Encoder-Decoder Nonnegative Matrix Factorization with β-Divergence for Data Clustering

Sayvan Soleymanbaigi[a], Amjad Seyedi[b], Fardin Akhlaghian Tab[a], Fatemeh Daneshfar[a,*]

[a] *Department of Computer Engineering, University of Kurdistan, Sanandaj, Iran*
[b] *Department of Mathematics and Operational Research, University of Mons, Mons, Belgium*

Abstract

Nonnegative Matrix Factorization (NMF), as a group representation learning model, produces part-based representation with interpretable features that can be applied to various problems, such as data clustering. The findings indicate that the NMF model with β divergence (β-NMF) performs excellently in clustering different data types and noise assumptions. However, existing NMF-based data clustering methods are defined within a latent decoder model, lacking a verification mechanism. Recently, self-representation techniques have been applied to a wide range of tasks, empowering models to autonomously learn and verify representations that faithfully reflect the intricacies and nuances inherent in their input data. This paper proposes a self-representation factorization model for data clustering that incorporates local information into its learning process. The Regularized Encoder-Decoder NMF model based on β divergence (β-REDNMF) integrates encoder and decoder factorizations into a β cost function that mutually verify and refine each other, resulting in the formation of more distinct clusters. To incorporate the local information into the method, we add a graph regularization to the model. The β-REDNMF, owing to its autoencoder-like architecture and utilization of local information, produces more informative word embeddings with generalization abilities that apply to various data types. We present an efficient and effective optimization algorithm based on multiplicative update rules to solve the proposed unified model. The experimental results on the ten well-known datasets show that the proposed β-REDNMF model outperforms other state-of-the-art data clustering methods.

Keywords: data representation, nonnegative matrix factorization, auto-encoder structure, β-divergence

*Corresponding author
Email addresses: s.soleymanbaigi@uok.ac.ir (Sayvan Soleymanbaigi), seyedamjad.seyedi@umons.ac.be (Amjad Seyedi), f.akhlaghian@uok.ac.ir (Fardin Akhlaghian Tab), f.daneshfar@uok.ac.ir (Fatemeh Daneshfar)

1. Introduction

The rapid advancement of technology has led to a huge increase in the amount and complexity of data in many fields. This trend, known as the "big data" era, offers great opportunities but also poses significant challenges for researchers [1]. The wealth of information embedded within high-dimensional datasets holds immense promise for advancing knowledge discovery and driving innovation. Applications such as image recognition [2] and text analysis [3] showcase the profound impact of efficiently utilizing high-dimensional data. Nevertheless, the inherent complexities of high-dimensional data also present significant analytical challenges. The existence of irrelevant or redundant features can cause ambiguous patterns and hinder precise analysis. High-dimensional data, often characterized by complex relationships, redundancy, and noise, pose significant challenges for analysis, processing, and interpretation [4]. Dimensionality reduction and representation learning are crucial in addressing these challenges, extracting meaningful information, and facilitating downstream tasks such as classification [5], clustering [6], and pattern recognition [7]. Matrix decomposition methods have become powerful tools for reducing dimensionality and representing data effectively. These approaches decompose a high-dimensional data matrix into factor matrices of lower dimensions, effectively capturing latent structures and relationships within the data. These techniques, such as Principal Component Analysis (PCA), Singular Value Decomposition (SVD), and Nonnegative Matrix Factorization (NMF) [8], are utilized across pattern recognition [9], social network analysis [10], and data clustering [11] domains.

NMF is a dimension reduction and data representation technique proposed by Lee and Seung [8]. Its applications are vast, ranging from hyperspectral image unmixing [12] and text clustering [13] to blind source separation [14]. To further enhance NMF's performance, researchers have proposed various improvements, often incorporating constraints or regularization. NMF's sparsity, while beneficial for interpretability, is a byproduct rather than a controlled feature, limiting direct manipulation of the representation's properties. Hoyer et al. [15] proposed NMF to incorporate a feature enabling direct control over sparsity. Kim and Park [16] extended the NMF framework by integrating sparsity regularizers like L_1-norm regularization, enhancing the interpretability of the resulting factorization. Cai et al. [17] introduced graph-regularized NMF (GNMF) considering data's intrinsic geometric structure, while Zeng et al. [18] proposed hypergraph-regularized NMF (HNMF) for exploring higher-order information. Shang et al. [19] proposed Dual graph regularized NMF (DNMF), which incorporates both data and feature manifolds using two distinct graph regularizations. This approach enhances clustering and dimensionality reduction by preserving the

intrinsic structures of both data and feature spaces. Graph Regularized Discriminative Nonnegative Matrix Factorization (GDNMF) [20] integrates manifold learning, sparse representation, and label information into a unified framework. It aims to enhance clustering by considering the local geometrical structure and class label information through adaptive graph construction and regularization constraints, resulting in improved discriminative representation of data. A theoretical examination [21] indicates that there exists a correspondence between NMF and K-means or spectral clustering methods. So, Ding et al. introduce orthogonal nonnegative matrix tri-factorization (ONMTF) [21] to enhance clustering interpretations, and Orthogonal Graph Regularized NMF (OGNMF) [22] proposed the extend of ONMF, which consider data geometric structure and matrix orthogonality.

Many of these NMF variants employ the Frobenius norm (Euclidean distance) for measuring matrix similarity, assuming a Gaussian data distribution. However, real-world data often deviates from this assumption, particularly in the presence of outliers. Different applications and data types use various cost functions to measure NMF approximation quality. The Kullback-Leibler divergence, commonly referred to as KL-divergence, functions as a cost metric within the context of NMF [23], aligning with the concept of additive Poisson noise [24]. Research has demonstrated that Non-negative Matrix Factorization with Kullback-Leibler divergence (NMFk) is analogous to Probabilistic Latent Semantic Indexing (PLSI) [25], and it has found application in text clustering. Regularized Asymmetric NMF (RANMF) [26] uses NMFk as the main objective function and the similarity between documents as regularized constraints. NMF with Itakura-Saito loss excelling at capturing subtle variations in areas like audio and spectral analysis and obtaining Gamma likelihood [24]. Norm $\mathcal{L}_{21}$ [27] and correntropy [28] are another losses used in NMF to enhance robustness against outliers.

However, each cost function effectively manages one type of data and noise. To overcome the drawbacks of conventional cost functions and enhance the flexibility and robustness of NMF, Févotte et al. [29] explored the use of β-divergence as the NMF cost function. This generalized divergence measure encompasses multiple cost functions, including the Frobenius norm and KL divergence as special cases. The parameter β controls the shape of the divergence function, allowing it to adapt to various data distributions and noise models. This approach handles various data formats, including audio, images, and text. Consequently, Shi et al. [30] introduced Symmetric NMF, known as β-SymNMF, leveraging β-divergence to address nonlinear data more effectively than β-NMF. In a similar, Robust NMF with β-divergence (β-RNMF) [31], enhancing algorithmic robustness compared to both β-NMF and β-SymNMF. Furthermore, BO-βNMF [32] employs

biorthogonal regularized β-NMF to ensure feature orthogonality post-dimensionality reduction and sparsity in data representation.

Self-representation involves learning to reconstruct the input data from a latent representation, forcing the model to capture the most informative features. In the encoder-decoder models, this concept is expanded by introducing both an encoder and a decoder model. This self-representation model integrates both terms into a unified cost function to extract a more general representation. The encoder part condenses the data into a lower-dimensional latent space, compressing it while still capturing important characteristics. Subsequently, the decoder part reconstructs the original data from this condensed representation, ensuring that vital information remains intact throughout the dimensionality reduction phase. In the context of encoder-decoder NMF [33], the decoder part is a basic NMF that reconstructs the original data matrix from a multiplicative of factors matrix (i.e. $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}$), and the encoder part transforms the original data matrix to latent representation space using basis matrix (i.e. $\boldsymbol{H} \approx \boldsymbol{W}^\top \boldsymbol{X}$).

By substituting the encoder approximation into the decoder approximation, we obtain $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{X}$, which is equivalent to the projective NMF model. This formulation reveals that the model is inherently a feature-side self-representation approach, where each feature is represented as a linear combination of other features. The projective nature of the model ensures that the learned representations capture the most relevant feature relationships while maintaining non-negativity constraints, making the results more interpretable and meaningful for various downstream tasks such as clustering, feature selection, and dimensionality reduction.

Despite the clear advantages provided by the interpretable part-based representation in the methods mentioned above, and the incorporation of various techniques [34], there is a noticeable lack of research exploring the combined effects of self-representation and β-divergence within these methods. This gap in research specifically pertains to understanding how the combination of self-representation and β-divergence impacts crucial factors such as model interpretability, robustness, generalization, and adaptability to diverse data types. This study aims to fill this gap by introducing an innovative self-representation β-NMF model with various noise assumptions. This model not only achieves strong performance in data clustering with excellent generalization capabilities but also addresses the challenge of optimizing the encoder-decoder β-NMF across different β values. In doing so, it seeks to bridge the divide between model performance and encoder-decoder optimization within the framework of β-NMF.

β-divergence loss is crucial in self-representation models, like the Encoder-Decoder factorization, because these models rely on accurately capturing feature relationships, making the choice of

divergence measure essential. Since self-representation expresses each feature as a combination of others, any noise or distortion can propagate through the structure, and β-divergence's adaptability to different noise distributions mitigates this effect. Additionally, self-representation models encounter varying scales of feature importance, and β-divergence's flexibility allows the model to balance contributions dynamically. In projective formulations ($\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{W}^\top \boldsymbol{X}$), where the weight matrix is applied twice, errors can be magnified, making an appropriate divergence measure critical for maintaining stability and meaningful transformations. Finally, because self-representation models must capture both direct and indirect feature relationships, β-divergence provides the necessary flexibility to model complex interdependencies, ensuring robustness and improved reconstruction quality.

To achieve this robust and structured factorization, this paper introduces a novel unsupervised data representation method called β-REDNMF (Regularized Encoder-Decoder NMF with β-divergence), which integrates an encoder-decoder architecture within the NMF framework, leveraging β-divergence to optimize clustering performance. By coupling the encoder and decoder modules, the model refines cluster representations, leading to more accurate and robust feature extraction. However, self-representation models struggle to learn meaningful representations without regularization, leading to degenerate representations that fail to capture the data structure [34]. To address this, we incorporate manifold regularization, which preserves the local geometric structure of the data by enforcing smoothness along the underlying data manifold. This ensures that data points belonging to the same cluster remain closer in the latent space, improving both clustering separability and interpretability. The method further introduces an optimization scheme based on multiplicative updates, ensuring convergence across different β values while maintaining stability and adaptability to diverse data distributions. Through this integration of self-representation, manifold constraints, and β-divergence, our approach provides a more structured, scalable, and noise-robust solution for unsupervised data learning.

The key contributions of this paper are summarized as follows:

- We present an innovative self-representation model that utilizes β-divergence to support different types of noise assumptions.
- This encoder-decoder NMF method with β divergence loss, despite its generalization capabilities, demonstrates robustness across different data types.
- The proposed model incorporates manifold regularization as local geometric information to improve the performance of the method and achieve more separate data representation.

- To address the optimization challenge associated with this model, we propose novel multiplicative updating rules for the proposed cost function optimization. These updating rules are designed to cover different β values.
- To evaluate the efficiency of the β-EDNMF model, extensive experiments are conducted on various data type datasets.

This paper has the following structure: First, the backgrounds of basic NMF, NMF with β divergence, and Encoder-Decoder NMF are introduced in Section 2. In Section 3, the proposed models and their numerical solutions are presented. The experimental results that show the effectiveness of our method are provided in Section 4. Finally, the conclusion will be provided in Section 5.

2. Background

This section introduces some preliminaries including basic NMF, NMF with β-divergence, and Encoder-Decoder NMF models. In this paper, we use capital bold letters (like $\boldsymbol{X}$) for matrices, lowercase bold letters (like $\boldsymbol{x}$) for vectors, and regular letters (like a) for scalars. We also use $\boldsymbol{x}_i, \boldsymbol{x}^{(j)}$, and X_{ij} to mean the i-th column vector, the j-th row vector, and the element in the i-th row and j-th column of matrix $\boldsymbol{X}$, respectively. In addition, β-divergence and Frobenius norm are denoted by $D_\beta(.\|.)$ and $\|.\|_F$, respectively.

2.1. NMF

NMF [8] is a data representation and dimensionality reduction technique widely used in various fields such as signal processing [14], text mining [26], and image analysis [7]. It decomposes a nonnegative data matrix into two lower-dimensional nonnegative matrices: a basis matrix and a coefficient matrix. The basis matrix represents a set of latent features or components, while the coefficient matrix indicates the contribution of each feature to the original data points.

Let's consider a nonnegative data matrix $\boldsymbol{X}$ of size $p \times n$, where n represents the number of samples or observations, and p represents the number of features. The goal of NMF is to factorize this matrix into two non-negative matrices, $\boldsymbol{W}$ and $\boldsymbol{H}$, such that $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}$, where $\boldsymbol{W}$ is of size $p \times k$ and $\boldsymbol{H}$ is of size $k \times n$, and k is the desired reduced dimensionality. Mathematically, the objective function of NMF can be formulated as [8]:

$$\min_{\boldsymbol{W},\boldsymbol{H}} ||\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}||_F^2 \quad \text{s.t.} \quad \boldsymbol{W} \geq 0, \quad \boldsymbol{H} \geq 0 \tag{1}$$

where $||\cdot||_F$ denotes the Frobenius norm, which measures the magnitude of the matrix. The constraints ensure that both $\boldsymbol{H}$ and $\boldsymbol{W}$ contain only nonnegative elements, making the factorization interpretable in terms of parts-based representations.

The optimization problem can be solved using various iterative algorithms, such as multiplicative update rules, alternating least squares (ALS), or gradient descent methods. One of the most commonly used algorithms for NMF is the multiplicative update algorithm, which iteratively updates the elements of $\boldsymbol{W}$ and $\boldsymbol{H}$ until convergence. The update rules for multiplicative NMF are given by [8]:

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{\boldsymbol{X}\boldsymbol{H}^\top}{\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^\top}, \tag{2}$$

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \frac{\boldsymbol{W}^\top\boldsymbol{X}}{\boldsymbol{W}^\top\boldsymbol{W}\boldsymbol{H}}. \tag{3}$$

2.2. β-NMF

β-NMF is a variant of the traditional NMF that introduces a parameter β to control the sparsity of the factor matrices [31]. Similar to NMF, β-NMF aims to approximate a nonnegative data matrix $\boldsymbol{X}$ with two nonnegative matrices $\boldsymbol{W}$ and $\boldsymbol{H}$, such that $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}$, but it allows for a more flexible representation by handling different data types. In this context, a and b represent two nonnegative scalar values drawn from two distributions (e.g., observed and predicted values), which are compared using the β-divergence, as described below:

$$D_\beta(a,b) = \begin{cases} \frac{a}{b} - \log\frac{a}{b} - 1 & \text{for } \beta = 0, \\ a\log\frac{a}{b} - a + b & \text{for } \beta = 1, \\ \frac{1}{\beta(\beta-1)}\left(a^\beta + (\beta-1)b^\beta - \beta a b^{\beta-1}\right) & \text{for } \beta \neq 0, 1. \end{cases} \tag{4}$$

The objective function of β-NMF can be formulated as [29],

$$D_\beta(\boldsymbol{X}, \boldsymbol{W}\boldsymbol{H}) = \sum_{i=1}^{p}\sum_{j=1}^{n} D_\beta(X_{ij}, [WH]_{ij}). \tag{5}$$

where $D_\beta(\boldsymbol{X}||\boldsymbol{W}\boldsymbol{H})$ is the β-divergence between $\boldsymbol{X}$ and $\boldsymbol{W}\boldsymbol{H}$. The β-divergence is a family of divergence measures that includes the Kullback-Leibler divergence ($\beta = 1$) [35] and the Itakura-Saito divergence ($\beta = 0$) [36] as special cases. By varying the parameter β, one can control the

sparsity level of the factor matrices.

The optimization problem for β-NMF can be solved using similar iterative algorithms as NMF, such as multiplicative update rules. The update rules for β-NMF are derived based on the gradient of the objective function with respect to the factor matrices W and H. The update rules for W and H in β-NMF are given by,

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{[\boldsymbol{W}\boldsymbol{H}]^{\odot(\beta-2)} \odot \boldsymbol{X}\boldsymbol{H}^\top}{[\boldsymbol{W}\boldsymbol{H}]^{\odot(\beta-1)}\boldsymbol{H}^\top} \tag{6}$$

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \frac{\boldsymbol{W}^\top[\boldsymbol{W}\boldsymbol{H}]^{\odot(\beta-2)} \odot \boldsymbol{X}}{\boldsymbol{W}^\top[\boldsymbol{W}\boldsymbol{H}]^{\odot(\beta-1)}} \tag{7}$$

2.3. Encoder-Decoder NMF

The previous NMF model has limitations. It relies solely on the decoder's ability to reconstruct the input data, ignoring the encoder's potential for learning representations. This means the entire process must be repeated for every new, unseen data point. We can modify the loss function to involve the encoder, as suggested in [37] to achieve this. This creates a unified loss function include both the encoder and the decoder, leading to a new model called Encoder-Decoder NMF. Here's the loss function of the Encoder-Decoder NMF using the Frobenius norm:

$$\min_{\boldsymbol{W},\boldsymbol{H}\geq 0} \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{H}\|_F^2 + \|\boldsymbol{H} - \boldsymbol{W}^\top\boldsymbol{X}\|_F^2, \tag{8}$$

The model (8) can address out-of-sample data by directly evaluating whether the data matrix $\boldsymbol{X}$ can be feasibly mapped to the representation $\boldsymbol{H}$ via the mapping matrix $\boldsymbol{W}$. This formulation ensures that the learned matrix $\boldsymbol{W}$, given sufficient samples of $\boldsymbol{X}$, is not only optimized for the given data but also generalizes to unseen data $\boldsymbol{X}'$, as it enforces a consistent and effective transformation between $\boldsymbol{X}$ and $\boldsymbol{H}$. Consequently, the learned mapping matrix $\boldsymbol{W}$ can handle new, unseen data without the need to repeat the entire process, improving both efficiency and generalizability. To solve the above optimization, Sun et al. [33] obtain multiplicative updating rules to update $\boldsymbol{W}$ and $\boldsymbol{H}$:

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{2\boldsymbol{X}\boldsymbol{H}^\top}{\boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^\top + \boldsymbol{X}\boldsymbol{X}^\top\boldsymbol{W}} \tag{9}$$

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \frac{2\boldsymbol{W}^\top \boldsymbol{X}}{\boldsymbol{W}^\top \boldsymbol{W}\boldsymbol{H} + \boldsymbol{H}} \tag{10}$$

Encoder-Decoder NMF offers a powerful framework for learning compact representations of complex data while preserving important characteristics. It has been successfully applied in various domains, including image processing [9], community detection [33], natural language processing (NLP) [37], where capturing underlying structures in the data is crucial for tasks such as feature extraction, denoising, and dimensionality reduction.

3. Proposed Model

This section introduces the Regularized Encoder-Decoder NMF model with β-divergence (β-REDNMF), a tailored factorization technique designed for addressing data clustering challenges. Its effectiveness primarily stems from four key elements: (1) Utilization of β-divergence loss, rendering it resilient against diverse noise distributions. (2) Incorporation of an encoder NMF term within the β-divergence-based framework alongside the Decoder NMF, refining and validating the factorization process, thereby enhancing representation performance. (3) Integration of manifold regularization into the proposed Encoder-Decoder β-NMF to augment local information during factorization. (4) Consolidation of the aforementioned aspects into a unified joint learning problem and adoption of an efficient alternating minimization strategy for optimization.

3.1. Encoder–Decoder β-NMF

Given a nonnegative data matrix $\boldsymbol{X} \in \mathbb{R}^{p\times n}$, each row of input corresponds to a feature and each column corresponds to a sample. β-NMF (5) tries to discover two nonnegative matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ where $\boldsymbol{W} \in \mathbb{R}^{p\times c}$ is a feature representation matrix and $\boldsymbol{H} \in \mathbb{R}^{c\times n}$ is a sample representation matrix. To enhance quality of representation and handle out-of-sample, β-EDNMF introduces encoder part which transform data matrix $\boldsymbol{X}$ into the sample representation $\boldsymbol{H}$ using feature representation matrix $\boldsymbol{W}$, with β divergence loss function as follows:

$$D_\beta(\boldsymbol{H}, \boldsymbol{W}^\top \boldsymbol{X}) = \frac{1}{\beta(\beta-1)} \sum_{k=1}^{r} \sum_{j=1}^{n} \left(H_{kj}^\beta + (\beta-1)[W^\top X]_{kj}^\beta - \beta H_{kj} [W^\top X]_{kj}^{\beta-1} \right) \tag{11}$$

β-EDNMF combines encoder (11) and decoder (5) parts as unified cost function as follows:

$$D_\beta(\boldsymbol{X}, \boldsymbol{WH}) + \lambda D_\beta(\boldsymbol{H}, \boldsymbol{W}^\top \boldsymbol{X}) = \frac{1}{\beta(\beta-1)} \sum_{i=1}^{p} \sum_{j=1}^{n} \left(X_{ij}^\beta + (\beta-1)[WH]_{ij}^\beta - \beta X_{ij} [WH]_{ij}^{\beta-1} \right)$$
$$+ \lambda \frac{1}{\beta(\beta-1)} \sum_{k=1}^{r} \sum_{j=1}^{n} \left(H_{kj}^\beta + (\beta-1)[W^\top X]_{kj}^\beta - \beta H_{kj} [W^\top X]_{kj}^{\beta-1} \right) \tag{12}$$

where the parameter λ controls the importance of self-expression representation.

3.2. Graph Regularized Encoder–Decoder β-NMF

To consider the locally invariant idea and the data smoothness property, the geometrical information of high-dimensional data is commonly incorporated as a regularization term. This involves encoding the information as a similarity graph, where nodes represent data and edges indicate the similarity between them. In this context, nearby data in the original space tend to be close to each other in the manifold space as well. Therefore, we further employ this term to regularize the model. We use the Gaussian similarity function to transform the distance to a similarity measurement and construct the similarity graph based on the p-nearest neighbor graph as follows,

$$\boldsymbol{S}_{ij} = \begin{cases} e^{-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{\sigma^2}}, & \text{if } \boldsymbol{x}_i \in \mathcal{N}_p(\boldsymbol{x}_j) \\ 0, & \text{otherwise,} \end{cases} \tag{13}$$

where S_{ij} is the similarity weight between samples i and j, and $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are the m-dimensional feature vectors from the set of features. We can use the Frobenius norm to compute the dissimilarity between feature vectors of samples. Considering the data similarity matrix $\boldsymbol{S}$, we can apply the following equation to compute dissimilarity among data feature vectors.

$$\begin{aligned} \mathcal{R} &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\boldsymbol{h}_i - \boldsymbol{h}_j\|^2 S_{ij} \\ &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\boldsymbol{h}_i^\top \boldsymbol{h}_i - 2\boldsymbol{h}_i^\top \boldsymbol{h}_j + \boldsymbol{h}_j^\top \boldsymbol{h}_j) S_{ij} \\ &= \sum_{i=1}^{n} \boldsymbol{h}_i^\top \boldsymbol{h}_i D_{ii} - \sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{h}_i^\top \boldsymbol{h}_j S_{ij} \\ &= \mathrm{Tr}(\boldsymbol{HDH}^\top) - \mathrm{Tr}(\boldsymbol{HSH}^\top) = \mathrm{Tr}(\boldsymbol{HLH}^\top) \end{aligned} \tag{14}$$

where $D_{jj} = \sum_{i=1}^{n} S_{ij}$ and the Laplacian graph $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{S}$. Combining this regularizer with the β-divergence cost function results in the proposed β-REDNMF. Given a non-negative data matrix, the proposed β-REDNMF tries to discover two non-negative matrices by considering the above constraints. The proposed β-REDNMF uses β-divergence and minimizes the cost function as:

$$\begin{aligned}\mathcal{L}(\boldsymbol{W},\boldsymbol{H}) &= D_\beta(\boldsymbol{X},\boldsymbol{W}\boldsymbol{H}) + \lambda D_\beta(\boldsymbol{H},\boldsymbol{W}^\top\boldsymbol{X}) + \gamma\mathcal{R} \\ &= \frac{1}{\beta(\beta-1)}\sum_{i=1}^{p}\sum_{j=1}^{n}\left(X_{ij}^\beta + (\beta-1)[WH]_{ij}^\beta - \beta X_{ij}[WH]_{ij}^{\beta-1}\right) \\ &+ \frac{\lambda}{\beta(\beta-1)}\sum_{k=1}^{r}\sum_{j=1}^{n}\left(H_{kj}^\beta + (\beta-1)[W^\top X]_{kj}^\beta - \beta H_{kj}[W^\top X]_{kj}^{\beta-1}\right) \\ &+ \gamma\mathrm{Tr}(\boldsymbol{H}\boldsymbol{L}\boldsymbol{H}^\top)\end{aligned} \tag{15}$$

where the hyperparameter γ controls the contribution of local information.

3.3. Optimization

The objective function (15) is non-convex, and thus quite challenging to solve. To obtain the optimal solution of the β-REDNMF objective function, we apply the multiplicative update rules of $\mathcal{L}_{\beta-\mathrm{REDNMF}}$ to optimize $\boldsymbol{W}$, $\boldsymbol{H}$ alternatively and iteratively. The constraint set for the objective function includes $\boldsymbol{W} \geq 0$, $\boldsymbol{H} \geq 0$. Therefore, the Lagrange function $\mathcal{L}$ is given by introducing Lagrange multipliers $\boldsymbol{\Psi}, \boldsymbol{\Phi}$ where $\boldsymbol{\Psi} = [\Psi_{ij}]$, and $\boldsymbol{\Phi} = [\Phi_{ij}]$. Finally, the Lagrange function of the β-REDNMF is,

$$\begin{aligned}\mathcal{L}(\boldsymbol{W},\boldsymbol{H}) =& \frac{1}{\beta(\beta-1)}\sum_{i=1}^{p}\sum_{j=1}^{n}\left(X_{ij}^\beta + (\beta-1)[WH]_{ij}^\beta - \beta X_{ij}[WH]_{ij}^{\beta-1}\right) \\ &+ \frac{\lambda}{\beta(\beta-1)}\sum_{k=1}^{r}\sum_{j=1}^{n}\left(H_{kj}^\beta + (\beta-1)[W^\top X]_{kj}^\beta - \beta H_{kj}[W^\top X]_{kj}^{\beta-1}\right) \\ &+ \gamma\mathrm{Tr}(\boldsymbol{H}\boldsymbol{L}\boldsymbol{H}^\top) - Tr(\Psi\boldsymbol{W}^\top) - Tr(\Phi\boldsymbol{H}^\top)\end{aligned} \tag{16}$$

The first-order partial derivatives w.r.t $\boldsymbol{W}$ and $\boldsymbol{H}$ are

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial W_{ik}} =& -\sum_{j=1}^{n} H_{kj}X_{ij}[WH]_{ij}^{\beta-2} + \sum_{j=1}^{n} H_{kj}[WH]_{ij}^{\beta-1} \\ &+ \lambda(-\sum_{j=1}^{n} X_{ij}H_{kj}[W^\top X]_{kj}^{\beta-2} + \sum_{j=1}^{n} X_{ij}[W^\top X]_{kj}^{\beta-1}) - \psi_{ik},\end{aligned} \tag{17}$$

$$\frac{\partial \mathcal{L}}{\partial H_{kj}} = -\sum_{i=1}^{p} W_{ik} X_{ij} [WH]_{ij}^{\beta-2} + \sum_{i=1}^{p} W_{ik} [WH]_{ij}^{\beta-1} + \lambda\left(\frac{-[W^\top X]_{kj}^{\beta-1} + H_{kj}^{\beta-1}}{\beta-1}\right) + \gamma\left(-\sum_{t=1}^{n} S_{tj}(H_{kj} - H_{kt})\right) - \phi_{kj}, \tag{18}$$

Based on the Karush-Kuhn-Tucker (KKT) conditions $\psi_{ik} W_{ik} = 0$, and $\phi_{kj} H_{kj} = 0$, we obtain all the updating rules, where they are written in matrix form as follows:

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{\boldsymbol{X} \odot [\boldsymbol{WH}]^{\beta-2}\boldsymbol{H}^\top + \lambda \boldsymbol{H} \odot [\boldsymbol{W}^\top \boldsymbol{X}]^{\beta-2}\boldsymbol{X}^\top}{[\boldsymbol{WH}]^{\beta-1}\boldsymbol{H}^\top + \lambda [\boldsymbol{W}^\top \boldsymbol{X}]^{\beta-1}\boldsymbol{X}^\top}, \tag{19}$$

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \frac{\boldsymbol{W}^\top \boldsymbol{X} \odot [\boldsymbol{WH}]^{\beta-2} + \frac{\lambda}{\beta-1}[\boldsymbol{W}^\top \boldsymbol{X}]^{\beta-2} + \gamma \boldsymbol{HS}}{\boldsymbol{W}^\top [\boldsymbol{WH}]^{\beta-1} + \frac{\lambda}{\beta-1}\boldsymbol{H}^{\beta-1} + \gamma \boldsymbol{HD}} \tag{20}$$

where $\odot$ indicates the Hadamard product. We find from equation 20 that it lacks definition when β equals 1. Consequently, we suggest a tailored updating rule for $\boldsymbol{H}$ when β is set to 1, derived from the β-divergence cost function. The cost function in $\beta = 1$ as KL-divergence can be considered as follows

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H} \geq 0} \mathcal{L} &= \mathcal{D}_{kl}(\boldsymbol{X} \| \boldsymbol{WH}) + \lambda \mathcal{D}_{kl}(\boldsymbol{H} \| \boldsymbol{W}^\top \boldsymbol{X}) + \gamma \mathcal{R} \\ &= \sum_{i=1}^{p}\sum_{j=1}^{n} X_{ij} \log \frac{X_{ij}}{[WH]_{ij}} - X_{ij} + [WH]_{ij} \\ &+ \lambda \sum_{k=1}^{c}\sum_{j=1}^{n} H_{kj} \log \frac{H_{kj}}{[W^\top X]_{kj}} - H_{kj} + [W^\top X]_{kj} \\ &+ \gamma \mathrm{Tr}(\boldsymbol{HLH}^\top) \end{aligned} \tag{21}$$

By establishing its Lagrangian function, taking the derivative, and adhering to the KKT conditions, we can derive the updating rule for the $\boldsymbol{H}$ matrix as follows

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \frac{\boldsymbol{W}^\top \frac{\boldsymbol{X}}{\boldsymbol{WH}} + \lambda \log(\boldsymbol{W}^\top \boldsymbol{X}) + \gamma \boldsymbol{HS}}{\boldsymbol{W}^\top \boldsymbol{1} + \lambda \log(\boldsymbol{H}) + \gamma \boldsymbol{HD}} \tag{22}$$

The optimization process of β-REDNMF is provided in Algorithm 1. The proof of convergence under β-divergence and encoder-decoder factorizations is well-established in the literature and can be found in [29, 37].

Algorithm 1 Regularized Encoder Decoder β-NMF (β-REDNMF)

Input: data matrix $\boldsymbol{X}$, number of cluster c, scale parameter λ, regularizer parameter γ;
Output: cluster-sample matrix $\boldsymbol{H}$;

Calculate similarity matrix $\boldsymbol{S}$ according to (13);
Initialize $\boldsymbol{W}$ and $\boldsymbol{H}$ matrices randomly;
while Convergence **do**
  Update feature representation matrix $\boldsymbol{W}$ according to (19);
  if $\beta \neq 1$ **then**
    Update sample representation matrix $\boldsymbol{H}$ according to (20)
  else
    Update sample representation matrix $\boldsymbol{H}$ according to (22)
  end if
end while
return $\boldsymbol{W}$ and $\boldsymbol{H}$;

3.4. Complexity Analysis

In the Regularized Encoder-Decoder β-NMF (β-REDNMF) algorithm (Algorithm 1), the computational complexity is dominated by the iterative optimization stage, which consists of t iterations. Updating the matrix $\boldsymbol{W}$ requires $O(dnrt)$ operations, while updating $\boldsymbol{H}$ involves $O(dnrt + n^2rt)$. Consequently, the overall complexity of the β-REDNMF algorithm is $O(dnrt + n^2rt)$, where d is the feature dimension, n is the number of samples, r is the latent dimension, and t is the number of iterations. The algorithm scales linearly with the feature and latent dimensions but quadratically with the number of samples due to the manifold regularization term in the update of $\boldsymbol{H}$. To contextualize the computational efficiency of our method, we compare the complexities of several related algorithms. The computational complexity of standard NMF [8] is $O(ndrt)$, RANMF [26] has $O(\max(nd, n^2)rt)$, DNMF [19] has $O(d^2n + n^2d + dnrt)$, and OEDF [37] has $O(\max(n^2, d^2)rt)$. Thus, β-REDNMF achieves competitive efficiency, scaling well with feature and latent dimensions, and remains comparable to existing methods. It is also worth noting that the convergence speed of the algorithm depends on the number of iterations. In section 4.7, empirical results show that β-REDNMF converges rapidly, which keeps the total computational cost of the update process relatively low.

4. Experimental Results

In this section, a comprehensive series of experiments is undertaken to assess the efficacy of the proposed model relative to 13 baseline models and state-of-the-art models across 10 benchmark datasets. These experiments include result analyses, parameter examinations, and ablation studies, all of which are thoroughly expounded upon.

4.1. Datasets

We conduct experiments across ten datasets to demonstrate the superior performance capabilities of β-REDNMF when compared to other methods under consideration. The BBCNews dataset encompasses news articles from the BBC News website, categorized into five topics: Business, Entertainment, Politics, Sport, and Tech, comprising 2225 documents spanning the years 2004 to 2005. The Reuters-21578 dataset comprises 21578 texts sourced from the Reuters news-wire in 1987. We use two variants of the Reuters-21578 dataset, denoted as Reuters4c and Reuters10c, featuring four and ten clusters, respectively, comprising 7632 and 9979 texts. The AGNews dataset comprises news articles sourced from diverse channels, categorized into four primary topics: World, Sports, Business, and Sci/Tech. The Yale face dataset, as described by Belhumeur et al. (1997), comprises images of 15 distinct individuals, each photographed under varying conditions, such as center-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink. Each image undergoes scaling to dimensions of 32 $\times$ 32 pixels in our experimental setup. The Coil20 dataset, developed by Columbia University, comprises a total of 1440 grayscale images featuring 20 distinct objects. Each object is represented by 72 images, all possessing dimensions of 32 $\times$ 32 pixels. The images are captured at intervals of five degrees and rotated on a turntable. The ORL dataset comprises 400 facial images, each corresponding to the visage of 40 individuals. Within this dataset, each person is represented by 10 images, showcasing variations in pose and expression, including variations such as open or closed eyes and smiling or neutral expressions. The MNIST dataset comprises images of handwritten digits sourced from American high school students. These images are standardized to a format of 28 $\times$ 28 pixels with grayscale values. The dataset encompasses a total of 70,000 images, covering digits from 0 to 9. To construct a subset, we randomly select 100 samples from each class, resulting in a new dataset containing 1000 images. The Fashion-MNIST dataset consists of 70,000 samples featuring a variety of clothing items such as shirts, trousers, pullovers, dresses, coats, sandals, sneakers, bags, and ankle boots. Each sample is represented as a grayscale image with dimensions of 28 $\times$ 28 pixels, annotated with labels from 10 distinct classes. To generate a subset, we randomly select 100 samples from each class, resulting in a new dataset containing 1000 images.

4.2. Evaluation metrics

To evaluate the performance of our proposed method, we employed the following metrics,

- Accuracy (ACC): is a measure of how often the clustering algorithm correctly classifies data points into their respective clusters. It is defined as the ratio of the number of correctly

Table 1: The detailed of the real-world datasets.

Dataset	#sample	#feature	#class
BBCNews	2225	2000	5
Reuters4c	7632	500	4
Reuters10c	9979	1000	10
AGNews	7600	1000	4
Coil20	1440	1024	20
ORL	400	1024	40
Yale	165	1024	15
MNIST	1000	784	10
FashionMNIST	1000	784	10
Isolet	1560	617	26

assigned labels to the total number of data points. This metric provides a straightforward assessment of the clustering performance but may not account for the complexity of the data distribution.

- Normalized Mutual Information (NMI): is a metric used to measure the similarity between two clustering results. It is particularly useful for comparing the true labels with the labels predicted by the clustering algorithm. NMI is defined based on the Mutual Information (MI) between the true label distribution and the predicted label distribution, normalized by the average of the entropy of these distributions. It ranges from 0 to 1, where 1 indicates perfect clustering.

- Adjusted Rand Index (ARI): is a measure of the similarity between two data clusterings that takes into account the chance grouping of elements. It is an adjusted version of the Rand Index (RI), which counts the pairs of elements that are either in the same or different clusters in the predicted and true clusterings. The ARI adjusts for the chance grouping by considering the expected RI of random clusterings. ARI values range from -1 to 1, with 1 indicating perfect agreement between the clustering results and 0 representing random labeling.

These metrics provide a comprehensive evaluation of the clustering performance, accounting for both the accuracy of label assignment and the similarity between the predicted and true data structures.

4.3. Compared methods

In this section, we compare our proposed method with several base and state-of-the-art approaches in the realm of NMF and its variants. The comparative analysis focuses on their

clustering performance and computational efficiency. The methods selected for comparison are as follows:

- **NMF**: This traditional method decomposes a non-negative data matrix into two lower-dimensional non-negative matrices, aiming to find a parts-based representation of the data. It uses the Frobenius norm as the divergence measure and is widely employed in various applications [8].

- **NMFk**: utilizes NMF with Kullback-Leibler divergence (KL-divergence) for evaluating the quality of the approximation [23].

- **β-NMF** (β Divergence NMF): An extension of NMF, β-NMF utilizes the β-divergence as a flexible measure to handle different data distributions and noise levels. By adjusting the β parameter, this method can control the sparsity and robustness of the factorization process [29].

- **GNMF** (Graph Regularized NMF): GNMF incorporates graph regularization into the NMF framework, leveraging the local geometric structure of the data. This method ensures that similar data points in the original space remain close in the lower-dimensional space, thereby enhancing clustering performance [17].

- **HNMF** (Hypergraph-Regularized NMF): HNMF integrates hypergraph regularization into Decoder NMF by constructing a hypergraph to effectively capture and utilize higher-order relationships among data points, enhancing feature learning and representation [18].

- **DNMF** (Dual regularization NMF): DNMF is a dual-regularization graph-based model that integrates the structural relationships within both the data and feature manifolds, ensuring a comprehensive representation of their geometric properties [19].

- **EDNMF** (Encoder-Decoder NMF): A recent advancement in NMF methodologies, EDNMF integrates both encoder and decoder processes within the NMF framework. This approach not only reconstructs the data but also learns robust representations, leading to improved generalization and clustering performance [33].

- **SeaNMF** (Semantic assisted NMF): maps the term-document matrix and semantic correlation matrix into a shared term-cluster space for topic modeling [38]

- **WRNMTF** (Word Regularized Non-negative Matrix Tri-Factorization): WRNMTF performs word clustering and document clustering, by a regularized Nonnegative Matrix

Tri-Factorization [39].

- **RANMF** (Regularized Asymmetric NMF): RANMF adds graph regularization to NMFk, which learns the semantic features of documents [26].
- **DGLCF** (Dual-graph Global and Local Concept Factorization): DGLCF introduces a novel approach to data clustering by leveraging the strengths of graph-based techniques and concept factorization to better reveal the complex inner manifold [40].
- **BO-βNMF** (Bi-orthogonal β-NMF): BO-βNMF optimizes a bi-orthogonal regularization incorporating β-divergence NMF, which allows for more robust and flexible factorization [32].
- **OEDFS** (Orthogonal Encoder-Decoder Feature Selection): OEDFS integrates feature selection into the NMF framework, focusing on discriminative features that enhance clustering performance [37].

By comparing our method against these established approaches across multiple benchmark datasets, we aim to demonstrate the superior performance and robustness of β-REDNMF in diverse clustering scenarios.

4.4. Results

In this section, we present a comprehensive evaluation of the proposed β-REDNMF method in comparison with several baseline models and state-of-the-art techniques across ten benchmark datasets. We run each method 10 times and report the mean and standard deviation. In Tables 2-4, the best results are marked by bold, and the second-best ones are marked by underline. The results presented in Table 2 highlight the effectiveness of our β-REDNMF compared to various state-of-the-art clustering techniques across multiple datasets, evaluated using the NMI metric. On the BBCNews dataset, our method achieves an NMI of 0.8609, outperforming all other methods, including β-NMF (0.8368) and NMFk (0.7990). This demonstrates our method's superior capability in accurately capturing the underlying cluster structure in text data. For the AGNews dataset, β-REDNMF significantly surpasses others with an NMI of 0.4390. In contrast, the closest competing method, β-NMF, achieves an NMI of 0.4178, indicating a clear advantage of our approach in handling diverse news categories. On the Reuters4C and Reuters10C datasets, our method also excels with NMIs of 0.6072 and 0.5598, respectively. These results are notably higher than those achieved by RANMF (0.5306 and 0.5279) and β-NMF (0.5466 and 0.5441), showcasing our method's robustness in clustering different types of document collections. In image datasets

Table 2: NMI outcomes for ten datasets. The top-performing result is showcased in **bold** format, while the second-best is indicated with an underline.

Method		BBCNews	AGNews	Reuters4c	Reuters10c	Yale	Coil20	ORL	MNIST	Fashion-MNIST	Isolet
NMF	mean	0.7566	0.2652	0.3579	0.4778	0.4348	0.6579	0.7028	0.3897	0.5080	0.7345
	std	0.0591	0.0026	0.0040	0.0099	0.0208	0.0104	0.0114	0.0230	0.0118	0.0113
NMFk	mean	0.7990	0.3906	0.5466	0.5175	0.4739	0.6753	0.6841	0.3772	0.4926	0.7203
	std	0.0800	0.0614	0.0107	0.0348	0.0378	0.0172	0.0033	0.0270	0.0223	0.0080
β-NMF	mean	0.8368	0.4178	0.5466	0.5441	0.5029	0.6753	0.7028	0.4017	0.5098	0.7345
	std	0.0571	0.0371	0.0107	0.0109	0.0406	0.0172	0.0119	0.0113	0.0103	0.0114
GNMF	mean	0.7890	0.2569	0.3179	0.3867	0.4895	0.7292	0.6723	0.4328	0.5468	0.8001
	std	0.0031	0.0042	0.0377	0.0309	0.0374	0.0295	0.0023	0.0145	0.0110	0.0073
HNMF	mean	0.7783	0.2592	0.3330	0.4575	0.4917	0.7460	0.7739	0.4328	0.5506	0.7810
	std	0.0387	0.0781	0.0090	0.0425	0.0735	0.0093	0.0188	0.0280	0.0016	0.0145
DNMF	mean	0.6616	0.3601	0.5684	0.5421	0.5288	**0.8343**	0.7904	0.4626	0.5547	**0.8175**
	std	0.0394	0.0331	0.0091	0.0166	0.0106	0.0002	0.0151	0.0054	0.0070	0.0011
EDNMF	mean	0.7407	0.2465	0.3577	0.4499	0.5138	0.7658	0.7214	0.4544	0.5283	0.7107
	std	0.0015	0.0383	0.0100	0.0356	0.0062	0.0109	0.0091	0.0187	0.01399	0.0105
SeaNMF	mean	0.7350	0.2713	0.3575	0.4818	-	-	-	-	-	-
	std	0.0651	0.0140	0.0021	0.0201	-	-	-	-	-	-
WRNMTF	mean	0.6752	0.2762	0.3595	0.4118	-	-	-	-	-	-
	std	0.0582	0.0140	0.0108	0.0431	-	-	-	-	-	-
RANMF	mean	0.8099	0.3982	0.5306	0.5279	0.4887	0.7212	0.8007	0.4143	0.5362	0.7646
	std	0.0568	0.0616	0.0554	0.0149	0.0237	0.0209	0.0133	0.0199	0.0151	0.0140
DGLCF	mean	0.7816	0.3152	0.4461	0.5007	0.4592	0.7834	0.7792	0.4216	0.5578	0.6495
	std	0.0420	0.0210	0.0212	0.0220	0.0100	0.0274	0.0128	0.0190	0.0301	0.0065
BO-βNMF	mean	0.8203	0.4007	0.5567	0.5482	0.4910	0.7376	0.7957	0.4215	0.5411	0.7767
	std	0.0659	0.0368	0.0334	0.0182	0.0197	0.0113	0.0141	0.0146	0.0208	0.0096
OEDFS	mean	0.7232	0.2621	0.4430	0.4683	0.5157	0.7711	0.7599	0.4174	0.5237	0.7263
	std	0.0924	0.0633	0.0989	0.0402	0.0191	0.0251	0.0088	0.0359	0.0292	0.0087
β-REDNMF	mean	**0.8609**	**0.4390**	**0.6072**	**0.5598**	**0.5560**	0.8038	**8033**	**0.4876**	**0.5783**	0.8001
	std	0.0013	0.0063	0.0258	0.0197	0.0187	0.0110	0.0068	0.0169	0.0127	0.0073

like Yale, Coil20, and ORL, our method maintains high performance with NMIs of 0.5560, 0.8038, and 0.8033, respectively. These scores are competitive with or exceed those of other methods, such as GNMF and EDNMF, highlighting the versatility and generalization ability of our approach in image clustering tasks. For large-scale datasets like MNIST and Fashion-MNIST, β-REDNMF achieves NMIs of 0.4876 and 0.5783, respectively. While these results are competitive, they also indicate areas for potential improvement, as some methods like DGCLF and β-NMF perform similarly well. However, our method still provides a strong performance, particularly in capturing the complex structure of handwritten digits and fashion items. On the Isolet dataset, our method achieves an NMI of 0.8001, which is one of the highest scores among the compared methods, demonstrating its effectiveness in clustering spoken letter data.

Table 3 presents the ARI scores for various clustering methods across ten benchmark datasets. The proposed β-REDNMF method consistently demonstrates superior performance compared to other state-of-the-art methods. It achieved the highest ARI score of 0.8900 on the BBCNews dataset, significantly outperforming other techniques such as NMFk (0.8087) and β-NMF (0.8474). This indicates a remarkable clustering quality and alignment with the true data structure. While most methods showed relatively lower performance on the AGNews dataset, our method achieved an ARI of 0.4723, highlighting a potential area for further improvement. For the Reuters-4C

Table 3: ARI outcomes for ten datasets. The top-performing result is showcased in **bold** format, while the second-best is indicated with an underline.

Method		BBCNews	AGNews	Reuters4c	Reuters10c	Yale	Coil20	ORL	MNIST	Fashion-MNIST	Isolet
NMF	mean	0.7623	0.2445	0.1456	0.3024	0.1757	0.4212	0.3916	0.2377	0.3259	0.5303
	std	0.1102	0.0156	0.0077	0.0315	0.0238	0.0149	0.0267	0.0207	0.0234	0.0341
NMFk	mean	0.8087	0.40666	0.5166	0.3746	0.2221	0.4294	0.3596	0.2091	0.3082	0.5183
	std	0.1156	0.0787	0.0435	0.0457	0.0394	0.0410	0.0193	0.0328	0.0241	0.0076
β-NMF	mean	0.8473	0.4292	0.5166	0.3663	0.2393	0.4294	0.3916	0.2383	0.3373	0.5303
	std	0.0950	0.0470	0.0435	0.0412	0.0476	0.0410	0.0267	0.0196	0.0221	0.0341
GNMF	mean	0.8210	0.2805	0.3576	0.3655	0.2081	0.4655	0.2327	0.2612	0.3608	0.6093
	std	0.0031	0.0046	0.0526	0.0702	0.0442	0.0346	0.0010	0.0226	0.0224	0.0136
HNMF	mean	0.8079	0.2673	0.3106	0.3210	0.2133	0.5467	0.4190	0.2827	0.3907	0.5949
	std	0.0781	0.0628	0.0774	0.0846	0.0189	0.0265	0.0427	0.0245	0.0107	0.0225
DNMF	mean	0.5805	0.3159	0.3931	0.4209	0.2817	**0.6722**	0.4424	0.3232	0.3974	**0.6234**
	std	0.0883	0.0627	0.0096	0.0545	0.0066	0.0013	0.0291	0.0094	0.0092	0.0052
EDNMF	mean	0.7717	0.2354	0.2068	0.2983	0.2411	0.6005	0.4351	0.3094	0.3818	0.4546
	std	0.0030	0.0196	0.0274	0.0516	0.0136	0.0269	0.0198	0.0262	0.0229	0.0206
SeaNMF	mean	0.7397	0.2515	0.1431	0.3087	-	-	-	-	-	-
	std	0.1060	0.0210	0.0051	0.0280	-	-	-	-	-	-
WRNMTF	mean	0.6337	0.2584	0.1421	0.2767	-	-	-	-	-	-
	std	0.0801	0.0680	0.0240	0.0610	-	-	-	-	-	-
RANMF	mean	0.8291	0.4104	0.4997	0.4214	0.2154	0.4565	0.4738	0.2238	0.3283	0.5942
	std	0.0761	0.0854	0.0718	0.0337	0.0317	0.0433	0.0274	0.0315	0.0671	0.0029
DGLCF	mean	0.8019	0.2715	0.2594	0.3481	0.1728	0.4727	0.4718	0.2734	0.4029	0.4770
	std	0.0750	0.0470	0.0590	0.0230	0.0094	0.1075	0.0215	0.0225	0.0296	0.0169
BO-βNMF	mean	0.8451	0.4292	0.4844	0.4167	0.2297	0.5174	0.4896	0.2642	0.3866	0.5721
	std	0.0881	0.0502	0.0400	0.0565	0.0176	0.0302	0.0326	0.0181	0.0216	0.0238
OEDFS	mean	0.7245	0.2521	0.3218	0.3053	0.2119	0.5963	0.4581	0.2459	0.3757	0.4349
	std	0.0595	0.0753	0.1766	0.0407	0.0302	0.0434	0.1821	0.0207	0.0243	0.0054
β-REDNMF	mean	**0.8900**	**0.4723**	**0.5942**	**0.4916**	**0.2991**	0.6328	**4918**	**0.3354**	**0.4057**	0.6093
	std	0.0013	0.0071	0.0574	0.0387	0.0289	0.0201	0.0145	0.0185	0.0206	0.0136

dataset, the OUR method recorded an ARI of 0.5942, surpassing β-NMF (0.5166) and BO-βNMF (0.4844). In the more complex Reuters-10C dataset, the β-REDNMF method excelled with an ARI of 0.4916, which is higher than the scores of NMFk (0.6746) and GNMF (0.3655). The Yale dataset presented significant challenges, but our method achieved an ARI of 0.2991, outperforming the best other method, EDNMF, which scored 0.2411. This indicates robust performance in facial image clustering tasks. On the Coil20 dataset, our method demonstrated competitive performance with an ARI of 0.6328, though methods like OEDFS (0.5963) and EDNMF (0.6005) performed slightly better, suggesting room for improvement in image clustering tasks. The method achieved an ARI of 0.4918 on the ORL dataset, showcasing its effectiveness in clustering complex datasets with high variability. This was notably higher than the performance of most other methods. For the MNIST dataset, the β-REDNMF method recorded an ARI of 0.3354, which is comparable to other top-performing methods like EDNMF (0.3094). On the Fashion-MNIST dataset, our method achieved an ARI of 0.4057, indicating solid performance in clustering challenging image datasets. The Isolet dataset results show the OUR method with an ARI of 0.6093, which is among the highest scores reported, indicating excellent performance in speech recognition tasks similar to GNMF.

Table 4 presents the clustering ACC of various methods on ten benchmark datasets. For the

Table 4: ACC outcomes for ten datasets. The top-performing result is showcased in **bold** format, while the second-best is indicated with an underline.

Method		BBCNews	AGNews	Reuters4c	Reuters10c	Yale	Coil20	ORL	MNIST	Fashion-MNIST	Isolet
NMF	mean	0.8726	0.4463	0.5245	0.4479	0.3945	0.5180	0.5575	0.4086	0.4978	0.6211
	std	0.0925	0.0095	0.0040	0.0313	0.0317	0.0276	0.0119	0.0168	0.0054	0.0383
NMFk	mean	0.8897	0.6802	0.7375	0.4719	0.4266	0.5527	0.5160	0.3894	0.4854	0.6070
	std	0.1066	0.0826	0.0447	0.0303	0.0348	0.0206	0.0180	0.0317	0.0332	0.0129
β-NMF	mean	0.9245	0.6877	0.7375	0.4854	0.4284	0.5527	0.5575	0.4188	0.4885	0.6211
	std	0.0007	0.0614	0.0447	0.0445	0.0304	0.0206	0.0119	0.0219	0.0116	0.0383
GNMF	mean	0.9007	0.4900	0.6599	0.4987	0.4169	0.5806	0.4575	0.4562	0.5251	0.6845
	std	0.0015	0.0175	0.0413	0.0551	0.0491	0.0425	0.0035	0.0220	0.0559	0.0175
HNMF	mean	0.9010	0.5025	0.6265	0.5150	0.4260	0.6372	0.5801	0.4866	0.5636	0.6608
	std	0.1147	0.0693	0.0311	0.0660	0.0242	0.0204	0.0352	0.0344	0.0197	0.0422
DNMF	mean	0.7765	0.6121	0.6936	0.5609	0.4323	**0.7453**	0.4758	0.5313	0.5553	**0.7024**
	std	0.0313	0.0545	0.0050	0.0122	0.0086	0.0006	0.0123	0.0092	0.0057	0.0120
EDNMF	mean	0.8922	0.4814	0.5854	0.4885	0.4254	0.6665	0.5351	0.4994	0.5328	0.5255
	std	0.0016	0.0043	0.0106	0.0263	0.0145	0.0295	0.0198	0.0187	0.0333	0.0258
SeaNMF	mean	0.8538	0.4676	0.5245	0.4565	-	-	-	-	-	-
	std	0.0940	0.0450	0.0041	0.0201	-	-	-	-	-	-
WRNMTF	mean	0.7823	0.4824	0.5203	0.3695	-	-	-	-	-	-
	std	0.0630	0.0701	0.0377	0.0290	-	-	-	-	-	-
RANMF	mean	0.9161	0.6765	0.7268	0.5300	0.4278	0.5904	0.6165	0.4112	0.5106	0.6837
	std	0.0545	0.0924	0.0391	0.0363	0.0390	0.0407	0.0236	0.0092	0.0921	0.0294
DGLCF	mean	0.9024	0.5552	0.5307	0.4861	0.3939	0.6025	0.6035	0.4718	0.5694	0.3840
	std	0.0620	0.0530	0.0210	0.0310	0.0148	0.0516	0.0198	0.0178	0.0232	0.0078
BO-βNMF	mean	0.9273	0.6869	0.7027	0.5080	0.4424	0.6066	0.6325	0.4764	0.5458	0.6484
	std	0.0700	0.0530	0.0372	0.0501	0.0273	0.0298	0.0151	0.0216	0.0121	0.0359
OEDFS	mean	0.8435	0.5643	0.6501	0.4955	0.4351	0.6547	0.5899	0.4686	0.5202	0.5498
	std	0.123	0.0741	0.1280	0.0286	0.0218	0.0187	0.0097	0.0205	0.0329	0.0147
β-REDNMF	mean	**0.9530**	**0.7523**	**0.7806**	**0.5763**	**0.4824**	0.6847	**0.6220**	**0.5460**	**0.5766**	0.6845
	std	0.0005	0.0067	0.0569	0.0495	0.0439	0.0274	0.0131	0.0282	0.0348	0.0175

BBC News dataset, our proposed method achieves an accuracy of 0.9530, which is significantly higher than the other methods. The closest competitor, β-NMF, scores 0.9245, indicating the effectiveness of β-REDNMF in handling text data. Similarly, on the Reuters-4C and Reuters-10C datasets, β-REDNMF attains accuracies of 0.7806 and 0.5763, respectively, outperforming all methods. These results highlight the robustness of OUR in clustering high-dimensional text data. On image datasets such as Yale, COIL20, and ORL, OUR also excels, with accuracies of 0.4824, 0.6847, and 0.6220, respectively. These results suggest that our proposed method can effectively capture complex patterns in visual data, leading to better clustering outcomes. For widely used datasets like MNIST and Fashion-MNIST, β-REDNMF achieves accuracies of 0.5460 and 0.5766, respectively. While some methods like EDNMF and DGCLF perform reasonably well on these datasets, the proposed method still maintains a competitive edge, showcasing its generalizability across different types of data.

Overall, the proposed β-REDNMF method exhibits strong and consistent performance across diverse benchmark datasets, confirming its generalizability and practical effectiveness in unsupervised learning tasks. Through an extensive comparative analysis with foundational methods such as β-NMF, GNMF, EDNMF, and state-of-the-art NMF methods, we demonstrate that our approach not only integrates and extends key ideas from spectral clustering and manifold learning but

also consistently outperforms or matches these methods in terms of clustering accuracy and data representation quality. These results highlight the advantages introduced by the encoder-decoder structure and graph regularization, reinforcing the value of β-REDNMF as a robust and scalable tool for high-dimensional data analysis.

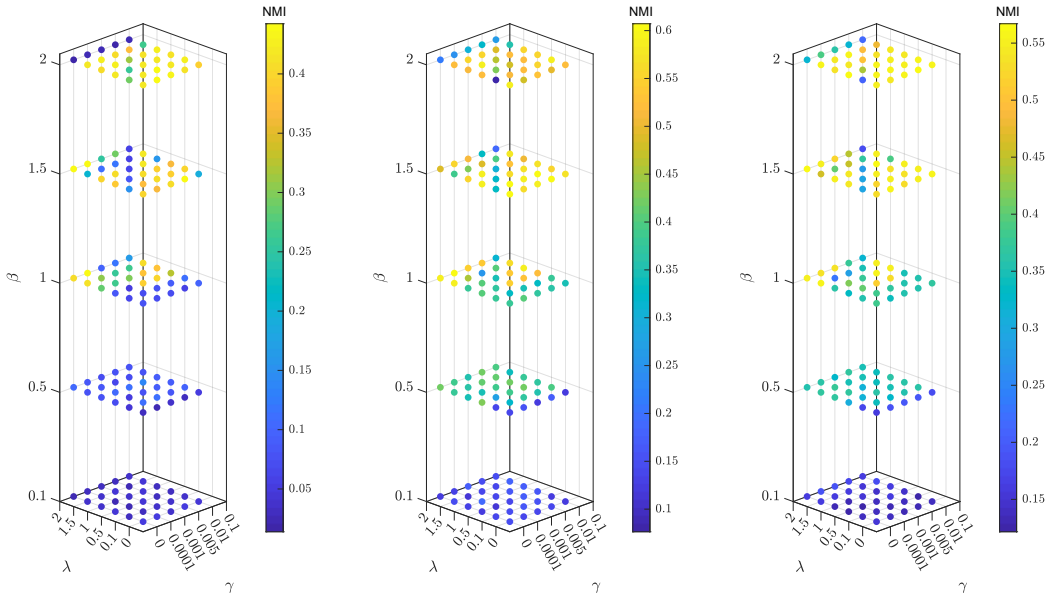
4.5. Parameter analysis

In this section, we analyze the influence of the hyperparameters on the clustering performance, where β, λ, and γ control the contribution of the cost function, self-expression, and local information, respectively. In a grid search, we analyze the simultaneous effect of both parameters on the proposed model. Figures 1-3 illustrate the proposed method's NMI, ACC, and ARI with various β, λ and γ on six datasets. Note that, this figure is 4D, i.e., three axes correspond to β, λ, and γ, and the color of the points reflects the fourth dimension. β parameter is the change parameter of the cost function in our model, and our selected values for analyzing this parameter on the six datasets are {0.1, 0.5, 1, 1.5, 2}. According to the results, it can be concluded that the optimal values for this parameter in the image datasets are 2 and 1.5. For text datasets, optimal values are 1 and 1.5. Also, we experimented with the proposed model with different values of λ from {0, 0.1, 0.5, 1, 1.5, 2}. For most datasets, larger λ usually leads to higher performance, which implies self-expression is more important. γ parameter controls graph regularization, and the values set for parameters are {0, 0.0001, 0.001, 0.005, 0.01, 0.1}. As we can observe in Fig.1-3, γ with small values usually performs better in terms of NMI, ACC, and ARI measures on the text datasets, but in image datasets, large values of γ can obtain better results.

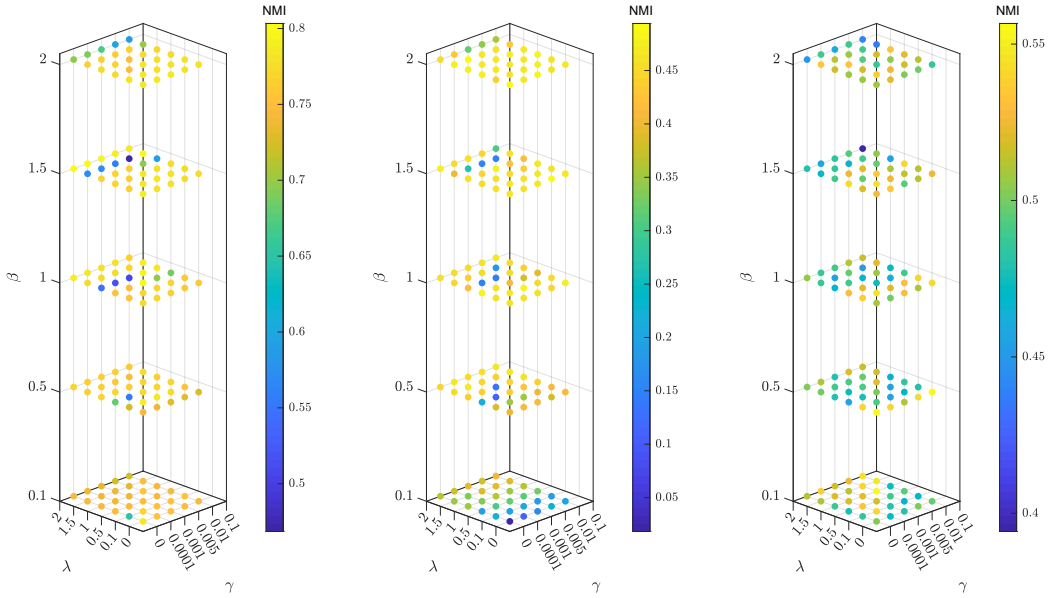
4.6. Ablation Study

To understand the contributions of different components of our proposed method, we conducted an ablation study by systematically removing or altering specific elements and observing the impact on performance. We evaluated the performance using three metrics: ACC, NMI, and ARI across several datasets: BBC News, AG News, Reuters-10C, Yale, COIL20, and Fashion-MNIST. The cases considered in the ablation study are as follows:

- Case I: Removing both components ($\lambda = 0$, $\gamma = 0$)
- Case II: Removing the first component ($\lambda = 0$)
- Case III: Removing the second component ($\gamma = 0$)
- Case IV: Full model (both components included)

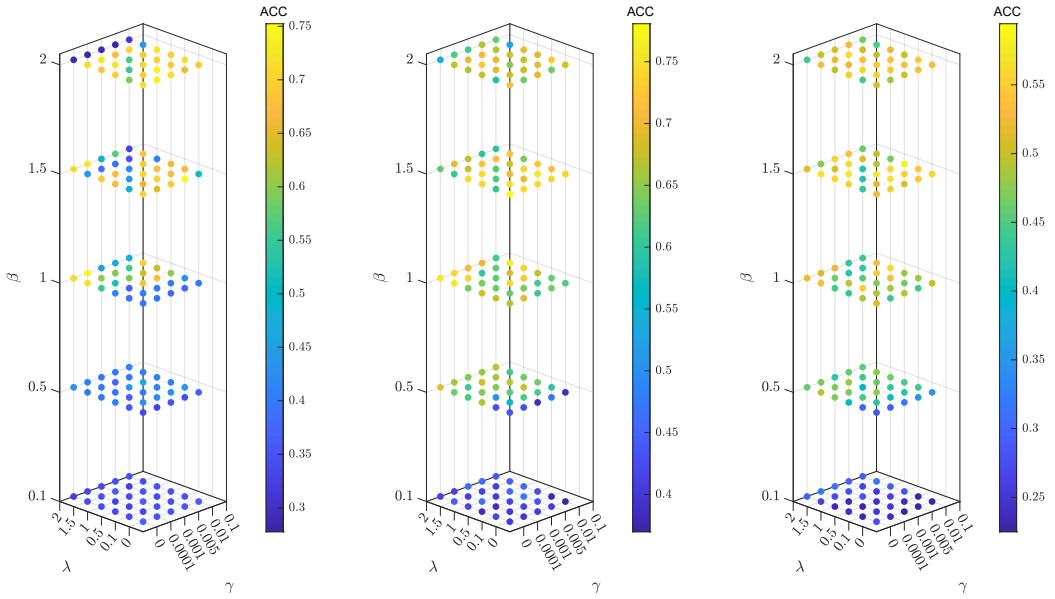


(a) AGNews (b) Reuters4c (c) Reuters10c

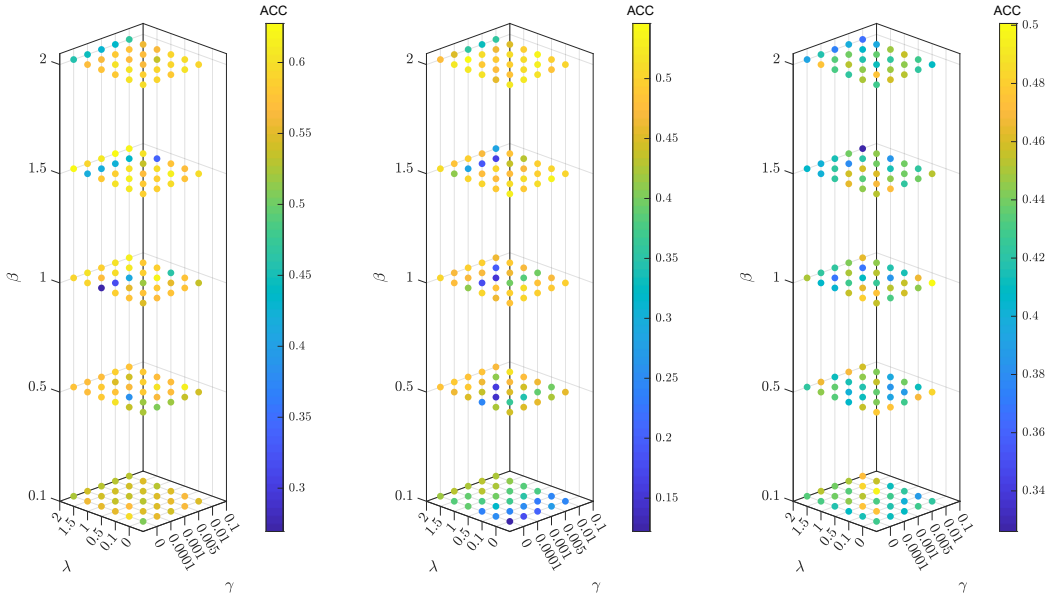


(d) ORL (e) MNIST (f) Yale

Figure 1: Parameter analysis (in terms of NMI) on the β, λ and γ parameters

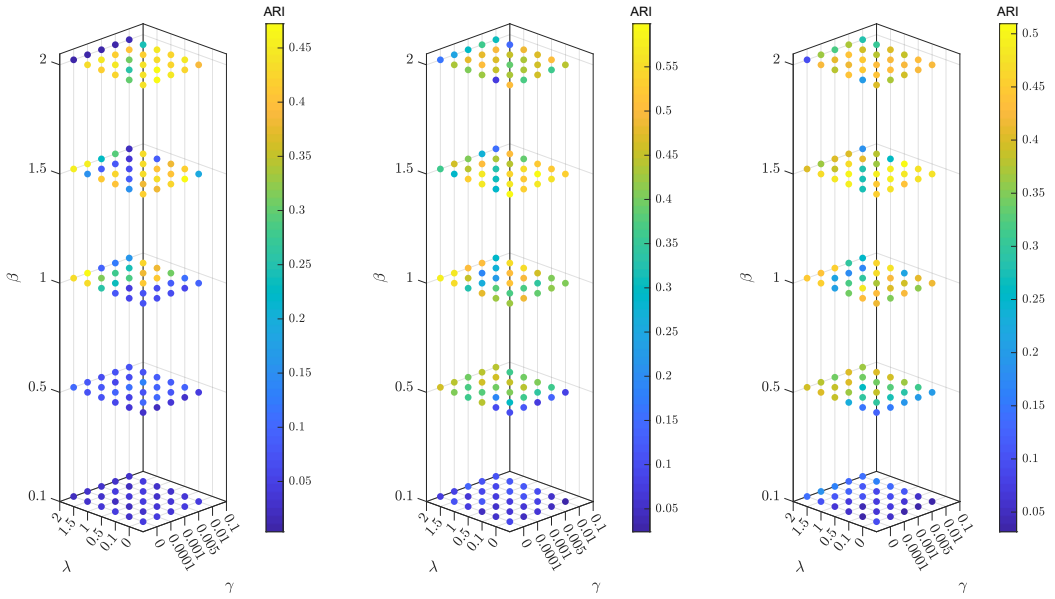


(a) AGNews (b) Reuters4c (c) Reuters10c

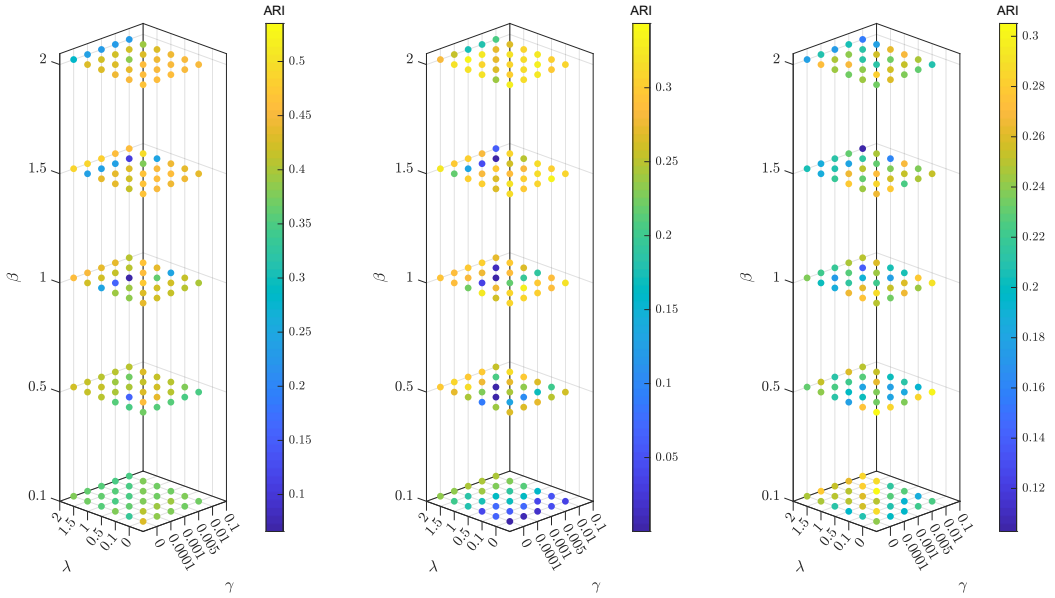


(d) ORL (e) MNIST (f) Yale

Figure 2: Parameter analysis (in terms of ACC) on the β, λ and γ parameters



(a) AGNews (b) Reuters4c (c) Reuters10c



(d) ORL (e) MNIST (f) Yale

Figure 3: Parameter analysis (in terms of ARI) on the β, λ and γ parameters

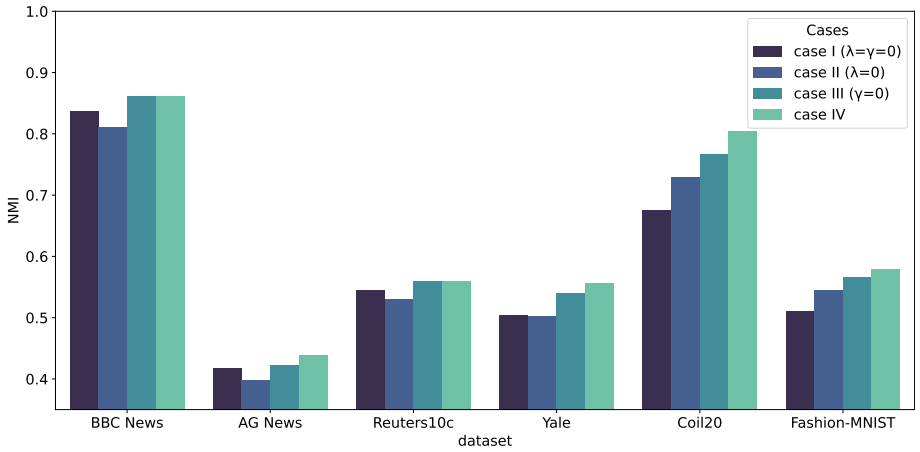


Figure 4: Ablation study on the effect of regularization terms of the proposed method based on NMI measure.

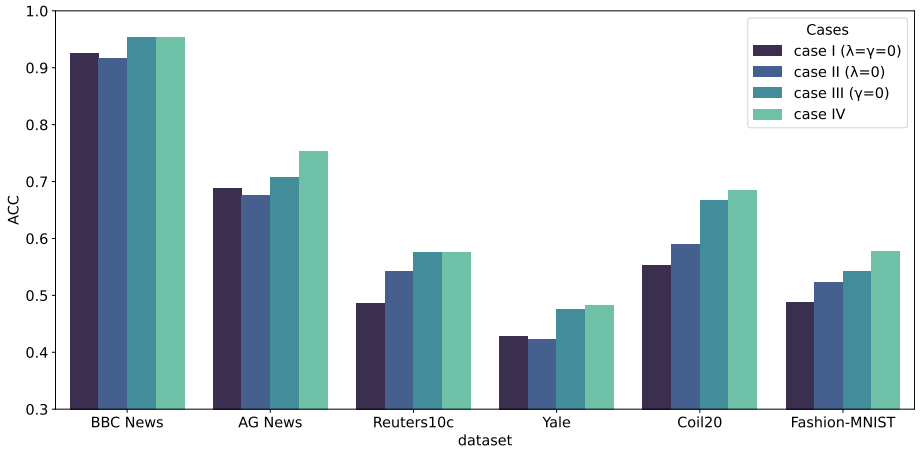


Figure 5: Ablation study on the effect of regularization terms of the proposed method based on ACC measure.

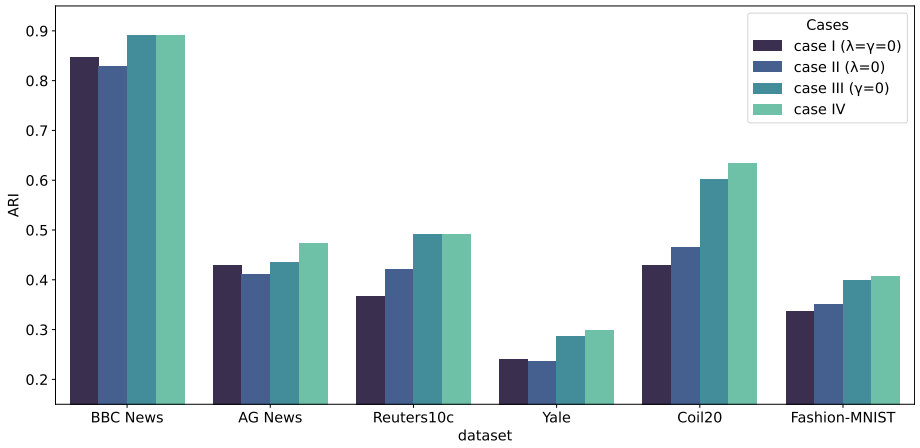


Figure 6: Ablation study on the effect of regularization terms of the proposed method based on ARI measure.

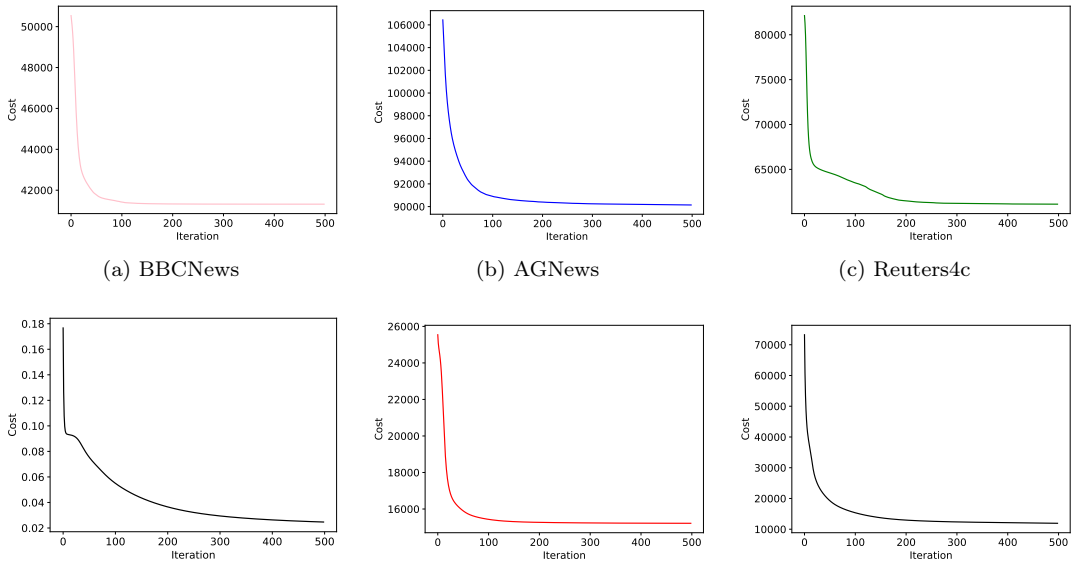
The results for ACC are summarized in Figure 4. The full model (Case IV) consistently outperforms the other cases across all datasets. For instance, in the BBCNews dataset, the full model achieves an accuracy close to 0.95, whereas Case I shows a significant drop in performance, highlighting the importance of both components in the model. The NMI results, shown in Figure 5, also indicate that the full model (Case IV) yields the best performance on all datasets. The improvement is particularly noticeable in the AGNews and Reuters-10C datasets. Cases II and III, where only one component is removed, show intermediate performance, suggesting that both components contribute considerably to the clustering quality. Figure 6 presents the ARI results. Similar to ACC and NMI, the full model (Case IV) consistently achieves the highest ARI scores, confirming its robustness and effectiveness. The performance drop in Case I across all datasets underscores the critical role of the omitted components. The ablation study clearly demonstrates the necessity of both components in our proposed method. The full model (Case IV) consistently outperforms the other cases across all evaluation metrics and datasets. This indicates that each component contributes uniquely and significantly to the overall performance of the model. Removing either component leads to a notable decline in clustering quality, highlighting their complementary roles in enhancing data representation and clustering accuracy.

The results of this study validate the design choices in our proposed method and underline the importance of a holistic approach to model development, where multiple synergistic components are integrated to achieve superior performance.

4.7. Convergence analysis

In this section, we conduct an empirical analysis of the convergence behavior of the RED-βNMF algorithm. Figure 7 illustrates how this method consistently minimizes the objective function in accordance with the update rules explained in Section 3.3. The diagrams within this figure depict the number of iterations on the X-axis and the corresponding objective function values on the Y-axis. Consequently, based on the observed convergence patterns across various datasets, it can be confidently concluded that this method exhibits convergence. The findings from Figure 7 indicate that the model converges rapidly, reaching convergence within a reasonable number of iterations.

Moreover, to analyze the impact of varying β values on the convergence behavior of the Beta-divergence REDNMF, Figure 8 presents the convergence curves corresponding to several β values, providing a comparative visual framework. As observed, the objective function values progressively decrease as the iterations advance for each β. The results show that lower β values (e.g., $\beta = 0.1$) lead to higher convergence rates, with the objective function converging to values



(a) BBCNews (b) AGNews (c) Reuters4c

(d) ORL (e) MNIST (f) Isolet

Figure 7: Convergence analysis of the proposed model on six datasets

around 62,000 at the first iteration and rapidly decreasing. In contrast, larger β values, such as $\beta = 2$, exhibit slower convergence, with the objective function starting at around 19,000 and gradually decreasing over the iterations. This suggests that while smaller β values result in faster convergence, larger values may offer more stability but at the cost of slower convergence. The trade-off between convergence speed and stability under different β values should be carefully considered depending on the specific requirements of the model.

Finally, to compare convergence analysis of the proposed method with other methods, Figure 9 shows the comparative convergence analysis of the NMF, GNMF, EDNMF, and the proposed method. While all methods exhibit a progressive reduction in error, their convergence rates and stability vary. Figure 9 demonstrates the superior performance of β–REDNMF compared to its ablated variants. The results show that β–REDNMF consistently achieves a lower objective value over iterations, significantly outperforming β–NMF ($\lambda = 0, \gamma = 0$), EDNMF ($\gamma = 0$), and GNMF ($\lambda = 0$). Specifically, EDNMF and GNMF exhibit slower convergence and higher final objective values, indicating suboptimal performance when either γ or λ is absent. The β–NMF ($\lambda = 0, \gamma = 0$) variant, which lacks both parameters, converges the slowest and maintains the highest objective value throughout, further underscoring the necessity of both components for efficient optimization. This dual-parameter approach appears to create a more favorable optimization landscape that facilitates faster convergence while simultaneously improving the quality of the

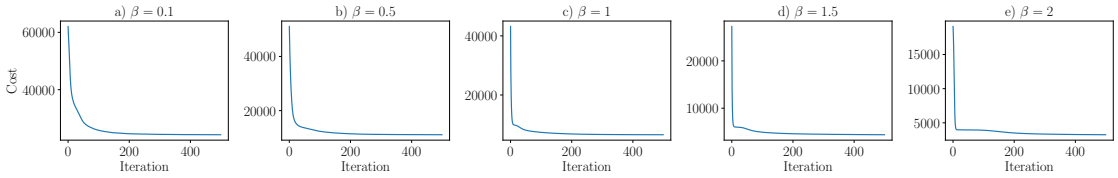


Figure 8: Convergence curves of the proposed method according to the different β values $\{0.1, 0.5, 1, 1.5, 2\}$

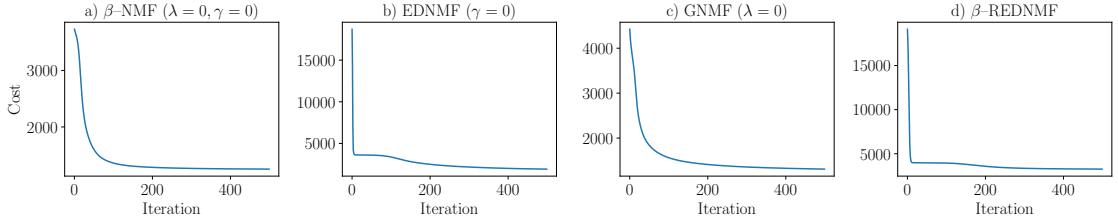


Figure 9: Convergence curves of NMF, EDNMF, GNMF, and β–REDNMF.

final solution. The consistent performance advantage observed across all iteration stages suggests that the proposed modifications not only accelerate the convergence process but also enhance the algorithm's ability to identify higher-quality local optima. These comprehensive findings provide strong empirical validation for the theoretical advantages of the proposed β-REDNMF formulation. The systematic ablation study conclusively demonstrates that both regularization parameters are essential for achieving optimal performance, as their individual or collective removal results in progressively degraded convergence behavior. The results establish that the proposed modifications represent genuine algorithmic improvements rather than marginal parameter tuning effects, thereby confirming the fundamental contribution of the dual regularization approach to enhanced matrix factorization performance. This evidence supports the adoption of β-REDNMF as a superior alternative to existing methods for applications requiring efficient non-negative matrix factorization with improved convergence guarantees.

5. Conclusion

In this paper, we proposed the Regularized Encoder-Decoder Nonnegative Matrix Factorization with β-divergence (β-REDNMF) model, a novel approach that enhances clustering performance by integrating encoder and decoder β-NMF modules within a unified cost function. By leveraging self-representation via encoder and decoder, the model effectively refines and verifies learned clusters while ensuring interpretability and robustness across diverse datasets. Additionally, the incorporation of graph-based regularization improves the model's ability to capture local geometric structures, making it well-suited for real-world data clustering tasks. The proposed

multiplicative update optimization strategy further ensures efficient convergence, allowing the model to maintain both accuracy and scalability. Experimental evaluations on multiple benchmark datasets demonstrated the superior performance of β-REDNMF compared to existing clustering methods.

Despite these strengths, β-REDNMF has certain limitations. As a shallow factorization model, it may struggle to capture hierarchical structures in highly complex datasets. Additionally, its reliance on graph-based regularization, while beneficial, may require careful tuning of hyper-parameters for optimal performance across different data types. Future research can explore several avenues to further enhance the capabilities of β-REDNMF. One potential direction is extending the model into a deep nonnegative matrix factorization framework, allowing it to better capture hierarchical and multi-level data representations. Moreover, incorporating additional side information, such as partial label supervision or feature correlations, could transform the model into a semi-supervised or multi-view learning approach. Beyond clustering, β-REDNMF could be adapted for broader applications, such as multi-view clustering, recommender systems, link prediction, and feature selection, where its self-representation structure could offer valuable insights.

Acknowledgements

Amjad Seyedi acknowledges the support by the European Union (ERC consolidator, eLinoR, No. 101085607).

References

[1] A. Adadi, A survey on data-efficient algorithms in big data era, Journal of Big Data 8 (1) (2021) 24.

[2] F. Daneshfar, B. S. Saifee, S. Soleymanbaigi, M. Aeini, Elastic deep multi-view autoencoder with diversity embedding, Information Sciences 689 (2025) 121482.

[3] P. Rahman, F. Daneshfar, H. Parvin, Multi-objective manifold representation for opinion mining, Expert Systems (2025). doi:10.1111/exsy.70092.

[4] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE transactions on pattern analysis and machine intelligence 35 (8) (2013) 1798–1828.

[5] F. Daneshfar, M. J. Aghajani, Enhanced text classification through an improved discrete laying chicken algorithm, Expert Systems 41 (8) (2024) e13553.

[6] Z.-W. Zhang, Z.-G. Liu, A. Martin, K. Zhou, Bsc: Belief shift clustering, IEEE Transactions on Systems, Man, and Cybernetics: Systems 53 (3) (2022) 1748–1760.

[7] H. Moayed, E. G. Mansoori, Deep and wide nonnegative matrix factorization with embedded regularization, Pattern Recognition 153 (2024) 110530.

[8] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, nature 401 (6755) (1999) 788–791.

[9] W. Barkhoda, A. Seyedi, N. Gillis, F. Akhlaghian Tab, Instance-wise distributionally robust nonnegative matrix factorization, Pattern Recognition 169 (2026) 111732.

[10] R. Mahmoodi, S. A. Seyedi, A. Abdollahpouri, F. Akhlaghian Tab, Enhancing link prediction through adversarial training in deep nonnegative matrix factorization, Engineering Applications of Artificial Intelligence 133 (2024) 108641.

[11] F. Daneshfar, S. Soleymanbaigi, P. Yamini, M. S. Amini, A survey on semi-supervised graph clustering, Engineering Applications of Artificial Intelligence 133 (2024) 108215.

[12] L. Dong, Y. Yuan, X. Luxs, Spectral–spatial joint sparse nmf for hyperspectral unmixing, IEEE Transactions on Geoscience and Remote Sensing 59 (3) (2020) 2391–2402.

[13] Y. Zhang, S. Feng, P. Wang, Z. Tan, X. Luo, Y. Ji, R. Zou, Y.-M. Cheung, Learning self-growth maps for fast and accurate imbalanced streaming data clustering, IEEE Transactions on Neural Networks and Learning Systems (2025).

[14] H. Sawada, N. Ono, H. Kameoka, D. Kitamura, H. Saruwatari, A review of blind source separation methods: two converging routes to ilrma originating from ica and nmf, APSIPA Transactions on Signal and Information Processing 8 (2019) e12.

[15] P. O. Hoyer, Non-negative matrix factorization with sparseness constraints., Journal of machine learning research 5 (9) (2004).

[16] J. Kim, H. Park, Sparse nonnegative matrix factorization for clustering, Tech. rep., Georgia Institute of Technology (2008).

[17] D. Cai, X. He, J. Han, T. S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE transactions on pattern analysis and machine intelligence 33 (8) (2010) 1548–1560.

[18] K. Zeng, J. Yu, C. Li, J. You, T. Jin, Image clustering by hyper-graph regularized non-negative matrix factorization, Neurocomputing 138 (2014) 209–217.

[19] F. Shang, L. Jiao, F. Wang, Graph dual regularization non-negative matrix factorization for co-clustering, Pattern Recognition 45 (6) (2012) 2237–2250.

[20] Z. Liu, F. Zhu, H. Xiong, X. Chen, D. Pelusi, A. V. Vasilakos, Graph regularized discriminative nonnegative matrix factorization, Engineering Applications of Artificial Intelligence 139 (2025) 109629.

[21] C. Ding, T. Li, W. Peng, H. Park, Orthogonal nonnegative matrix t-factorizations for clustering, in: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp. 126–135.

[22] J. He, D. He, B. Liu, W. Wang, Orthogonal graph regularized nonnegative matrix factorization for image clustering, in: Big Data: 7th CCF Conference, BigData 2019, Wuhan, China, September 26–28, 2019, Proceedings 7, Springer, 2019, pp. 325–337.

[23] D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, Advances in neural information processing systems 13 (2000).

[24] K. Devarajan, A statistical framework for non-negative matrix factorization based on generalized dual divergence, Neural Networks 140 (2021) 309–324.

[25] C. Ding, T. Li, W. Peng, Nonnegative matrix factorization and probabilistic latent semantic indexing: Equivalence chi-square statistic, and a hybrid method, in: AAAI, Vol. 42, 2006, pp. 137–43.

[26] M. H. Aghdam, M. D. Zanjani, A novel regularized asymmetric non-negative matrix factorization for text clustering, Information Processing & Management 58 (6) (2021) 102694.

[27] D. Kong, C. Ding, H. Huang, Robust nonnegative matrix factorization using l21-norm, in: Proceedings of the 20th ACM international conference on Information and knowledge management, 2011, pp. 673–682.

[28] S. Peng, W. Ser, Z. Lin, B. Chen, Robust sparse nonnegative matrix factorization based on maximum correntropy criterion, in: 2018 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE, 2018, pp. 1–5.

[29] C. Févotte, J. Idier, Algorithms for nonnegative matrix factorization with the β-divergence, Neural computation 23 (9) (2011) 2421–2456.

[30] M. Shi, Q. Yi, J. Lv, Symmetric nonnegative matrix factorization with beta-divergences, IEEE Signal Processing Letters 19 (8) (2012) 539–542.

[31] Y. Li, X. Zhang, M. Sun, Robust non-negative matrix factorization with β-divergence for speech separation, ETRI Journal 39 (1) (2017) 21–29.

[32] R. Yuan, C. Leng, B. Li, A. Basu, β-divergence NMF with biorthogonal regularization for data representation, Engineering Applications of Artificial Intelligence 121 (2023) 106014.

[33] B.-J. Sun, H. Shen, J. Gao, W. Ouyang, X. Cheng, A non-negative symmetric encoder-decoder approach for community detection, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 597–606.

[34] J. Wright, Y. Ma, High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications, Cambridge University Press, 2022.

[35] T. Van Erven, P. Harremos, Rényi divergence and kullback-leibler divergence, IEEE Transactions on Information Theory 60 (7) (2014) 3797–3820.

[36] C. Févotte, N. Bertin, J.-L. Durrieu, Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis, Neural computation 21 (3) (2009) 793–830.

[37] M. Mozafari, S. A. Seyedi, R. P. Mohammadiani, F. A. Tab, Unsupervised feature selection using orthogonal encoder-decoder factorization, Information Sciences (2024) 120277.

[38] T. Shi, K. Kang, J. Choo, C. K. Reddy, Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations, in: Proceedings of the 2018 world wide web conference, 2018, pp. 1105–1114.

[39] A. Salah, M. Ailem, M. Nadif, Word co-occurrence regularized non-negative matrix tri-factorization for text data co-clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

[40] N. Li, C. Leng, I. Cheng, A. Basu, L. Jiao, Dual-graph global and local concept factorization for data clustering, IEEE Transactions on Neural Networks and Learning Systems 35 (1) (2022) 803–816.