	0.318	0.157	0.909
	60	KjNMF	0.975	0.581	0.302	0.818	0.968	0.558	0.456	0.455	0.907	0.578	0.756	0.727	0.981	0.862	0.286	1	0.965	0.578	0.306	0.727
		jNMF	0.945	0.64	0.159	0.364	0.938	0.642	0.162	0.273	0.902	0.637	0.228	0.727	0.9	0.686	0.154	0.455	0.922	0.611	0.255	0.273
		iONMF	0.926	0.355	0.153	0.909	0.925	0.356	0.156	0.727	0.903	0.301	0.427	0.818	0.925	0.326	0.142	0.909	0.849	0.33	0.264	0.818
	90	KjNMF	0.981	0.589	0.481	1	0.974	0.565	0.4	0.818	0.81	0.564	0.456	1	0.987	0.848	0.421	0.818	0.971	0.556	0.509	1
		jNMF	0.962	0.646	0.254	0.636	0.955	0.645	0.246	0.364	0.94	0.643	0.585	0.636	0.913	0.691	0.275	0.636	0.923	0.615	0.442	0.909
		iONMF	0.941	0.356	0.244	0.545	0.941	0.358	0.202	0.636	0.933	0.325	0.543	0.818	0.94	0.352	0.338	0.545	0.856	0.317	0.576	0.818
LGG $\mathbf{X}_I \in \mathbb{R}^{427\times p_i}$ p_i : 508 miRNA, 975 gene, 292 CNV and 219 proteins	30	KjNMF	0.954	0.576	0.254	0.366	0.947	0.614	0.251	0.464	0.889	0.6	0.309	0.627	**0.96**	**0.928**	**0.227**	**0.895**	0.939	0.628	0.236	0.647
		jNMF	0.835	0.599	0.108	0.908	0.85	0.601	0.111	0.908	0.807	0.699	0.112	0.928	0.801	0.7	0.133	0.922	0.786	0.698	0.125	0.941
		iONMF	0.863	0.376	0.12	0.889	0.907	0.382	0.118	0.85	0.835	0.371	0.129	0.902	0.888	0.36	0.196	0.935	0.869	0.373	0.113	0.922
	60	KjNMF	0.973	0.619	0.513	0.725	0.964	0.582	0.429	0.739	0.937	0.617	0.594	0.745	0.977	0.922	0.335	0.889	0.957	0.581	0.567	0.68
		jNMF	0.941	0.633	0.223	0.68	0.936	0.634	0.204	0.444	0.888	0.631	0.423	0.824	0.877	0.737	0.16	0.902	0.925	0.603	0.237	0.68
		iONMF	0.91	0.374	0.175	0.935	0.927	0.379	0.162	0.895	0.898	0.34	0.31	0.895	0.912	0.365	0.284	0.954	0.865	0.351	0.245	0.954
	90	KjNMF	0.979	0.585	0.676	0.804	0.973	0.626	0.748	0.81	0.964	0.624	0.749	0.85	0.982	0.917	0.539	0.908	0.964	0.622	0.627	0.784
		jNMF	0.959	0.639	0.265	0.647	0.948	0.64	0.4	0.817	0.89	0.637	0.509	0.889	0.876	0.739	0.31	0.948	0.931	0.603	0.471	0.765
		iONMF	0.893	0.378	0.208	0.935	0.927	0.376	0.192	0.895	0.9	0.36	0.24	0.974	0.915	0.375	0.181	0.961	0.873	0.359	0.229	0.961
BRCA $\mathbf{X}_I \in \mathbb{R}^{623\times p_i}$ p_i : 638 miRNA, 533 gene, 155 CNV and 226 proteins	30	KjNMF	0.952	0.646	0.303	0.389	0.946	0.649	0.315	0.442	0.898	0.645	0	0	0.968	0.934	0.366	0.947	0.94	0.595	0.506	0.85
		jNMF	0.818	0.605	0.174	0.965	0.864	0.627	0.183	0.982	0.768	0.684	0	0	0.788	0.706	0.168	0.956	0.825	0.662	0.118	0.364
		iONMF	0.872	0.361	0.165	0.947	0.891	0.361	0.174	0.947	0.844	0.346	0.206	0.982	0.852	0.354	0.169	0.965	0.812	0.342	0.173	0.973
	60	KjNMF	0.972	0.607	0.693	0.894	0.963	0.609	0.55	0.761	0.916	0.652	0.685	0.664	0.981	0.905	0.492	0.947	0.959	0.643	0.726	0.867
		jNMF	0.942	0.66	0.376	0.832	0.927	0.661	0.365	0.779	0.902	0.66	0.491	0.947	0.859	0.688	0.246	0.991	0.922	0.611	0.255	0.273
		iONMF	0.941	0.356	0.244	0.545	0.941	0.358	0.202	0.636	0.933	0.325	0.543	0.818	0.94	0.352	0.338	0.545	0.856	0.317	0.576	0.818
	90	KjNMF	0.976	0.618	0.769	0.965	0.973	0.66	0	0	0.912	0.653	0.797	0.823	0.987	0.897	0.693	0.956	0.962	0.606	0.789	0.956
		jNMF	0.959	0.662	0.432	0.894	0.942	0.662	0	0	0.896	0.661	0.677	0.991	0.882	0.69	0.319	0.973	0.923	0.615	0.442	0.909
		iONMF	0.923	0.353	0.311	0.982	0.932	0.356	0.269	1	0.913	0.336	0.44	0.965	0.914	0.348	0.356	0.982	0.854	0.335	0.336	0.973

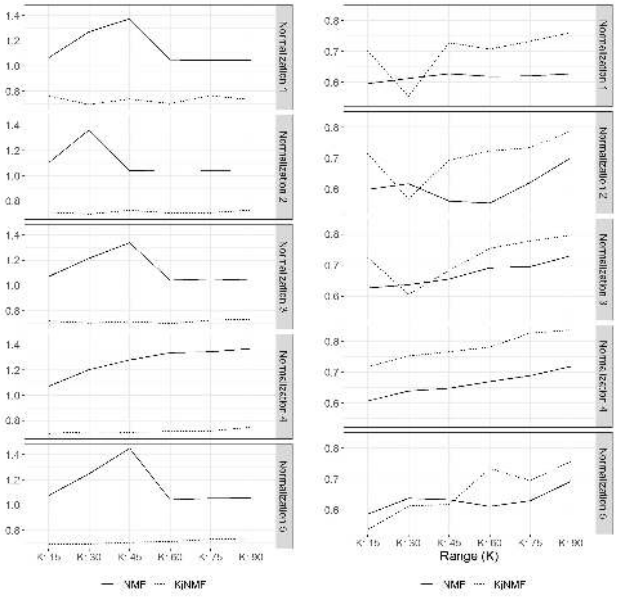
3) Case of study: Low-Grade Glioma

We evaluated the case of LGG, a brain type of cancer, to assess the knowledge identified by the clusters using the KjNMF method. We found differences in the number of enriched clusters generated by both algorithms. We also noticed that although the proportion of variables used in KjNMF was less than in jNMF; with a value close to 0.75 (Fig. 5a and Supplementary Fig. F6), this did not reduce the number of enrichment terms found for KjNMF. The Performance Score (enriched and stable clusters) obtained with KjNMF was higher or equal in most cases compared to those obtained with jNMF and iONMF (Fig. 5b and metrics plots of LGG in Supplementary Fig. F4). This lead to a better distribution of the groups with a good density of molecules per cluster, facilitating its biological interpretation. As a consequence, we found that there are sets of hyperparameters that can satisfy all evaluation metrics, i.e., finding reliable results for interpretation.

Although we worked with a small (15) and a large (90) number of clusters (k), we explored the enrichment of $k = 30$ since the number of variables included in each module could be poorly or highly concentrated in $k = 90$ or $k = 15$, respectively. The best results for $k = 30$ were found in the normalization method 4 with (refer to Table 1 results in bold) with the hyperparameters as follows: $\gamma_1 = 1 \times 10^{-10}$, $\gamma_2 = 1\times10^{-8}$, $\omega = 1\times10^{-10}$, $\lambda_1 = 2\times10^{-9}$ and $\lambda_2 = 2\times10^{-9}$. The value of σ, a hyperparameter of kernel function, was 4.

We used clusterProfiler v.3.14.3 package [21] to perform gene enrichment analysis for each one of the 30 clusters. We found 196 enriched terms (Supplementary File 4), of which 36 were not identified by jNMF (by using the best set of hyperparameters used in Table 1). Among the new terms, we highlight the terms associated with oxidative phosphorylation, aminoacids metabolism, and glycosphingolipid biosynthesis because this allows us to identify that the clusters contain information on basic and more complex processes related to the disease.

A feature of the jNMF methods is that they allow association between clusters, i.e., the k_{th} cluster of miRNA, proteins, and genes can be biologically associated, this is known as a co-cluster. We investigated whether there was any relationship between gene/miRNA and gene/protein co-clusters. In the case of gene/miRNA co-cluster, we explored the top 5 co-clusters with the most molecules. We used OmicsNet tool [22] and miRNet v.2.0 [**?**] to interpret co-clusters between miRNA and genes. The enriched terms involved in the co-clusters were related to cancer processes as presented in Table 2 (Supplementary File 5). For example, gene cluster No. 7 and miRNA cluster No. 7 have the terms Fc epsilon RI signaling pathway and T-Cell activation. Interestingly, these two pathways are related in the context of type I hypersensitivity reactions. Briefly, allergens stimulate T helper type 2 (Th2) cells to secrete interleukin (IL)-4



(a) (b)

FIGURE 5: Evaluation metrics for KjNMF, iONMF and jNMF using LGG data. (a) the ratio between variables assigned in clusters and the total variables and (b) Performance score which refers to the ratio of enrichment terms captured by the clusters over the number of total terms (using all the genes), the higher the performance.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

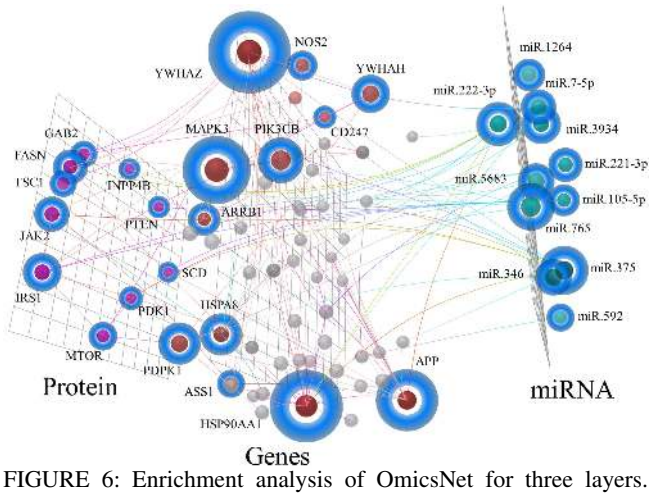


FIGURE 6: Enrichment analysis of OmicsNet for three layers. Highlighted nodes belong to the co-cluster 21 that we selected to analyze. Dark blue-filled nodes correspond to the related biomolecules in the enriched terms listed above. The size of the nodes is for visual perspective.

causing B cells to release immunoglobulin (Ig)-E which bind to the high-affinity FcϵRI of the mast cells which, as a consequence, release biologically active mediators, causing mainly an allergic reaction [30], [31]. In cancer, the studies of atopy as a risk factor for cancer have reached contradictory results [32], [33]. Therefore, our multi-omic strategy can be used to explore whether allergic diseases might be associated with cancer risk.

To interpret a specific co-cluster case for gene/protein and gene/miRNA, we explored co-cluster No. 21, which contains 157 genes, 54 miRNAs, and 47 proteins. We used OmicsNet tool [22] to obtain an interaction network between omics. Then, we used a minimum-network option in OmicsNet tool [22] because it generates a network with the input molecules which allows us to identify associations between them. This network showed a relationship among the molecules of the three input omics (Fig. 6).

We further inspect gene/protein associations using Ingenuity Pathway Analysis (IPA) software [23]. The enrichment analysis for genes/proteins comprises the PI3K/AKT signaling, Insulin Receptor Signaling, IGF-1 signaling, and other biological pathways with a $p - value$ lower than 1×10^{-8} (Supplementary File 6). This cluster grouped genes and proteins that were associated with tumor cell growth and survival. Our results, therefore, suggest that our KjNMF method improves the association in terms of enrichment. We also highlighted that, even by using a restricted set of genes, we were able to identify important clusters related to low-grade glioma, i.e., KjNMF does not create clusters by randomly assigning genes but rather incorporating all the information from the omics and previous knowledge.

IX. DISCUSSION

In different areas such as health, bioinformatics, or robotics, the integration of data from different sources becomes indispensable for decision making. In particular, data integration becomes increasingly important in biological data science to reveal new and meaningful insights into the complex pathological processes and disease pathways. Since biological data represent complex systems, there is a need to developing strategies that facilitate their understanding. Therefore, we proposed this methodology of omic data integration through joint factorization based on kernels.

Here we presented the KjNMF which, is a method that takes the standard jNMF method based on kernels to perform the factorization in a high-dimensional feature space. Different kernel NMF approaches have been developed and their advantage over the conventional method has been demonstrated [10]. However, we went one step further to integrate prior knowledge and explore the advantages of doing this integration in a feature space where the non-linearity in the patterns of the real-world data can be understood by linear relationships.

By using the Gaussian isotropic distribution to create synthetic data, we found that our method performs better than the standard jNMF methods. It has been shown that by incorporating non-linearity into the linear methods, they can perform much better with high-dimensional data [34]. To establish the best performance of KjNMF vs. jNMF methods, we defined a set of metrics capable of identifying the best set of hyperparameters. These hyperparameter selection metrics required the following conditions to be met: high cluster structure stability ($\rho \approx 1$) and high cluster stability on a train and test dataset (high values of AUC). KjNMF satisfied these conditions which are interesting because in the standard jNMF methods the AUC remained low (≈ 0.60), i.e., the patterns of the data hardly hold and the interpretation may be different for these patterns. In both cases, ρ remained above 0.8, which allows us to establish that once the stop threshold is reached the structure of the clusters remains similar in each iteration of the algorithm. Therefore, it is clear that the synthetic data are not suitable for linear methods since they contain non-linear structures.

In the biological domain, it was found that molecules were not randomly assigned to a cluster, that is, both methods jNMF and KjNMF found meaningful clusters. We identify two advantages of KjNMF over jNMF and iONMF methods: the enrichment analysis of KjNMF showed new biological terms that jNMF could not detect, there was also an association between co-clusters of different molecules, i.e., the co-cluster associations are created between common processes such as immune response processes. This is interesting because other authors have found no direct relationship between co-clusters [7]. We can attribute this also to the inclusion of prior knowledge constraints which narrowed the search space of feasible solutions, i.e., they force the formation of these clusters. We incorporated these specific constraints for each layer, just as the standard jNMF method [6]. These constraints allow associations between molecules of the same category (e.g., gene-gene) and between different categories (e.g., miRNA-protein).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3096801, IEEE Access

TABLE 2: Enrichment of miRNA and genes clusters.

Cluster	Molecule	Enriched term	Hits	p-value adjusted[1]
c-7	Gene	FcϵRI signaling pathway	14	9.41×10^{-9}
	miRNA	T-Cell Activation	10	5.93×10^{-7}
c-10	Gene	MicroRNAs in cancer	12	3.65×10^{-3}
	miRNA	Actin Filament Network Formation	6	3.18×10^{-6}
c-14	Gene	Renal cell carcinoma	15	9.65×10^{-9}
	miRNA	Kidney Neoplasms	15	4.35×10^{-16}
c-21	Gene	SNARE interactions in vesicular transport	8	1.67×10^{-4}
	miRNA	Inflammation	20	8.624×10^{-5}
c-30	Gene	cGMP-PKG signaling pathway	79	4.02×10^{-27}
	miRNA	Cholesterol Homeostasis	13	2.24×10^{-3}

[1] Adjusted p-values were calculated using hypergeometric test.

Along with this, we also explored the normalization methods during the updating of $\mathbf{A}_I$ and $\mathbf{H}_I$ matrices, since these methods reduce the non-uniqueness problem of NMF methods [14]. As it was evaluated in [14], the normalization allows the stability of the obtained clusters avoiding that initial conditions or noise alter its formation. Methods 1, 2, and 3 are modifications of the original jNMF implementation. We found that the performance of the jNMF method presented was lower in the AUC metrics and the information captured (number of enrichment clusters) than the KjNMF method. Methods 4 and 5 use l_2 and l_1 norms, respectively. Specifically, method 4 was the one that showed the best performance in all the evaluated scores. The use of the norms (l_1 and l_2) allows the resulting matrices to consist only of the relevant components. Therefore, the noise of the data may influence the correct formation of the clusters, we recommend the use of the normalization method 4 since the l_2 norm can reduce many values near zero without loss of information [35]. The other normalization methods were associated with a low AUC, so it is likely that the clusters are affected by the origin or noise of the data ($\mathbf{X}_I^{test}$), as a consequence this may affect the interpretation of the clusters.

The main limitation of the proposed method is related to the loss of interpretation of the samples or patients. This is because in the jNMF the base matrix ($\mathbf{W}$) has a centroid interpretation where each sample will belong, e.g., clusters of patients with distinguishable biological characteristics are generated. The probability of survival, for instance, is statistically different between the clusters obtained by using the jNMF methods [36]. In our approach, this is not easily achieved since this matrix is located in a feature space of infinite dimensions. Despite this, pre-imaging research based on different strategies could be explored in the future [37], even though a disadvantage of using pre-imaging is that the solution is not unique. However, we believe that the proposed approach represents an advance in data integration using kernel-based joint matrix factorization. Thus, our method applies to other areas and diseases where databases are currently available.

X. CONCLUSION

A method of integration using joint non-negative matrix factorization based on kernels (KjNMF) was presented. This method used the Gaussian kernel, demonstrating its efficiency against the standard jNMF method. The advantages over the standard jNMF method are: (i) it allows working with data with non-linear structures, (ii) it allows identifying associations between variables (interpretable clusters), and (iii) it shows better stability of the clusters in training and testing datasets. Besides, novel evaluation mechanisms were proposed in our method, such as (iv) the incorporation of prior knowledge in the kernel proposal, (v) the incorporation of normalization methods to avoid the non-uniqueness problem, (vi), and the use of metrics to determine a set of hyperparameters useful in cluster interpretation.

We use synthetic data and biological data to demonstrate these advantages. In all methods, the KjNMF outperformed standard jNMF methods concerning cluster stability and interpretability metrics, i.e., the proposed method allows to reduce the non-uniqueness problem, facilitating the interpretation by finding new clusters. In addition, we used different methods to normalize the low-rank matrices. We find that a good approach to reduce the noise that may be included in the real-world data is to use the l_2 norm. Therefore, our approach allows understanding complex data that can be integrated by increasing the value of the interpretation.

REFERENCES

[1] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. M. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, and J. M. Stuart, "The Cancer Genome Atlas Pan-Cancer analysis project," *Nat. Genet.*, vol. 45, no. 10, pp. 1113–1120, oct 2013. [Online]. Available: http://www.nature.com/articles/ng.2764

[2] M. Ghandi, F. W. Huang, J. Jané-Valbuena, G. V. Kryukov, C. C. Lo, E. R. McDonald, J. Barretina, E. T. Gelfand, C. M. Bielski, H. Li, K. Hu, A. Y. Andreev-Drakhlin, J. Kim, J. M. Hess, B. J. Haas, F. Aguet,

B. A. Weir, M. V. Rothberg, B. R. Paolella, M. S. Lawrence, R. Akbani, Y. Lu, H. L. Tiv, P. C. Gokhale, A. de Weck, A. A. Mansour, C. Oh, J. Shih, K. Hadi, Y. Rosen, J. Bistline, K. Venkatesan, A. Reddy, D. Sonkin, M. Liu, J. Lehar, J. M. Korn, D. A. Porter, M. D. Jones, J. Golji, G. Caponigro, J. E. Taylor, C. M. Dunning, A. L. Creech, A. C. Warren, J. M. McFarland, M. Zamanighomi, A. Kauffmann, N. Stransky, M. Imielinski, Y. E. Maruvka, A. D. Cherniack, A. Tsherniak, F. Vazquez, J. D. Jaffe, A. A. Lane, D. M. Weinstock, C. M. Johannessen, M. P. Morrissey, F. Stegmeier, R. Schlegel, W. C. Hahn, G. Getz, G. B. Mills, J. S. Boehm, T. R. Golub, L. A. Garraway, and W. R. Sellers, "Next-generation characterization of the Cancer Cell Line Encyclopedia," *Nature*, vol. 569, no. 7757, pp. 503–508, 2019. [Online]. Available: https://www.nature.com/articles/s41586-019-1186-3

[3] I. Subramanian, S. Verma, S. Kumar, A. Jere, and K. Anamika, "Multi-omics Data Integration, Interpretation, and Its Application," *Bioinform. Biol. Insights*, vol. 14, pp. 1–24, jan 2020. [Online]. Available: http://journals.sagepub.com/doi/10.1177/1177932219899051

[4] Y. Zheng, "Methodologies for Cross-Domain Data Fusion: An Overview," *IEEE Trans. Big Data*, vol. 1, no. 1, pp. 16–34, 2015. [Online]. Available: http://ieeexplore.ieee.org/document/7230259/

[5] M. Zitnik, F. Nguyen, B. Wang, J. Leskovec, A. Goldenberg, and M. M. Hoffman, "Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities," *Inf. Fusion*, vol. 50, pp. 71–91, oct 2019. [Online]. Available: http://arxiv.org/abs/1807.00123 https://linkinghub.elsevier.com/retrieve/pii/S1566253518304482

[6] L. Zhang and S. Zhang, "A General Joint Matrix Factorization Framework for Data Integration and Its Systematic Algorithmic Exploration," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 9, pp. 1971–1983, sep 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8762181/

[7] Z. Yang and G. Michailidis, "A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data," *Bioinformatics*, vol. 32, no. 1, pp. 1–8, 2015. [Online]. Available: https://academic.oup.com/bioinformatics/article/32/1/1/1743821

[8] M. Stražar, M. Žitnik, B. Zupan, J. Ule, and T. Curk, "Orthogonal matrix factorization enables integrative analysis of multiple RNA binding proteins," *Bioinformatics*, vol. 32, no. 10, pp. 1527–1535, may 2016. [Online]. Available: https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btw003

[9] C. Ding, T. Li, and M. I. Jordan, "Convex and Semi-Nonnegative Matrix Factorizations for Data Clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, 2010. [Online]. Available: https://ieeexplore.ieee.org/document/4685898

[10] V. H. Duong, W. C. Hsieh, P. T. Bao, and J. C. Wang, "An overview of kernel based nonnegative matrix factorization," *IEEE Int. Conf. Orange Technol. ICOT 2014*, vol. 2, pp. 227–231, 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6956641

[11] D. Zhang and W. Liu, "An Efficient Nonnegative Matrix Factorization Approach in Flexible Kernel Space," *Proc. Twenty-First Int. Jt. Conf. Artif. Intell.*, pp. 1345–1350, 2009. [Online]. Available: https://www.ijcai.org/Proceedings/09/Papers/226.pdf

[12] Y. Li and A. Ngom, "A new Kernel non-negative matrix factorization and its application in microarray data analysis," *IEEE Symp. Comput. Intell. Comput. Biol. CIBCB 2012*, pp. 371–378, 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6217254

[13] A. E. Páez-Torres and F. A. González, "Online kernel matrix factorization," *Lect. Notes Comput. Sci.*, vol. 9423, pp. 651–658, 2015. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-25751-8_78

[14] H. Yang and C. Seoighe, "Impact of the choice of normalization method on molecular cancer class discovery using nonnegative matrix factorization," *PLoS ONE*, vol. 11, no. 10, pp. 1–17, 2016. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0164880

[15] R. Zdunek, "Convex Nonnegative Matrix Factorization with Rank-1 Update for Clustering," 2015, pp. 59–68. [Online]. Available: http://link.springer.com/10.1007/978-3-319-19369-4_6

[16] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-View Clustering via Joint Nonnegative Matrix Factorization," in *Proceedings of the 2013 SIAM International Conference on Data Mining*. Philadelphia, PA: Society for Industrial and Applied Mathematics, may 2013, pp. 252–260. [Online]. Available: https://epubs.siam.org/doi/10.1137/1.9781611972832.28

[17] Y.-X. Wang and Y.-J. Zhang, "Nonnegative Matrix Factorization: A Comprehensive Review," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1336–1353, jun 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6165290/

[18] S. A. Vavasis, "On the Complexity of Nonnegative Matrix Factorization," *SIAM J. Optim.*, vol. 20, no. 3, pp. 1364–1377, jan 2010. [Online]. Available: http://epubs.siam.org/doi/10.1137/070709967

[19] V. Gligorijevic, N. Malod-Dognin, and N. Przulj, "Patient-specific data fusion for cancer stratification and personalised treatment," *Biocomput. 2016*, vol. 21, pp. 321–332, 2016. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/9789814749411_0030

[20] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proc. Natl. Acad. Sci.*, vol. 101, no. 12, pp. 4164–4169, mar 2004. [Online]. Available: https://www.pnas.org/content/101/12/4164

[21] G. Yu, L.-G. Wang, Y. Han, and Q.-Y. He, "clusterProfiler: an R Package for Comparing Biological Themes Among Gene Clusters," *Omi. A J. Integr. Biol.*, vol. 16, no. 5, pp. 284–287, may 2012. [Online]. Available: http://www.liebertpub.com/doi/10.1089/omi.2011.0118

[22] G. Zhou and J. Xia, "OmicsNet: a web-based tool for creation and visual analysis of biological networks in 3D space," *Nucleic Acids Res.*, vol. 46, no. W1, pp. W514–W522, jul 2018. [Online]. Available: https://academic.oup.com/nar/article/46/W1/W514/5033998

[23] A. Krämer, J. Green, J. Pollard, and S. Tugendreich, "Causal analysis approaches in Ingenuity Pathway Analysis," *Bioinformatics*, vol. 30, no. 4, pp. 523–530, feb 2014. [Online]. Available: https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btt703

[24] R. Oughtred, C. Stark, B.-J. Breitkreutz, J. Rust, L. Boucher, C. Chang, N. Kolas, L. O'Donnell, G. Leung, R. McAdam, F. Zhang, S. Dolma, A. Willems, J. Coulombe-Huntington, A. Chatr-aryamontri, K. Dolinski, and M. Tyers, "The BioGRID interaction database: 2019 update," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D529–D541, 2019. [Online]. Available: https://academic.oup.com/nar/article/47/D1/D529/5204333

[25] A. Franceschini, D. Szklarczyk, S. Frankild, M. Kuhn, M. Simonovic, A. Roth, J. Lin, P. Minguez, P. Bork, C. von Mering, and L. J. Jensen, "STRING v9.1: protein-protein interaction networks, with increased coverage and integration," *Nucleic Acids Res.*, vol. 41, no. D1, pp. D808–D815, nov 2012. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23203871 http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3531103 http://academic.oup.com/nar/article/41/D1/D808/1057425/STRING-v91-proteinprotein-interaction-networks

[26] G. Sales, E. Calura, D. Cavalieri, and C. Romualdi, "graphite - a Bioconductor package to convert pathway topology to gene network," *BMC Bioinformatics*, vol. 13, no. 1, p. 20, 2012. [Online]. Available: http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-13-20

[27] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, "limma powers differential expression analyses for RNA-sequencing and microarray studies," *Nucleic Acids Res.*, vol. 43, no. 7, p. e47, apr 2015. [Online]. Available: http://academic.oup.com/nar/article/43/7/e47/2414268/limma-powers-differential-expression-analyses-for

[28] X. Meng, J. Wang, C. Yuan, X. Li, Y. Zhou, R. Hofestädt, and M. Chen, "CancerNet: a database for decoding multilevel molecular interactions across diverse cancer types," *Oncogenesis*, vol. 4, no. 12, pp. e177–e177, dec 2015. [Online]. Available: http://www.nature.com/articles/oncsis201540

[29] L. Wei, Z. Jin, S. Yang, Y. Xu, Y. Zhu, and Y. Ji, "TCGA-assembler 2: software pipeline for retrieval and processing of TCGA/CPTAC data," *Bioinformatics*, vol. 34, no. 9, pp. 1615–1617, may 2018. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/29272348 http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC5925773 https://academic.oup.com/bioinformatics/article/34/9/1615/4764001

[30] K. Amin, "The role of mast cells in allergic inflammation," *Respir. Med.*, vol. 106, no. 1, pp. 9–14, jan 2012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0954611111003325

[31] T. Kambayashi and G. A. Koretzky, "Proximal signaling events in FcRI-mediated mast cell activation," *J. Allergy Clin. Immunol.*, vol. 119, no. 3, pp. 544–552, mar 2007. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0091674907001790

[32] F. Sadeghi and M. Shirkhoda, "Allergy-Related Diseases and Risk of Breast Cancer: The Role of Skewed Immune System on This Association," *Allergy Rhinol.*, vol. 10, p. 215265671986082, jan 2019. [Online]. Available: http://journals.sagepub.com/doi/10.1177/2152656719860820

[33] R. Kozłowska, A. Bożek, and J. Jarząb, "Association between cancer and allergies," *AAsthma Clin. Immunol.*,

vol. 12, no. 1, p. 39, dec 2016. [Online]. Available: http://aacijournal.biomedcentral.com/articles/10.1186/s13223-016-0147-8

[34] D. Cohen-Steiner and A. C. de Vitis, "Spectral Properties of Radial Kernels and Clustering in High Dimensions," jun 2019. [Online]. Available: http://arxiv.org/abs/1906.10583

[35] P. K. Sharma and G. Holness, "L2-norm transformation for improving k-means clustering," *Int. J. Data Sci. Anal.*, vol. 3, no. 4, pp. 247–266, jun 2017. [Online]. Available: http://link.springer.com/10.1007/s41060-017-0054-1

[36] S. Zhang, Q. Li, J. Liu, and X. J. Zhou, "A novel computational framework for simultaneous integration of multiple types of genomic data to identify microrna-gene regulatory modules," *Bioinformatics*, vol. 27, no. 13, pp. 401–409, 2011. [Online]. Available: https://academic.oup.com/bioinformatics/article/27/13/i401/177697

[37] P. Honeine and C. Richard, "Preimage Problem in Kernel-Based Machine Learning," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 77–88, mar 2011. [Online]. Available: http://ieeexplore.ieee.org/document/5714388/

DIEGO SALAZAR is a Ph.D. in Engineering candidate. He received a BA degree in pharmacy from the National University of Colombia in 2011. In 2015, he received MBiolSci from the Pontifical Xavierian University. He currently works as a lecturer in statistics in the Industrial Engineering department at the University of Los Andes. He has expertise in statistical learning, linear statistical models, and data mining for decision-making. Also, he has worked in clinics and the pharmaceutical industry. His research interests include causal inference, multi-modal machine learning, kernel-based models, and cancer.

JUAN RIOS studied industrial engineering with a minor in computer science at the University of Los Andes, and graduated in 2020. Throughout his career, he was a mentor on subjects such as statistics, programming, and since 2019 he has worked for a global consulting firm as a data and customer behavior analyst for multiple worldwide companies.

SARA ACEROS studied industrial and biomedical engineer at the University of Los Andes also received a master's degree in an industrial engineer from the same university. She was a graduate teaching assistant in statistics courses at the University of Los Andes and her research interests include machine learning, statistics, and mathematics methods to integrate and analyze biological data.

OSCAR FLÓREZ-VARGAS earned his Ph.D. in computer science from the University of Manchester, United Kingdom, in 2016, and his M.Sc. in biochemistry from the National University of Colombia in 2008. He is a post-doctoral fellow at the National Cancer Institute, United States of America. His research focuses on computational analysis of genotype-phenotype relationships to further evaluate the contribution of specific genomic regions, genes, and genetic variants to cancer phenotypes.

CARLOS VALENCIA is an Associate Professor in the Industrial Engineering department at the University of Los Andes. He holds a Ph.D. in Industrial Engineering with specialization in statistics from the H. Milton Steward School of Industrial and Systems Engineering, at the Georgia Institute of Technology; and two M.Sc. degrees in statistics and engineering. His research interest is in applied statistics and the use of regularization mechanisms for functional estimation in complex and high-dimensional settings. His research has been published in different journals related to statistics, data analysis, and simulation.