Meta-Unsupervised Learning: Application to Non-Negative Matrix Factorization

by

Ameer Ahmed Khan

A thesis submitted to the
Department of Computer Science
in conformity with the requirements for
the degree of M.Sc. Computer Science

Bishop's University
Canada
March, 2025

Abstract

Meta-learning was initially developed for supervised learning to enable models to generalize across tasks by leveraging prior experience. However, its potential in the unsupervised learning domain remains underexplored. To demonstrate the feasibility of meta-learning in unsupervised settings, we apply it to Non-negative Matrix Factorization (NMF), a widely used technique for decomposing non-negative data matrices into interpretable, lower-dimensional representations. While NMF has found applications in topic modeling, image processing, bioinformatics, and recommendation systems, it faces persistent challenges such as rank selection, optimization stability, uniqueness, and computational efficiency. Traditional approaches primarily focus on improving initialization strategies to enhance convergence and generalization. However, these methods often fail to exploit structural similarities across tasks. This paper introduces a meta-learning paradigm for NMF that systematically learns optimal factorization parameters from small-scale tasks and transfers this knowledge to improve learning on larger tasks. By discovering fine structures in small tasks and leveraging them to guide factorization on more complex datasets, our approach directs the search process toward a more optimal and structured search space, reducing the risk of suboptimal solutions and improving model robustness. This meta-unsupervised learning framework enhances NMF's ability to uncover meaningful patterns while maintaining adaptability across different domains. Additionally, we evaluate the model under noisy conditions and demonstrate its robustness by filtering noise over learning epochs, further enhancing its interpretability and stability. By integrating meta-learning principles, our method improves optimization stability and enhances interpretability and generalizability, bridging the gap between NMF-based models and advanced autonomous (unsupervised) learning strategies. The source code of this work is available at https://github.com/akhan232/meta-nmf.

Acknowledgments

I sincerely thank Almighty God for supplying me with strength, wisdom, and persistence to finish this work. His direction has been my main drive during this academic path.

I am most profoundly grateful to Dr. Rachid Hedjam for his sustained guidance and valuable feedback, which supported my research project from start to finish. Under his dedicated mentorship, I felt privileged to study and he was crucial in developing this research work.

My profound gratitude goes to my parents and family for their infinite love, encouragement, and enormous sacrifices, which have inspired me since the beginning. Their absolute trust in my ability enables me to push myself toward superior achievement. Their continuous support made this entire quest possible.

I express my gratitude to Bishop's University for providing me with an excellent academic environment and the necessary resources to conduct my research. The university's commitment to academic excellence has been instrumental in my academic growth and success.

Contents

1 Introduction **1**
- 1.1 Context and Problem Statement 1
- 1.2 Objectives 3
- 1.3 Hypothesis 4
- 1.4 Thesis Structure 4

2 Literature Review **6**
- 2.1 Background on Dimensionality Reduction Techniques 6
  - 2.1.1 Principal Component Analysis 7
  - 2.1.2 Non-Negative Matrix Factorization 8
  - 2.1.3 Comparison of NMF and PCA 14
  - 2.1.4 Regularized NMF (NMF with Regularization Terms) 14
- 2.2 Some Applications of NMF 18
- 2.3 Advantages and challenges of NMF 21
- 2.4 Supervised Meta-Learning 23
- 2.5 Meta-Learning for Unsupervised Learning 24

3 Method: Meta-Unsupervised Learning **25**
- 3.1 Problem formulation 25
- 3.2 Meta-Unsupervised Learning on Small Datasets: 26
- 3.3 Meta-Unsupervised Learning on Larger (Target) Dataset: 27

4 Experiments, Analysis and Discussion **30**
- 4.1 Dataset description 30
- 4.2 Task preparation protocol and Parameter Setting 31
- 4.3 Objective Analysis 32
  - 4.3.1 Subjective and Interpretability Analysis 41

5 Conclusion and Future Directions **46**
- References 47

List of Tables

4.1 Key characteristics of benchmark datasets. 30
4.2 Parameter Settings. 31
4.3 Clustering performance of NMF and Meta-NMF on data without noise, using mean NMI over ranks 32
4.4 Clustering performance of GNMF and Meta-GNMF on data without noise, using mean NMI over ranks 32
4.5 Clustering performance of GNMF NNDSVD and Meta-GNMF NNDSVD on data without noise, using mean NMI over ranks 32
4.6 Clustering performance of NMF NNDSVD and Meta-NMF NNDSVD on data without noise, using mean NMI over ranks 33
4.7 Clustering performance of NMF and Meta-NMF on data with added noise, using mean NMI over ranks 33
4.8 Clustering performance of GNMF and Meta-GNMF on data with added noise, using mean NMI over ranks 34
4.9 Clustering performance of NMF NNDSVD and Meta-NMF NNDSVD on data with added noise, using mean NMI over ranks 34
4.10 Clustering performance of GNMF NNDSVD and Meta-GNMF NNDSVD on data with added noise, using mean NMI over ranks 34

List of Figures

2.1 Principal Component Analysis. 8
2.2 Geometrical interpretation of PCA and NMF (credit to [77]). 14
2.3 Capability of PCA and NMF to extract local important features from faces [43] 15
2.4 Meta genes and molecular pattern discovery [7]. 18
2.5 NMF factors and loading maps show different material phases present in the natural igneous sample [20]. 19
2.6 Detailed analysis of human lymph sample. NMF deduced three regions from the nodules: 1. mantle zone, 2. core zone and 3. outer part. [56] 20
2.7 Results of NMF with four components: (a) the extracted spectral signatures (endmembers) and (b-e) the spatial distribution of their corresponding abundances [34]. 21

3.1 Flowchart illustrating a single update epoch ($k \rightarrow k+1$) for the factorized matrices in both the small datasets (2 for example) and the target dataset. The arrows pointing to $\mathbf{W}$ indicate that only $\mathbf{W}$ is explicitly computed, while the matrix $\mathbf{H}$ is implicitly computed from $\mathbf{W}^\top\mathbf{X}$. The spring arrows indicate the NMF optimization process used to learn the factorized matrices. The bullet marks indicate the positions of the learned factorized matrices within the parameter search space. 29

4.1 Task Split Protocol. 31
4.2 Comparison of NMF-based models with their Meta variants with no noise added to Blobs dataset. 35
4.3 Comparison of NMF-based models with their Meta variants with no noise added to COIL20 dataset. 36
4.4 Comparison of NMF-based models with their Meta variants with no noise added to Digits dataset. 36
4.5 Comparison of NMF-based models with their Meta variants with no noise added to Faces dataset. 37

4.6 Comparison of NMF-based models with their Meta variants with no noise added to FMNIST dataset. 37
4.7 Comparison of NMF-based models with their Meta variants with noise added to Blobs dataset. 38
4.8 Comparison of NMF-based models with their Meta variants with noise added to COIL20 dataset. 38
4.9 Comparison of NMF-based models with their Meta variants with noise added to Digits dataset. 39
4.10 Comparison of NMF-based models with their Meta variants with noise added to Faces dataset. 39
4.11 Comparison of NMF-based models with their Meta variants with noise added to FMNIST dataset. 40
4.12 Comparison of Meta-based NMF models based on real-world benchmark datasets without noise (left to right), COIL20, Digits, Faces, FMNIST. 40
4.13 Comparison of Meta-based NMF models based on real-world benchmark datasets with noise (left to right), COIL20, Digits, Faces, FMNIST. 41
4.14 Face parts extracted by NMF (top) and Meta-NMF (bottom), without added noise. 42
4.15 Face parts extracted by NMF (top) and Meta-NMF (bottom), with added noise. 42
4.16 Face parts extracted by GNMF (top) and Meta-GNMF (bottom), without added noise. 43
4.17 Face parts extracted by GNMF (top) and Meta-GNMF (bottom), with added noise. 43
4.18 Face parts extracted by NMF NNDSVD (top) and Meta-NMF NNDSVD (bottom), without added noise. 44
4.19 Face parts extracted by NMF NNDSVD (top) and Meta-NMF NNDSVD (bottom), with added noise. 44
4.20 Face parts extracted by GNMF NNDSVD (top) and Meta-GNMF NNDSVD (bottom), without added noise. 45
4.21 Face parts extracted by GNMF NNDSVD (top) and Meta-GNMF NNDSVD (bottom), with added noise. 45

Chapter 1

Introduction

1.1 Context and Problem Statement

Over the last few years, the availability of large volumes of high-dimensional data has emerged as a significant problem in data analysis and different machine-learning techniques. This led to the importance of dimensionality reduction in the broad context of various areas ranging from image processing to bioinformatics. Non-negative Matrix Factorization (NMF) has become an outstanding tool for dimensionality reduction with significant advantages stemming from inherent constraints and interpretability of results [64]. In contrast to other methods like the well-developed PCA, NMF consists of piece-by-piece decompositions that can be easily related to the features in data sets, which is why it is essential for many applications demanding the results to be explained. However, NMF has shown significant results in various fields, including text mining and image analysis [23, 39, 2, 65].

NMF stands out from other dimensionality reduction techniques due to its unique ability to generate interpretable, non-negative representations [64]. Unlike PCA, which captures variance through linear projections and often results in difficult-to-interpret negative values, NMF produces additive combinations corresponding to meaningful data parts [70]. This is particularly valuable in domains like image processing, where parts-based decompositions align with human perception. Deep learning techniques include NMF as a newer version of DR than autoencoders since it functions independently of extensive training and large datasets [29]. Storing data in a latent space by autoencoders through neural networks creates complicated yet difficult-to-interpret characteristics in such spaces. NMF delivers transparent interpretations of its results, establishing it as the ideal method when diagnostic or biological models need transparent output. Furthermore, NMF offers advantages over traditional clustering techniques, such as k-means. NMF data points can belong to multiple clusters simultaneously through its feature, allowing for overlapping cluster memberships rather than k-means assignment to single clusters [47]. The ability of NMF to assign the same data points to multiple clusters enables enhanced application success in document clustering and music analysis tasks.

The application of NMF brings efficient results to various domains of use. NMF has gained acceptance in image processing because it can identify critical visual components. The method shows remarkable success in face recognition problems through the identification of localized facial features, according to a study published by Lee and Seung [43]. NMF-based frameworks successfully boosted degraded document text extraction, according to research from Salehani et al. [65]. Similarly, Aonishi et al. [2] utilized NMF to analyse brain imaging data in neuroscience for neural pattern identification. NMF is an efficient approach to detecting topics within big text databases through text mining and topic modelling applications. Research conducted by Latif et al. [39] proves that NMF works effectively in Twitter data analytics to generate easy-to-understand subject groupings. The team of Gallego et al. [23] implemented NMF to perform consumer review classification and document clustering analysis for marketing research applications. Bioinformatics researchers have adopted NMF extensively to analyse extensive biological datasets. Gene expression analysis with NMF effectively identifies gene clusters and reveals biological processes, as Jung et al. [33]established through their research. Sweeney et al. [67], alongside other researchers, use NMF-based frameworks to identify cell types through single-cell RNA sequencing by examining expression patterns.

Recent developments have also seen NMF incorporated into recommender systems to enhance personalization. Khan et al. [36] demonstrated how NMF-based models improve recommendations by decomposing user-item interaction matrices in collaborative filtering applications. Researchers like ur Rehman et al. [63] have also explored hybrid approaches integrating NMF with meta-learning to address recommendation systems' cold-start problem and sparsity issues. These diverse applications highlight NMF's versatility in solving real-world problems across various fields, reinforcing its importance as a dimensionality reduction and representation learning tool.

Meta-learning for the dimensionality reduction in NMF is an under-researched fragment with potential effectiveness and efficacy. Meta-learning has emerged as a promising approach to improving machine learning algorithms in parallel with the developments in dimensionality reduction. Significantly, meta-learning, also known as "learning how to learn," is typically described as learning to exploit previous experience across many closely related tasks better to perform a new task [30, 12]. With exceptional success in several domains, recent meta-learning advances demonstrate the potential of applying dimensionality reduction methods [4, 21, 24]. Meta-learning and matrix factorization meet at an intersection that we can use to address these longstanding challenges and continue pushing the field of automated machine learning forward. Recent work has shown how meta-learning principles can improve optimization algorithm performance across different tasks, suggesting similar benefits to NMF applications [30, 12]. Despite this, the integration of meta-learning with NMF has not been primarily explored, especially in enhancing the clustering performance according to Normalized Mutual Information (NMI).

Standard NMF methods encounter multiple serious challenges that constrain their performance quality. These negative aspects include sensitivity problems to starting positions, non-smooth optimisation challenges, and difficulties transferring understanding between different tasks. The NMF algorithms demonstrate a strong sensitivity to their initialisation conditions, leading to unique results when run multiple times. Different starting inputs generate highly dissimilar results, thus creating challenges to obtaining dependable and repeatable models [4, 21]. For applications demanding consistent results repeatability, this undesirable sensitivity becomes problematic.

The optimisation problem for NMF frequently produces non-convex solutions because it leads to local optima instead of global ones. The algorithm cannot discover this constraint's top possible data representation. Standard optimisation methods produce inferior results, mainly in situations involving high-dimensional data and intricate underlying structures, according to Gan et al. [24]. The challenge intensifies because dataset complexity and size continue growing progressively.

The existing strategies for resolving these problems concentrate on separate NMF tasks without recognising the possibility of knowledge exchange between functions. Graph-regularized NMF (GNMF) operates as an independent factorisation approach which avoids utilising former factorisation solutions during processing [53, 17]. Due to the lack of mutual information sharing between treatment units, they struggle to generate efficient adaptations in new tasks. A fresh implementation leveraging meta-learning concepts must develop NMF's clustering ability and boost its stability and domain adaptability.

1.2 Objectives

The main objective of the thesis is to develop a meta-learning enhanced NMF framework that improves its convergence towards a more optimal solution, as well as its sensitivity to initialization and clustering performance. Specifically, the proposed methodology aims to:

- Design and implement a meta-learning model for NMF that learns and shares knowledge across multiple sub-tasks

- Apply the Meta Enhancement to different NMF variants to showcase its effectiveness

- Evaluate the framework's performance across multiple benchmark datasets

- Create a comprehensive comparative analysis framework to assess the advantages of meta-learning enhanced NMF over traditional approaches

1.3 Hypothesis

The proposed method will be supported by the following hypotheses, which we will validate by experiment.

- Integrating NMF with the Meta-learning paradigm will demonstrate statistically significant improvements in clustering performance compared to traditional NMF approaches across multiple datasets.
- The proposed meta-learning framework will significantly reduce initialization sensitivity and provide more consistent and reproducible results across different experimental conditions.
- The proposed meta-learning enhancement to NMF will make it more immune to noise.
- The proposed approach will improve the capability to concentrate on important facial features.

1.4 Thesis Structure

The thesis presents a thorough study of the enhanced Non-negative Matrix Factorization (NMF), which distributes content throughout five targeted chapters dedicated to exploring the research issue.

- **Chapter 1** establishes a solid base by presenting vital information about the high-dimensional data challenges. This segment identifies the research focus by explaining how traditional NMF methods lack effectiveness and then presents the benefits of utilizing meta-learning approaches. The first chapter establishes the research aims and main hypotheses before providing conceptual ground for the following sections.
- **Chapter 2** contains a thorough literature examination that studies standard dimensionality reduction methods and NMF applications across various fields and their current methodological difficulties. The theoretical foundation builds through this chapter by performing a critical exploration of preceding investigations, followed by the explicit definition of gaps for which the thesis intends remediation.
- **Chapter 3** outlines the proposed meta-learning enhanced NMF framework through its methodological core. The description details all aspects of the new approach by presenting distinctive methods to share knowledge and optimize processes and experimental procedures.

- **Chapter 4** demonstrates an empirical study through experimental results with comparative analysis and statistical performance evaluation of the proposed method.

- The thesis ends in **Chapter 5** through the integration of primary results and the assessment of research achievements alongside proposed investigation directions.

Chapter 2

Literature Review

This chapter comprehensively reviews the existing literature on dimensionality reduction techniques, focusing on Principal Component Analysis (PCA) and Non-Negative Matrix Factorization (NMF). It explores these methods' theoretical foundations, advantages, and limitations, particularly in the context of interpretability and performance. The chapter further discusses advancements in NMF, including regularized approaches such as Graph-Regularized NMF and Orthogonal NMF. Additionally, the role of meta-learning in both supervised and unsupervised settings is examined, emphasizing its relevance to enhancing NMF-based models. The chapter concludes with a critical review of previous research, identifying key gaps and potential directions for future studies in meta-learning and NMF.

2.1 Background on Dimensionality Reduction Techniques

Dimensionality reduction techniques are essential in machine learning and data analysis as they help simplify complex data while preserving its essential structures [31, 69]. These methods are widely used in various applications, including image processing, text mining, and bioinformatics, as they generate escalating high-dimensional data and require dimensionality reduction techniques. The category of dimensionality reduction consists of methods that fall into linear and non-linear classifications. Principal Component Analysis (PCA) and Independent Component Analysis (ICA) function as traditional methods for data dimension reduction until lower dimensions retain maximum data variance[3, 55]. The techniques display specific restrictions during their application to detect natural non-linear structures in complicated data. Unsupervised learning requires dimensionality reduction to improve data's computational efficiency and interpretability [55]. The training of unsupervised models depends on automatic pattern detection within unlabeled datasets because supervision through labelled data is not feasible. The domain implements clustering algorithms like K-Means, DBSCAN, and hierarchical clustering, while Non-Negative Matrix Factorization (NMF) is a matrix factorization technique [24]. NMF is an effective unsupervised method for cluster discovery through data decomposition because it generates interpretable parts from input data [21, 12]. Non-negativity constraints in NMF create results that better suit practical needs such as topic modeling, image processing, and genomic data analysis because NMF does not apply orthogonal transformations, which sometimes produce negative values like PCA. NMF's advantages have been accompanied by significant operational problems, which include its sensitivity to initial conditions, non-convex optimization approach, and inability to generalize between datasets. Recent work in NMF improvement has concentrated on implementing meta-learning techniques to let models leverage experience from similar tasks to enhance their performance during data analysis of previously unseen information [30, 12]. The joining of standard unsupervised learning frameworks with this integration represents a critical development that allows such systems to become more robust and adaptive.

2.1.1 Principal Component Analysis

Principal Component Analysis (PCA) is a statistical technique that transforms high-dimensional data into a lower-dimensional subspace while preserving as much variance as possible [32]. The method achieves its goal by identifying new principal component axes that produce maximum data variance. PCA implements its mathematical procedures using either the eigendecomposition of covariance matrices or Singular Value Decomposition principal [9]. The PCA methodology starts by computing data means for subtraction, followed by covariance matrix calculation. The computational process determines both eigenvectors and eigenvalues of the matrix, which indicates the principal components while showing the amount of explained variance through each element. When using principal components corresponding to the largest eigenvalues in PCA you can reduce data dimensions without losing essential data patterns [28].

Although the method successfully reduces the data dimensions, there are specific limitations. The main disadvantage of PCA is its output of negative values that create challenges during non-negative representation applications such as image processing and bioinformatics fields [43]. The linear nature of PCA transforms limit its capability to detect complex nonlinear patterns found in actual data sets. Thus the method proves insufficient for many practical applications. When using PCA it is assumed that data variance functions as the primary element for pattern distinction yet this assumption fails to hold true in various situations where class separation matters more than variance preservation [69]. The main weakness of PCA emerges from its weak performance when dealing with abnormal data points. Extreme values affect PCA calculation since it works by maximization of variance and this effect results in invalid data interpretations [26]. Interpreting PCA components becomes difficult especially when clear feature representation is necessary in particular domains. Alternative dimension reduction techniques led to the growing popularity of NMF because of its limitations. NMF resolves PCA limitations through its non-negativity constraint that produces more easily interpretable decompositions while optimizing applications for parts-based representations.

Formulation of PCA

PCA optimizes the projection of data $X \in \mathbb{R}^{n \times p}$ onto a lower-dimensional subspace by maximizing the variance of the projected data. Formally, it solves:

$$\max_{W} \operatorname{Var}(XW) \quad \text{subject to} \quad W^T W = I,$$

where $W \in \mathbb{R}^{p \times k}$ is the matrix of orthogonal projection directions (principal components), and I is the identity matrix. This is equivalent to finding the eigenvectors of the covariance matrix $C = \frac{1}{n-1} X^T X$ corresponding to the largest eigenvalues. The projected data is $Z = XW$, capturing the most significant variance in k dimensions. Fig. 2.1 provides a visual presentation of this concept,

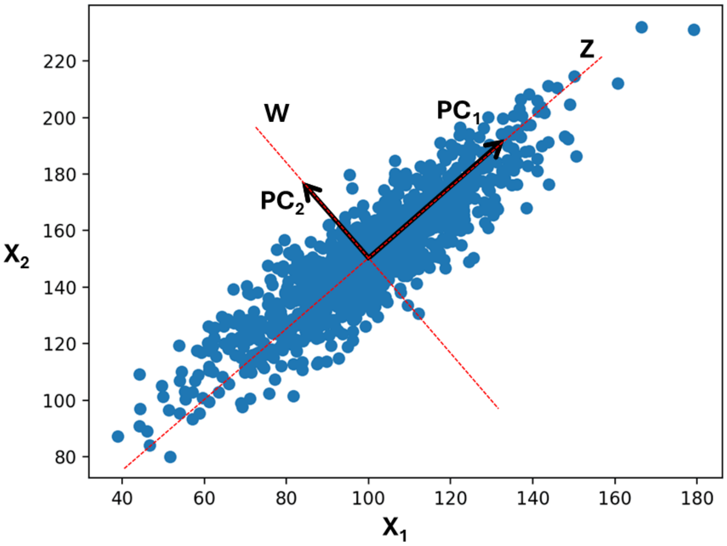


Figure 2.1: Principal Component Analysis.

2.1.2 Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) [43] operates as an effective dimensionality reduction framework which finds wide application throughout machine learning together with bioinformatics and image processing. NMF differs from conventional matrix decomposition methods PCA and SVD because it requires both matrices to contain non-negative values during decomposition [11, 64]. The characterization of the decomposition matrices as non-negative improves feature extraction because users can obtain meaningful interpretable features from the process [22]. Non-negativity constraints lead to effective performance and interpretability in the NMF algorithm. Natural datasets, including images, documents and gene expression profiles, contain values that can only be positive or zero. NMF maintains non-negativity as an enforcement rule to keep components and their coefficient values inside meaningful, realistic boundaries. The non-negative characteristic makes it suitable for datasets with uninterpretable negative values, including image pixel intensity or text word frequency data. Through its non-negative requirement, NMF generates decomposition representations that use parts rather than whole-pattern transformations, which are typical in factorization algorithms such as PCA. Moreover, NMF-based decomposition enables the interpretation of data through the subdivision of components into meaningful parts, which makes it an ideal solution for feature extraction operations [13].

Formulation of NMF

Formally, let $\mathbf{X} \in \mathbb{R}^{m \times n}$ be a matrix of n columns representing the non-negative samples and m rows representing their features, and r (lower rank) is a positive integer $< \min(m, n)$. NMF aims to find non-negative matrices $\mathbf{W} \in \mathbb{R}^{m \times r}$ and $\mathbf{H} \in \mathbb{R}^{r \times n}$ that minimize the following cost function:

$$\mathbf{W}, \mathbf{H} = \frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2, \tag{2.1}$$

where $\| \cdot \|_F^2$ represents the Frobenius norm. The model in Eq. 2.1 can also be formulated as an optimization problem of the form:

$$\min_{\mathbf{W},\mathbf{H}>0} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 = \min_{\mathbf{W},\mathbf{H}>0} \sum_{i,j} (\mathbf{X} - \mathbf{W}\mathbf{H})_{ij}^2. \tag{2.2}$$

Using multiplicative updates for the non-negative optimization system proposed by [42], $\mathbf{H}$ and $\mathbf{W}$ are updated by:

$$\mathbf{H}^{(t+1)} \leftarrow \mathbf{H}^{(t)} \odot \frac{\mathbf{W}^{(t)\top}\mathbf{X}}{\mathbf{W}^{(t)\top}\mathbf{W}^{(t)}\mathbf{H}^{(t)}}, \tag{2.3}$$

$$\mathbf{W}^{(t+1)} \leftarrow \mathbf{W}^{(t)} \odot \frac{\mathbf{X}\mathbf{H}^{(t+1)\top}}{\mathbf{W}^{(t)}\mathbf{H}^{(t+1)}\mathbf{H}^{(t+1)\top}}, \tag{2.4}$$

where $\odot$ stands for the element-wise matrix product, and $\frac{A}{B}$ stands for the element-wise matrix division. $\mathbf{H}^{(0)}$ and $\mathbf{W}^{(0)}$ are set to random values, and the updates are repeated until $\mathbf{W}$ and $\mathbf{H}$ become stable.

Optimization process for Updating W and H

In the context of Non-Negative Matrix Factorization (NMF), we have $\Theta = (\mathbf{W}, \mathbf{H})$. Instead of simultaneously updating both matrices, the algorithm can be simplified by following **block-coordinate descent** scheme in which algorithm takes turns in updating both matrices, where one matrix is updated in one iteration keeping the other matrix constant and updating the second matrix in next iteration keeping the first one constant. The optimization process presented here is taken from [8]:

$$\mathbf{H} \leftarrow \mathbf{H} - \eta_{\mathbf{H}} \circ \nabla_{\mathbf{H}} D(\mathbf{X}, \mathbf{WH}) \tag{2.5}$$

$$\mathbf{W} \leftarrow \mathbf{W} - \eta_{\mathbf{W}} \circ \nabla_{\mathbf{W}} D(\mathbf{X}, \mathbf{WH}) \tag{2.6}$$

The standard Euclidean distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ of N elements can be expressed as

$$d_{\text{EUC}}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \tag{2.7}$$

Alternatively, we can characterize the Euclidean norm as:

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_N^2} = \sqrt{\sum_{i=1}^{N} x_i^2} \tag{2.8}$$

which allows us to reinterpret eq. 2.8 as the norm of the difference vector:

$$d_{\text{EUC}}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| \tag{2.9}$$

For matrices, the equivalent is the Frobenius norm, defined by

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} x_{ij}^2} \tag{2.10}$$

Using the Frobenius norm of their difference, we can express the Euclidean-like distance of matrices as:

$$D_{\text{EUC}}(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|_F = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} (x_{ij} - y_{ij})^2} \tag{2.11}$$

For NMF, we can make gradient computations simple by using the square of the Frobenius norm. By doing that we get following cost function:

$$D_{\text{EUC}}(\mathbf{X}, \mathbf{WH}) = \|\mathbf{X} - \mathbf{WH}\|_F^2 = \sum_m \sum_n (x_{mn} - \mathbf{WH}|_{mn})^2 \tag{2.12}$$

where $\mathbf{WH}|_{mn}$ represents the mn-th element of $\mathbf{WH}$. we have two options for calculating the gradient of this equation: one is the compact matrix notation which leverages the matrix operations to express the gradients, while the other option is element-by-element derivatives which includes the calculation of derivative of equation with respect to every element. Both of these will provide us with the same result; however, we will be discussing it using matrix calculus to keep it simple and elegant.

To begin, let's focus on a key property of the matrix trace. As a reminder, the trace is simply the sum of the elements along the main diagonal of a square matrix. The specific property we'll be utilizing is:

$$\text{tr}(\mathbf{X}^\top \mathbf{Y}) = \sum_{i=1}^{M} \sum_{j=1}^{N} x_{ij} y_{ij} \tag{2.13}$$

Importantly, the trace of a matrix product equals the sum of their element-wise (Hadamard) products, provided that both matrices have the same size. Consequently, the Frobenius norm (eq. 2.10) can be rewritten as

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} x_{ij}^2} = \sqrt{\text{tr}(\mathbf{X}^\top \mathbf{X})}, \tag{2.14}$$

The Euclidean NMF cost function can then be expressed as:

$$D_{\text{EUC}}(\mathbf{X}, \mathbf{WH}) = \|\mathbf{X} - \mathbf{WH}\|_F^2 = \text{tr}[(\mathbf{X} - \mathbf{WH})^\top (\mathbf{X} - \mathbf{WH})]. \tag{2.15}$$

Using the addition and multiplication property of the transpose, we get

$$D_{\text{EUC}}(\mathbf{X}, \mathbf{WH}) = \text{tr}[(\mathbf{X}^\top - \mathbf{H}^\top \mathbf{W}^\top)(\mathbf{X} - \mathbf{WH})] = \text{tr}(\mathbf{X}^\top \mathbf{X} - \mathbf{X}^\top \mathbf{WH} - \mathbf{H}^\top \mathbf{W}^\top \mathbf{X} + \mathbf{H}^\top \mathbf{W}^\top \mathbf{WH}) \tag{2.16}$$

We leverage several matrix algebraic properties, including:

- Addition property: $\text{tr}(\mathbf{A} + \mathbf{B}) = \text{tr}(\mathbf{A}) + \text{tr}(\mathbf{B})$
- Cyclic permutation: $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{CAB}) = \text{tr}(\mathbf{BCA})$

Key gradient computation properties include:

$$\begin{aligned}\nabla_{\mathbf{X}}\mathrm{tr}(\mathbf{A}\mathbf{X}) &= \mathbf{A}^\top \\ \nabla_{\mathbf{X}}\mathrm{tr}(\mathbf{X}^\top\mathbf{A}) &= \mathbf{A} \\ \nabla_{\mathbf{X}}\mathrm{tr}(\mathbf{X}^\top\mathbf{A}\mathbf{X}) &= (\mathbf{A}+\mathbf{A}^\top)\mathbf{X} \\ \nabla_{\mathbf{X}}\mathrm{tr}(\mathbf{X}\mathbf{A}\mathbf{X}^\top) &= \mathbf{X}(\mathbf{A}+\mathbf{A}^\top)\end{aligned} \tag{2.17}$$

These properties form the foundation for systematically computing gradients in matrix factorization problems. For $\nabla_{\mathbf{H}} D_{\mathrm{EUC}}(\mathbf{X}, \mathbf{WH})$, we proceed term by term:

First term: $\nabla_{\mathbf{H}}\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) = 0$, since it does not depend on $\mathbf{H}$

Second term:

$$\nabla_{\mathbf{H}}\mathrm{tr}(\mathbf{X}^\top\mathbf{W}\mathbf{H}) = (\mathbf{X}^\top\mathbf{W})^\top = \mathbf{W}^\top\mathbf{X} \tag{2.18}$$

Third term:

$$\nabla_{\mathbf{H}}\mathrm{tr}(\mathbf{H}^\top\mathbf{W}^\top\mathbf{X}) = \mathbf{W}^\top\mathbf{X} \tag{2.19}$$

Last term:

$$\nabla_{\mathbf{H}}\mathrm{tr}(\mathbf{H}^\top\mathbf{W}^\top\mathbf{W}\mathbf{H}) = [\mathbf{W}^\top\mathbf{W} + (\mathbf{W}^\top\mathbf{W})^\top] = 2\mathbf{W}^\top\mathbf{W}\mathbf{H} \tag{2.20}$$

Combining these terms with the appropriate signs from Eq. 2.16, we obtain:

$$\nabla_{\mathbf{H}} D_{\mathrm{EUC}}(\mathbf{X}, \mathbf{WH}) = -2\mathbf{W}^\top\mathbf{X} + 2\mathbf{W}^\top\mathbf{W}\mathbf{H} \tag{2.21}$$

For $\nabla_{\mathbf{W}} D_{\mathrm{EUC}}(\mathbf{X}, \mathbf{WH})$, we follow a similar approach:

First term: $\nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{X}^\top\mathbf{X}) = 0$

Second term:

$$\nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{X}^\top\mathbf{W}\mathbf{H}) = \nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{H}\mathbf{X}^\top\mathbf{W}) = (\mathbf{H}\mathbf{X}^\top)^\top = \mathbf{X}\mathbf{H}^\top \tag{2.22}$$

Third term:

$$\nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{H}^\top\mathbf{W}^\top\mathbf{X}) = \nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{X}\mathbf{H}^\top\mathbf{W}^\top) = \mathbf{X}\mathbf{H}^\top \tag{2.23}$$

Last term:

$$\nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{H}^\top\mathbf{W}^\top\mathbf{W}\mathbf{H}) = \nabla_{\mathbf{W}}\mathrm{tr}(\mathbf{W}\mathbf{H}\mathbf{H}^\top\mathbf{W}^\top) = \mathbf{W}[(\mathbf{H}\mathbf{H}^\top)^\top + \mathbf{H}\mathbf{H}^\top] = 2\mathbf{W}\mathbf{H}\mathbf{H}^\top \tag{2.24}$$

Finally, we get:

$$\nabla_{\mathbf{W}} D_{\mathrm{EUC}}(\mathbf{X}, \mathbf{WH}) = -2\mathbf{X}\mathbf{H}^\top + 2\mathbf{W}\mathbf{H}\mathbf{H}^\top \tag{2.25}$$

Now we have $\nabla_{\mathbf{H}} D_{\text{EUC}}(\mathbf{X}, \mathbf{WH})$ (eq. 2.21) and $\nabla_{\mathbf{W}} D_{\text{EUC}}(\mathbf{X}, \mathbf{WH})$ (eq. 2.25); by substituting these results back into Equations 2.5 and 2.6, we derive the block-coordinate gradient descent algorithm for Euclidean NMF.

$$\mathbf{H} \leftarrow \mathbf{H} + \eta_{\mathbf{H}} \circ (\mathbf{W}^\top \mathbf{X} - \mathbf{W}^\top \mathbf{WH}) \tag{2.26}$$

$$\mathbf{W} \leftarrow \mathbf{W} + \eta_{\mathbf{W}} \circ (\mathbf{XH}^\top - \mathbf{WHH}^\top) \tag{2.27}$$

We can disregard the scalar 2, as it can be incorporated into the learning rates.

In traditional gradient descent, the learning rates ($\eta_{\mathbf{H}}$ and $\eta_{\mathbf{W}}$) are positive, which can lead to negative elements in the update rules due to subtraction. This violates the non-negativity constraint crucial for NMF. To address this, Lee and Seung [42] introduced a clever technique in 2001: using data-adaptive learning rates that eliminate subtraction from the update process, thus preventing negative elements. This is achieved by strategically defining the learning rates. For instance, for H, we can set:

$$\eta_{\mathbf{H}} = \frac{\mathbf{H}}{\mathbf{W}^\top \mathbf{WH}} \tag{2.28}$$

(where the fraction line represents element-wise division), the first update rule transforms into:

$$\mathbf{H} \leftarrow \mathbf{H} + \frac{\mathbf{H}}{\mathbf{W}^\top \mathbf{WH}} \circ (\mathbf{W}^\top \mathbf{X} - \mathbf{W}^\top \mathbf{WH}) = \mathbf{H} + \mathbf{H} \circ \frac{\mathbf{W}^\top \mathbf{X}}{\mathbf{W}^\top \mathbf{WH}} - \mathbf{H} \circ \frac{\mathbf{W}^\top \mathbf{WH}}{\mathbf{W}^\top \mathbf{WH}} = \mathbf{H} \circ \frac{\mathbf{W}^\top \mathbf{X}}{\mathbf{W}^\top \mathbf{WH}} \tag{2.29}$$

Similarly for **W** we can set:

$$\eta_{\mathbf{W}} = \frac{\mathbf{W}}{\mathbf{WHH}^\top} \tag{2.30}$$

update rule of **W** becomes:

$$\mathbf{W} \leftarrow \mathbf{W} + \frac{\mathbf{W}}{\mathbf{WHH}^\top} \circ (\mathbf{XH}^\top - \mathbf{WHH}^\top) = \mathbf{W} + \mathbf{W} \circ \frac{\mathbf{XH}^\top}{\mathbf{WHH}^\top} - \mathbf{W} \circ \frac{\mathbf{WHH}^\top}{\mathbf{WHH}^\top} = \mathbf{W} \circ \frac{\mathbf{XH}^\top}{\mathbf{WHH}^\top} \tag{2.31}$$

Through this transformation, the additive update rules become multiplicative update rules. This eliminates the possibility of generating negative elements, as all values involved are positive and the updates only involve multiplications and divisions.

$$\mathbf{H} \leftarrow \mathbf{H} \circ \frac{\mathbf{W}^\top \mathbf{X}}{\mathbf{W}^\top \mathbf{WH}} \tag{2.32}$$

$$\mathbf{W} \leftarrow \mathbf{W} \circ \frac{\mathbf{X}\mathbf{H}^\top}{\mathbf{W}\mathbf{H}\mathbf{H}^\top} \tag{2.33}$$

2.1.3 Comparison of NMF and PCA

NMF excels at capturing inherent data structure, a potential weakness of PCA. While PCA ensures feature orthogonality—a mathematically guaranteed property due to its reliance on eigenvectors of a symmetric covariance matrix—this orthogonality can sometimes obscure the data's true underlying relationships. Unlike PCA, NMF prioritizes data representation over strict orthogonality, leading to potentially more meaningful feature decompositions. Figure 2.2 illustrates this difference.

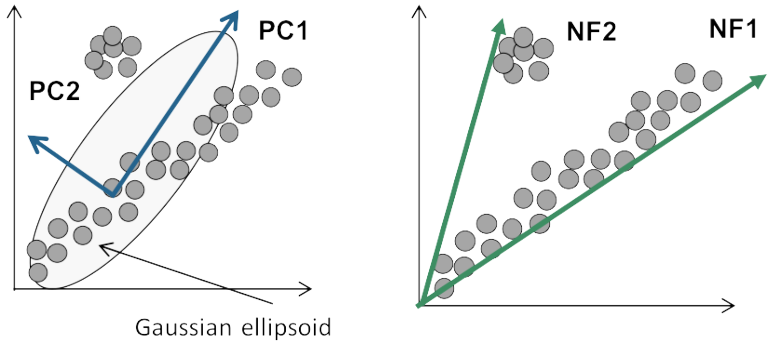


Figure 2.2: Geometrical interpretation of PCA and NMF (credit to [77]).

Both NMF and PCA reduce complex data, but they do so differently. NMF decomposes data, like gene expression or images, into interpretable, parts-based features (e.g., facial features like "eyes" or "nose"). These can be combined to reconstruct the original. PCA, on the other hand, provides a series of progressively refined, holistic approximations of the data, like increasingly accurate "generic" faces as shown in Fig. 2.3. The choice between them depends on whether you need interpretable parts or a hierarchical representation of the whole.

2.1.4 Regularized NMF (NMF with Regularization Terms)

Non-Negative Matrix Factorization (NMF) has emerged as a fundamental tool in machine learning, data analysis, and pattern recognition due to its ability to extract interpretable and meaningful components from high-dimensional data [58]. However, despite its advantages in feature extraction and dimensionality reduction, standard NMF suffers from inherent limitations, such as sensitivity to initialization, susceptibility to noise, and a lack of constraints to guide the factorization process [16]. To address these challenges, researchers have introduced regularized NMF, which incorporates additional constraints into the factorization objective [46,

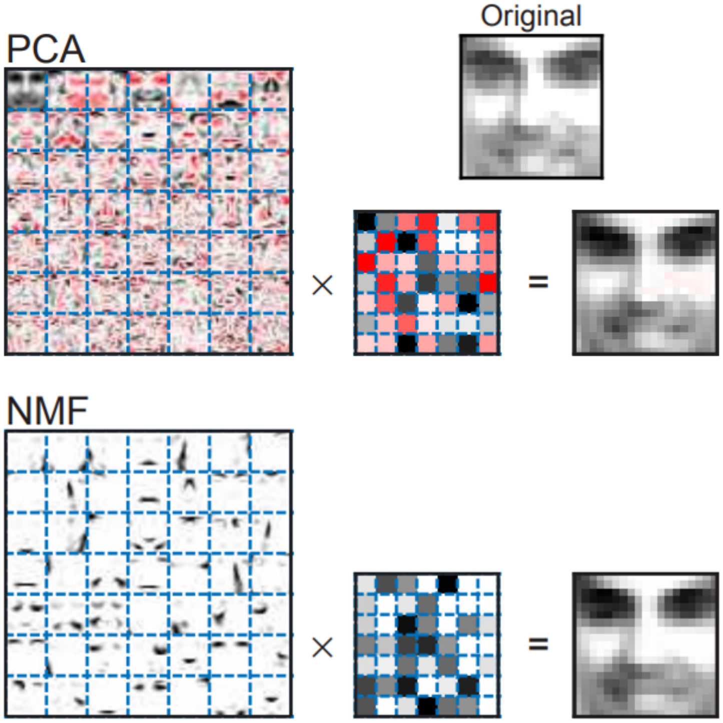


Figure 2.3: Capability of PCA and NMF to extract local important features from faces [43]

61, 49]. These regularization terms help improve generalization, stability, and interpretability by introducing prior knowledge or desirable properties into the factorization process.

Regularization in NMF is essential to mitigate some of the common drawbacks associated with traditional NMF approaches. One primary issue is the non-convex nature of the NMF optimization problem, which often leads to convergence to suboptimal solutions. Since NMF relies on iterative updates, different initializations can result in significantly different factorizations, making it difficult to obtain consistent results. Regularization constraints help steer the optimization process toward meaningful solutions by incorporating structural information or penalizing undesired properties. Additionally, regularized NMF can enhance sparsity, smoothness, orthogonality, or local structure preservation, depending on the application domain [43].

Graph-Regularized NMF (G-NMF)

One of the most widely studied forms of regularized NMF is graph-regularized NMF (G-NMF), which seeks to preserve the local geometric structure of data during factorization. In many real-world datasets, such as images, text corpora, and biological data, the underlying data distribution follows a manifold structure. Traditional NMF does not account for this geometric relationship, often leading to loss of critical structural information. Graph-regularized NMF introduces a graph Laplacian term into the NMF objective function, which enforces that similar data points in the high-dimensional space remain close in the lower-dimensional representation [11, 4, 73]. The mathematical formulation of G-NMF is given by:

$$\min_{\mathbf{W},\mathbf{H}\geq 0} ||\mathbf{X} - \mathbf{W}\mathbf{H}||_F^2 + \lambda \operatorname{Tr}(\mathbf{H}\mathbf{L}\mathbf{H}^\top) \tag{2.34}$$

where $\mathbf{L}$ is the graph Laplacian matrix and λ controls the influence of the graph regularization term.

Despite its advantages, G-NMF faces some challenges. Constructing an optimal similarity graph is non-trivial and requires careful selection of parameters such as the number of neighbors and weight functions. Additionally, the computational complexity of computing the graph Laplacian increases with dataset size, making large-scale applications difficult. Researchers have attempted to address these issues by integrating adaptive graph learning strategies and optimizing the efficiency of graph construction algorithms [74].

Orthogonal NMF (O-NMF)

Another critical variant of regularized NMF is orthogonal NMF (O-NMF), which introduces orthogonality constraints on either the basis matrix W or the coefficient matrix H [15, 48]. The motivation behind this constraint is to ensure that different components in the factorized matrices capture distinct and non-overlapping features. Unlike traditional NMF, where basis vectors can have redundant or correlated structures, O-NMF enforces decorrelation, making it particularly useful for clustering and classification tasks [60, 47].

The mathematical formulation of O-NMF is given by:

$$\min_{\mathbf{W},\mathbf{H}\geq 0} ||\mathbf{X} - \mathbf{W}\mathbf{H}||_F^2, \quad \text{s.t.} \quad \mathbf{W}^\top\mathbf{W} = \mathbf{I} \quad \text{or} \quad \mathbf{H}^\top\mathbf{H} = \mathbf{I} \tag{2.35}$$

where I is the identity matrix, enforcing orthogonality on W or H.

While O-NMF improves feature separation and enhances interpretability, it presents significant computational challenges. The non-convexity of the orthogonality constraint makes optimization difficult, often leading to slow convergence and increased sensitivity to noise. Moreover, enforcing strict orthogonality can sometimes reduce flexibility in capturing the underlying data distribution, particularly when dealing with highly overlapping clusters [35].

Sparse NMF

Another important category of regularized NMF is sparse NMF, which introduces sparsity-inducing constraints on the factorized matrices. Sparsity is a desirable property in many applications, as it enhances interpretability by ensuring that only a small subset of features contributes to each factorized component [59, 41, 61]. In text mining, for example, sparse NMF helps extract distinct topics where each topic is represented by a limited number of words, leading to better topic coherence. Similarly, in bioinformatics, sparsity enables the identification of key genes associated with specific biological processes while filtering out irrelevant noise [38].

Sparse NMF is typically implemented by incorporating an L_1 norm or an L_0 norm constraint into the NMF objective function:

$$\min_{\mathbf{W},\mathbf{H}\geq 0} ||\mathbf{X} - \mathbf{WH}||_F^2 + \lambda(|\mathbf{W}|_1 + |\mathbf{H}|_1) \tag{2.36}$$

where λ controls the level of sparsity.

Although L_1 -based regularization is computationally efficient, it may not always produce truly sparse representations, as it only encourages sparsity rather than strictly enforcing it [25].

Smooth NMF

Another direction in regularized NMF research involves smooth NMF, which incorporates smoothness constraints to regularize the factorized matrices [45, 54]. In many applications, abrupt changes in factorized components are undesirable, as they may introduce artifacts or instability. Smooth NMF addresses this by enforcing gradual variations in the matrix elements, making it particularly useful in signal processing, image analysis, and time-series modeling. Smoothness constraints are often implemented using L_2 norm penalties or total variation minimization techniques, which suppress rapid fluctuations and enhance continuity in learned representations [66].

The mathematical formulation of smooth NMF is given by:

$$\min_{\mathbf{W},\mathbf{H}\geq 0} ||\mathbf{X} - \mathbf{WH}||_F^2 + \lambda \sum_{i,j} (\mathbf{H}_{i,j} - \mathbf{H}_{i,j+1})^2, \tag{2.37}$$

where λ controls the degree of smoothness applied to $\mathbf{H}$. While smooth NMF provides improved stability and noise resistance, it may reduce the ability to capture fine-grained details in certain applications. Furthermore, defining an appropriate smoothness level requires domain expertise, as excessive smoothing can blur important structures in the data. Recent advances have explored adaptive smoothness constraints, where the degree of smoothness is learned dynamically based on dataset characteristics, thereby offering a more flexible approach [76].

2.2 Some Applications of NMF

NMF functions as a strong analytical approach for reducing dimensions and cluster analysis and extracting features across numerous fields of study. Because of its power to create understandable positive expressions, NMF finds broad application in bioinformatics, image processing, text mining, material science, recommendation systems, and spectroscopy.

Bioinformatics research extensively uses NMF techniques to analyse large biological datasets. The method finds its main application in disease-related gene expression analysis by revealing groups of genes that co-activate together. BioNMF represents a tool [57] created to demonstrate how NMF effectively extracts important patterns from genetic expression information. According to [18], NMF functions as a tool for cancer subtype detection through matrix decomposition, generating biologically significant components. [1] applied NMF within a multi-label classification platform to discover human disease-related gene modules, thus expanding its value in precision medicine. NMF has proven its usefulness in reducing thousands of genes data to a few meta genes and the ability to deduce meaningful information from microarray data related to cancer [7], which is also presented in Fig. 2.4

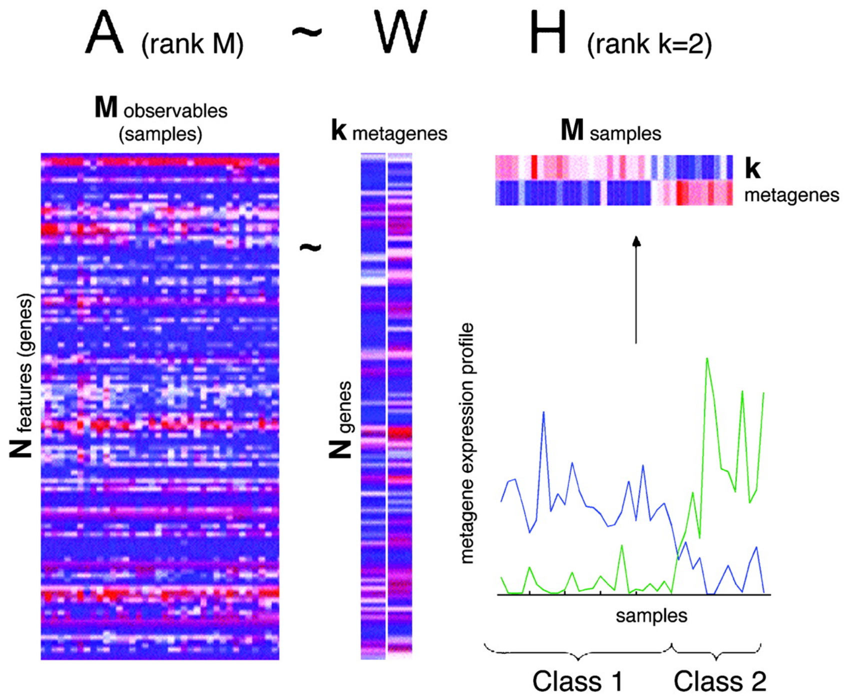
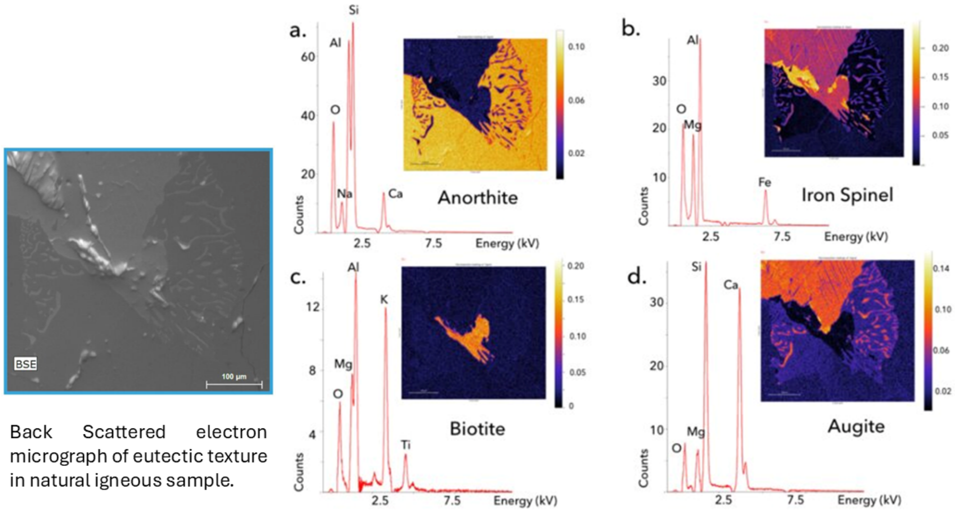


Figure 2.4: Meta genes and molecular pattern discovery [7].

NMF serves image processing applications, performing tasks including face recognition, object detection, and image denoising. The additive decomposition of images through NMF identifies important features like facial elements, including eyes, mouth, and nose. NMF produces localised sparse representations that are an attractive substitute for the standard Principal Component Analysis (PCA). The research team of [20] showed NMF, together with other machine learning techniques, could extract microscopic image structural patterns through their materials science heterogeneity analysis (Fig. 2.5).



Back Scattered electron micrograph of eutectic texture in natural igneous sample.

Figure 2.5: NMF factors and loading maps show different material phases present in the natural igneous sample [20].

Mass spectrometry and chemical imaging heavily depend on NMF for their operations. [56] published a paper comparing different NMF-related methods for practical applications in mass spectrometry imaging data analysis. Fig. 2.6 shows the deduction of three regions from human lymph sample, by NMF. [19] developed weighted NMF to improve high-resolution mass spectrometry analysis, and [40] investigated its usage in MR spectroscopy studies. Researchers demonstrate with their work that NMF enhances the identification of compounds present in heterogeneous mixtures and develops better methods to study materials.

Text mining represents a key domain for NMF applications, which uses the method primarily for topic modeling tasks. NMF constructs interpretive topic-word distributions through its non-negative factor matrix, while LDA fails to generate such results. The technique proves helpful across three significant applications, including document grouping, analysis of sentiment and retrieval tasks. Individuals evaluating short-text topic mining schemes tested both NMF and LDA-based approaches through experiments, according to [14] and proposed Knowledge guided

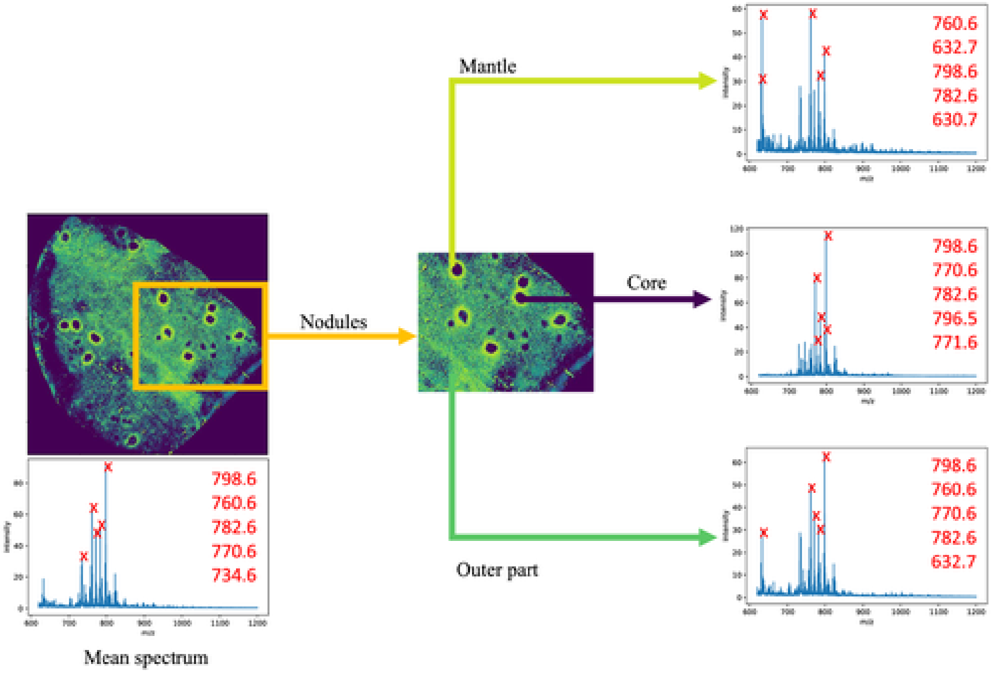


Figure 2.6: Detailed analysis of human lymph sample. NMF deduced three regions from the nodules: 1. mantle zone, 2. core zone and 3. outer part. [56]

NMF for short text topic mining. NMF achieved high effectiveness in topic extraction from textual data according to [52], which employed both NMF and the existing model.

The non-negative matrix factorisation technique uses spectroscopic and microscopical data analysis to identify complex material spectra. [34] developed physically constrained linear unmixing, which introduced NMF as part of a universal analysis method for heterogeneous materials. Fig. 2.7 shows the results of Non-negative Matrix Factorization (NMF) applied to ToF-SIMS data, with the optimal decomposition found using four components. Research findings show that NMF improves spectral data understanding, which produces a better understanding of material properties.

NMF is an analytical tool in neuroscience and behavioral research to process locomotion pattern data. [27] utilised NMF as one of their dimensionality reduction techniques to extract shape-based features in *C. elegans* locomotion studies. The application is helpful for neurological disorder research by detecting different behavioral patterns. The framework of NMF delivers successful performance in recommendation systems aimed at collaborative filtering. NMF generates customised recommendation outputs through its ability to extract user-specific preferences by breaking down user-item interaction data. Collaborative filtering systems received

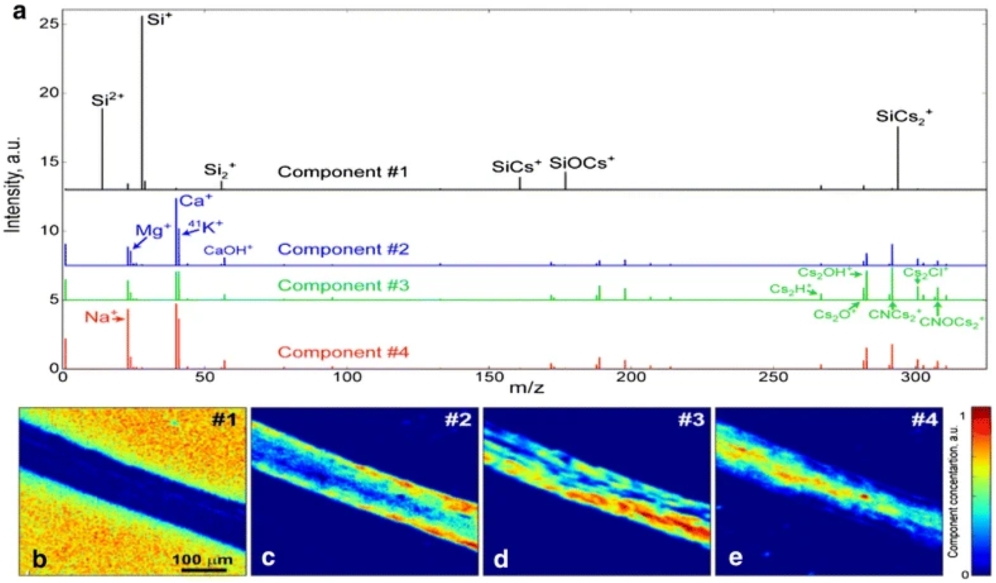


Figure 2.7: Results of NMF with four components: (a) the extracted spectral signatures (endmembers) and (b-e) the spatial distribution of their corresponding abundances [34].

improved recommendations through an efficient NMF-based approach developed by [51] and Bayesian NMF clustering techniques examined by [5]. [44] presented a detailed study of matrix factorisation methods, specifically discussing NMF's contribution to recommendation precision. NMF is vital for enhancing recommendation accuracy since it generates meaningful connections between users and items in consumer platforms.

NMF has wide applications across domains because it maintains its core position as an essential method for pattern analysis, clustering, and dimensionality reduction. It remains a fundamental tool for analysing high-dimensional data while providing robust and interpretable solutions in bioinformatics, image processing, text mining, material science, recommendation systems, and spectroscopy. Integrating NMF capabilities with meta-learning and deep learning methods continues to develop new possibilities for diverse field applications.

2.3 Advantages and challenges of NMF

In recent years, NMF has become increasingly popular due to its valuable properties and applications [43, 64]. Below, we summarize its key advantages and challenges.

Advantages

1. **Interpretable part-based representations:** NMF produces more naturally interpretable features than traditional methods like PCA, which involve holistic data transformations [70].

2. **Topic modeling applications:** NMF excels in text mining and information retrieval, where columns correspond to word sets defining topics and rows represent documents [39, 23].

3. **Image decomposition:** Successfully used in facial recognition to separate images into components like textures, edges, and facial features, improving classification and clustering tasks [2, 65].

4. **Bioinformatics utility:** Valuable for analyzing gene expression data to identify patterns representing biological processes or disease states, aiding medical diagnostics and research [33].

5. **Sparse data compatibility:** The non-negative constraint makes NMF well-suited for preserving sparsity in real-world datasets like document term matrices, increasing computational efficiency and avoiding overfitting [22].

6. **Model transparency:** NMF's interpretable results provide a basis for applications requiring transparent and explainable machine learning models [67].

Challenges

1. **Initialization sensitivity:** NMF's iterative optimization approach makes it highly sensitive to initialization, generating different solutions based on starting matrices and preventing consistent results across runs [21, 53].

2. **Non-convex optimization:** The optimization function often converges to local minima rather than absolute minima, especially problematic in high-dimensional datasets [24].

3. **Limited scalability:** Computational complexity increases dramatically with dataset size, making NMF impractical for large-scale real-time applications [67].

4. **Transfer learning limitations:** NMF lacks built-in mechanisms to transfer knowledge between tasks, limiting its generalizability and requiring advanced techniques like meta-learning for improvement [63, 30].

5. **Optimization challenges:** Despite research into regularization techniques and constraint-based approaches, a universally optimal strategy for overcoming local minima has not been found [17].

6. **Computational trade-offs:** Optimized algorithms exploring parallel computing and stochastic updates often compromise between accuracy and efficiency [16].

Meta-learning approaches offer promising solutions to these challenges by improving initialization strategies, enabling adaptive optimization, and facilitating cross-task knowledge transfer [31, 72, 50, 63]. However, integrating meta-learning with NMF introduces additional computational complexity and requires further research to develop efficient algorithms that can effectively capture task similarities across datasets.

2.4 Supervised Meta-Learning

We are inspired by the MAML model, a framework designed to enable rapid adaptation of deep learning models to new tasks with minimal data. The key idea behind MAML is to learn a set of model parameters that are well-suited for fine-tuning a variety of tasks. Rather than training separate models for each task, MAML seeks to optimize a shared set of parameters that can be quickly adapted with a small number of gradient updates. This approach is model-agnostic, meaning it can be applied to any model that uses gradient-based learning, including neural networks. MAML has been demonstrated to outperform traditional learning methods in tasks such as few-shot classification and reinforcement learning, showing the potential of meta-learning for improving the efficiency of model adaptation to new tasks. Therefore, given a task distribution $p(T)$, MAML aims to find a model initialization θ^* such that after a few gradient steps on a task, the model performs well on that task. The objective is to optimize the model parameters θ to minimize the loss on a set of tasks with minimal updates. The Meta-objective function to optimize is as follows:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{T\sim p(T)} \left[\mathcal{L}_{\text{val}} \left(\theta - \alpha\nabla_\theta \mathcal{L}_{\text{train}}(\theta)\right)\right], \tag{2.38}$$

where T is a task sampled from the task distribution $p(T)$, $\mathcal{L}_{\text{train}}(\theta)$ is the training loss on the task, $\mathcal{L}_{\text{val}}(\theta)$ is the validation loss on the task after updating the model parameters θ and α is the learning rate used for the task-specific updates, and $\mathbb{E}_{T\sim p(T)}$ represents the expectation over tasks T that are sampled from a task distribution $p(T)$. This means that the optimization objective is averaged over multiple tasks rather than a single task. The task-specific model parameters are updated as follows:

$$\theta' = \theta - \alpha\nabla_\theta \mathcal{L}_{\text{train}}(\theta), \tag{2.39}$$

where θ' are the parameters after one step of gradient descent on the task's training data.

2.5 Meta-Learning for Unsupervised Learning

While meta-learning has been extensively studied in supervised settings, its application in other machine learning paradigms remains a developing area [68]. Unlike supervised approaches, where labeled data guide model training, unsupervised meta-learning must infer patterns and structures directly from unlabeled data. One of the most promising applications is in clustering and dimensionality reduction, particularly in the context of Non-Negative Matrix Factorization (NMF).

Meta-learning for unsupervised learning aims to address the limitations of conventional unsupervised algorithms by enabling models to generalize factorization patterns across tasks. One of the main challenges in unsupervised meta-learning is the absence of explicit feedback signals, making it difficult to optimize model performance. To address this, researchers have explored self-supervised learning approaches, where proxy tasks such as contrastive learning or pseudo-labeling guide the meta-learning process [37]. Additionally, adversarial learning has been used to enhance unsupervised meta-learning, improving model robustness and generalization [71].

Recent studies have demonstrated the effectiveness of meta-learning-enhanced NMF in clustering applications [30]. By leveraging meta-learning principles, models can optimize initialization, dynamically adjust regularization parameters, and enhance generalization across diverse data distributions. Furthermore, meta-learning has been applied in anomaly detection, where models learn to detect outliers in new datasets based on prior knowledge from related tasks [62, 75].

Future directions in unsupervised meta-learning include the development of more scalable algorithms capable of handling large-scale datasets and the integration of meta-learning with generative models, such as Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs), to improve representation learning. Additionally, researchers aim to enhance interpretability in unsupervised meta-learning models, making them more transparent and explainable in critical applications such as healthcare and finance.

Chapter 3

Method: Meta-Unsupervised Learning

3.1 Problem formulation

In the proposed framework, we define three distinct types of tasks, each associated with datasets of increasing size to facilitate progressive knowledge transfer and generalisation. First, *Meta-Tasks* operate on small-scale meta-datasets, which serve as auxiliary learning problems designed to capture local structures and extract transferable representations. These meta-datasets are intentionally small to ensure computational efficiency and prevent overfitting to any single sub-problem. Second, the *Learning Target Task* is conducted on a *target dataset*, which is typically larger than the meta-datasets, allowing the model to refine the learned representations and adapt to a broader data distribution. The learning target dataset acts as an intermediate stage, balancing specificity and generalizability by leveraging knowledge acquired from the *meta-tasks*. Finally, the *Final Generalization Task* is performed on the ultimate evaluation dataset, which is the largest and entirely unseen during the learning phase. The increased size of this dataset ensures a robust evaluation of the model's generalisation ability, as it encompasses a more comprehensive and diverse data distribution. The hierarchical structuring of these tasks, where the meta-datasets are smaller than the learning target dataset and the ultimate dataset is the largest, ensures a gradual, scalable learning process, preventing overfitting at early stages while maximising the model's adaptability and generalisation performance.

We first define the relevant notations and concepts to lay the foundation for our approach. Let $\mathcal{T}_s = \{\mathcal{T}_{s_i}; \forall i = 1..N\}$ a set of N *meta-tasks*. Each *meta-task* represents an individual task within the larger problem. Each *meta-task* corresponds to a dataset $\mathbf{X}_{s_i} \in \mathbb{R}^{m \times n_s}$, where m represents the number of features, and n_s refers to the number of samples in the *meta-task*. The *learning target task*, denoted as $\mathcal{T}_t$, is the primary task that the model aims to improve upon using the knowledge learned from multiple *meta-tasks*. The dataset for the *learning target task* is represented as $\mathbf{X}_t \in \mathbb{R}^{m \times n_t}$, where n_t is the number of samples in the *learning target task*. The *final generalisation*, denoted as $\mathcal{T}_e$, is the ultimate task on which the model is evaluated. This evaluation leverages the knowledge acquired from previous tasks (meta-tasks and learning target task) during the meta-learning phase. The dataset for $\mathcal{T}_e$ is represented as $\mathbf{X}_e \in \mathbb{R}^{m \times n_e}$, where n_e denotes the number of samples in the dataset. We assume that $\mathcal{T}_s$, $\mathcal{T}_t$ and $\mathcal{T}_e$ are i.i.d. This implies that, theoretically, each task is statistically independent of the others and drawn from the same underlying probability distribution. In meta-learning, this assumption is critical because it ensures that the model's performance on T_e generalises well, as it reflects the same statistical properties as T_s and T_t. By assuming tasks are i.i.d., meta-learning frameworks can reliably evaluate the model's ability to adapt to new, unseen tasks from the same distribution, which is a meta-learning objective.

To achieve effective learning (Meta-learning phase), our method leverages the concept of NMF to decompose the data matrices $\mathbf{X}_{s_i}$ and $\mathbf{X}_t$ into non-negative factors that reveal latent structures in the data. On the other hand, the meta-learning framework allows the model to adapt and transfer knowledge across *meta-tasks* to enhance the performance on the *learning target task*. This approach ensures that the model can effectively generalize to the *ultimate generalization task* $\mathcal{T}_e$, as the decomposition and meta-learning processes jointly optimize the model's ability to handle unseen data from the same distribution (Evaluation phase). By leveraging the latent structures in $\mathbf{X}_e$ and the knowledge transferred from meta-tasks, our method ensures robust evaluation and generalization on the ultimate generalization dataset, aligning with the core objectives of meta-learning. In the next subsections, we will describe the key components of the methodology, including the process of task-specific NMF decomposition, the meta-learning algorithm for transferring knowledge, and how these components are integrated for improved unsupervised learning performance.

In the context of meta-learning, particularly applied to tasks like NMF, the model parameters ($i.e.$, $\mathbf{W}$ and $\mathbf{H}$) is trained on small datasets to learn initial parameters, which are then transferred and updated based on a learning target dataset. The update process is as follows:

3.2 Meta-Unsupervised Learning on Small Datasets:

The model uses small datasets (*meta-tasks*), to learn, in an unsupervised way, the factorized matrices $\mathbf{W}_{s_i}$ and $\mathbf{H}_{s_i}$ at each epoch k. The objective is to minimize the following loss function:

$$\mathcal{L}_{\text{meta}}\big((\mathbf{W}_{s_i}, \mathbf{H}_{s_i})^{(k+1)}\big) = \frac{1}{2}\|\mathbf{X}_{s_i} - (\mathbf{W}_t^{(k)} + \mathbf{W}_{s_i}^{(k)})\mathbf{H}_{s_i}^{(k)}\|_F^2; \forall i, \tag{3.1}$$

Minimizing the loss function in Eq. (3.1) can be approximated by reducing the Frobenius norm between the original dataset $\mathbf{X}_{s_i}$ and the product of the reconstructed factorized matrices $\mathbf{W}_{s_i}$ and $\mathbf{H}_{s_i}$, where $\mathbf{W}_{s_i}$ is augmented with $\mathbf{W}_t$, for each *meta-task*. Therefore, the learning of the factorized matrices for each *meta-task* is carried out as follows:

$$(\mathbf{W}_{s_i}, \mathbf{H}_{s_i}) = \min \frac{1}{2} \|\mathbf{X}_{s_i} - (\mathbf{W}_t + \mathbf{W}_{s_i})\mathbf{H}_{s_i}\|_F^2; \forall i. \tag{3.2}$$

As mentioned in Eq. (3.2), the previously learned $\mathbf{W}_t$ contributes to the initialization of each $\mathbf{W}_{s_i}$, facilitating a better localization of $\mathbf{W}_{s_i}$ in the parameter search space across epochs. This process can be interpreted as a form of knowledge transfer between the *learning target task* and the *meta-tasks*, enhancing convergence and improving factorization stability. Numerically solving Eq. (3.2) is performed using the NMF algorithm, following the same procedure as outlined in Eq. (2.1).

3.3 Meta-Unsupervised Learning on Larger (Target) Dataset:

After learning the parameters on the small datasets, the model is fine-tuned (initialized) on the larger target dataset $\mathbf{X}_t$. The learned parameters $(\mathbf{W}_t, \mathbf{H}_t)$ are updated by minimizing the following loss function:

$$\mathcal{L}\big((\mathbf{W}_t, \mathbf{H}_t)^{(k+1)}\big) = \mathbb{E}_{\mathcal{T}_{s_i} \sim p(\mathcal{T})} \left[\mathcal{L}_{\mathcal{T}_{s_i}} \big((\mathbf{W}_{s_i}, \mathbf{H}_{s_i}, \mathbf{W}_t)^{(k)}\big) \right], \tag{3.3}$$

where $\mathcal{L}_{\mathcal{T}_{s_i}}(.)$ is the loss for the i^{th} *meta-task*, and $p(\mathcal{T})$ is the distribution over the small datasets. The minimization of Eq. (3.3) can be carried out as follows:

$$(\mathbf{W}_t, \mathbf{H}_t) = \min \frac{1}{2} \|\mathbf{X}_t - (\mathbf{W}_t + \frac{1}{N} \sum_i \alpha_i \mathbf{W}_{s_i})\mathbf{H}_t\|_F^2. \tag{3.4}$$

As indicated in Eq. (3.4), the parameters $\mathbf{W}_{s_i}$ of the *meta-tasks* contribute to the initialization of the parameter $\mathbf{W}_t$ for subsequent learning stages on the *learning target-task*. Theoretically, the hyperparameter α is introduced to weight the parameter of each *meta-task* based on the accuracy of each $\mathbf{W}_{s_i}$ with respect to the reconstruction loss computed in Eq. (3.1). In practice, we set $\alpha_i = 1$ for all *meta-tasks*. Similarly to the Meta-unsupervised learning approach on small datasets, $\mathbf{W}_t$ is augmented by the average of $\mathbf{W}_{s_i}$, enabling knowledge sharing between the target NMF and the meta-NMFs applied to the meta-tasks. Numerically solving Eq. (3.4) is performed using the NMF algorithm, following the same procedure as outlined in Eq. (2.1). Ultimately, the learned matrices $\hat{\mathbf{W}}_t$ and $\hat{\mathbf{H}}_t$ are used to initialize the NMF model, which then factorizes the evaluation data, $\mathbf{X}_e$, to solve the generalization task $\mathcal{T}_e$.

Finally, the steps of the proposed Meta-unsupervised learning using NMF are presented in Algorithm (1). Figure 3.1 illustrates one epoch of the algorithm required to update the parameters for each *meta-task* as well as those of the *target-task*. This process is repeated for many epochs to progressively converge towards the optimal parameters $(\hat{\mathbf{W}}_t, \hat{\mathbf{H}}_t)$. The latter will be used as initial parameters for the final evaluation task.

Algorithm 1 Meta-Unsupervised Learning with NMF

```
Require: $\mathbf{X}_{s_i} \in \mathbb{R}^{m \times n_s}; i = 1..N$        ▷ Data matrix for each meta-task
Require: $\mathbf{X}_t \in \mathbb{R}^{m \times n_t}$        ▷ Data matrix for the learning target task
Require: $r$        ▷ Rank of factorization
Require: $\alpha$        ▷ Weighting factor for knowledge transfer
Require: $max\_epochs$        ▷ Maximum number of epochs
Ensure: $\hat{\mathbf{W}}_t, \hat{\mathbf{H}}_t$        ▷ Optimal factorized matrices for the learning target task

        META LEARNING PHASE

    # Meta-Learning on Meta-Tasks:
 1: Initialize: $\mathbf{W}_{s_i}, \mathbf{H}_{s_i}, \mathbf{W}_t, \mathbf{H}_t \sim \mathcal{U}(0,1)$
 2: for $i = 1..N$ do
 3:     Update $\mathbf{W}_{s_i}, \mathbf{H}_{s_i}$ defined Eq. (3.2)        ▷ Une NMF update rule
 4: end for
    # Meta-Transfer to Learning Target Task:
 5: Update $\mathbf{W}_t, \mathbf{H}_t$ defined in Eq. (3.4)        ▷ Une NMF update rule
 6: $\mathbf{W}_{s_i} \longleftarrow \mathbf{W}_{s_i} + \mathbf{W}_t$
 7: $\mathbf{H}_{s_i} = \mathbf{W}'_{s_i}\mathbf{X}_{s_i}$
 8: $\mathbf{W}_t \longleftarrow \mathbf{W}_t + \frac{1}{N}\sum_i \alpha_i \mathbf{W}_{s_i}$
 9: $\mathbf{H}_t = \mathbf{W}'_t\mathbf{X}_t$
10: for $epoch = 1$ to $max\_epochs$ do
11:     Repeat steps (2)-(7)
12: end for
13: $\hat{\mathbf{W}}_t \longleftarrow \mathbf{H}_t;\ \hat{\mathbf{H}}_t \longleftarrow \mathbf{W}_t$

        EVALUATION PHASE

Require: $\mathbf{X}_e \in \mathbb{R}^{m \times n_e}$        ▷ Evaluation set (ultimate target)
Ensure: $(\mathbf{W}_e, \mathbf{H}_e)$        ▷ Factorized matrices for the evaluation set
14: $(\mathbf{W}_e, \mathbf{H}_e) \longleftarrow \boldsymbol{nmf\_model}(\mathbf{X}_e, \hat{\mathbf{W}}_t, \hat{\mathbf{H}}_t)$
```

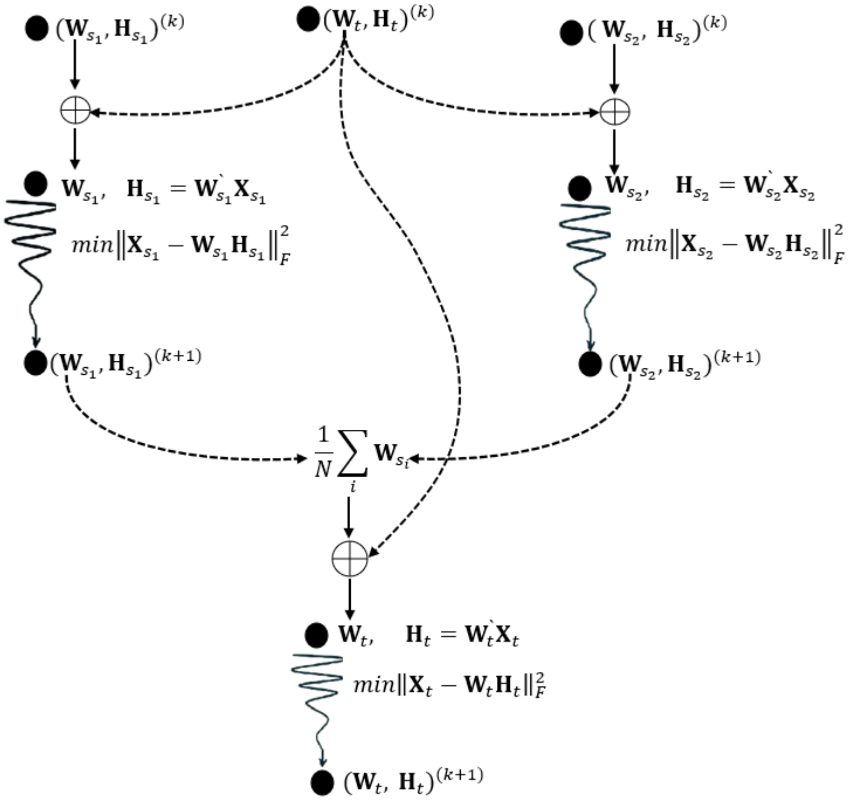

$(\mathbf{W}_{s_1}, \mathbf{H}_{s_1})^{(k)}$
$(\mathbf{W}_t, \mathbf{H}_t)^{(k)}$
$(\mathbf{W}_{s_2}, \mathbf{H}_{s_2})^{(k)}$
$\oplus$
$\oplus$
$\mathbf{W}_{s_1}, \quad \mathbf{H}_{s_1} = \mathbf{W}_{s_1}^{`}\mathbf{X}_{s_1}$
$min\|\mathbf{X}_{s_1} - \mathbf{W}_{s_1}\mathbf{H}_{s_1}\|_F^2$
$\mathbf{W}_{s_2}, \quad \mathbf{H}_{s_2} = \mathbf{W}_{s_2}^{`}\mathbf{X}_{s_2}$
$min\|\mathbf{X}_{s_2} - \mathbf{W}_{s_2}\mathbf{H}_{s_2}\|_F^2$
$(\mathbf{W}_{s_1}, \mathbf{H}_{s_1})^{(k+1)}$
$(\mathbf{W}_{s_2}, \mathbf{H}_{s_2})^{(k+1)}$
$\frac{1}{N}\sum_i \mathbf{W}_{s_i}$
$\oplus$
$\mathbf{W}_t, \quad \mathbf{H}_t = \mathbf{W}_t^{`}\mathbf{X}_t$
$min\|\mathbf{X}_t - \mathbf{W}_t\mathbf{H}_t\|_F^2$
$(\mathbf{W}_t, \mathbf{H}_t)^{(k+1)}$

Figure 3.1: Flowchart illustrating a single update epoch ($k \longrightarrow k+1$) for the factorized matrices in both the small datasets (2 for example) and the target dataset. The arrows pointing to $\mathbf{W}$ indicate that only $\mathbf{W}$ is explicitly computed, while the matrix $\mathbf{H}$ is implicitly computed from $\mathbf{W}^\top\mathbf{X}$. The spring arrows indicate the NMF optimization process used to learn the factorized matrices. The bullet marks indicate the positions of the learned factorized matrices within the parameter search space.

Chapter 4

Experiments, Analysis and Discussion

To evaluate the impact of the meta-learning paradigm on solving problems in unsupervised contexts, we apply this concept to multiple NMF variants, including the original NMF [43] and G-NMF [10]. Additionally, to benchmark the effectiveness of meta-learning against traditional parameter initialization methods, we include NMF-NNDSVD and GNMF-NNDSVD (Non-Negative Double Singular Value Decomposition) [6] as a comparative baseline. We conduct various experiments on different benchmark datasets described in Sec. 4.1 using different parameter settings as indicated in Sec. 4.2. The analysis and discussion of the achieved results based on the Normalized Mutual Information (NMI) metric of the models are provided in Sec. 4.3. Finally, Subjective and Interpretability Analysis is provided in Sec. 4.3.1.

4.1 Dataset description

Five benchmark datasets are used, namely, Blobs (synthetic), Digits, Fashion-MNIST (FMNIST), Olivetti Faces, and COIL-20 briefly described in Tab. 4.1.

Table 4.1: Key characteristics of benchmark datasets.

Dataset	# of samples	# of features	# of clusters
Blobs	10000	500	5
Digits	1797	64	10
FMNIST	6000	784	10
Olivetti Faces	400	1024	40
COIL20	1440	1024	20

4.2 Task preparation protocol and Parameter Setting

As outlined above, each task corresponds to a dataset. In our experiment, assuming the i.i.d. assumption, each dataset is randomly divided into two sets: a learning set (50%) and an evaluation set (50%). The learning set is further divided into two parts: five *meta-sets*, each containing 50 samples and a target learning set (rest of the data). Task split protocol can be seen in Fig. 4.1.

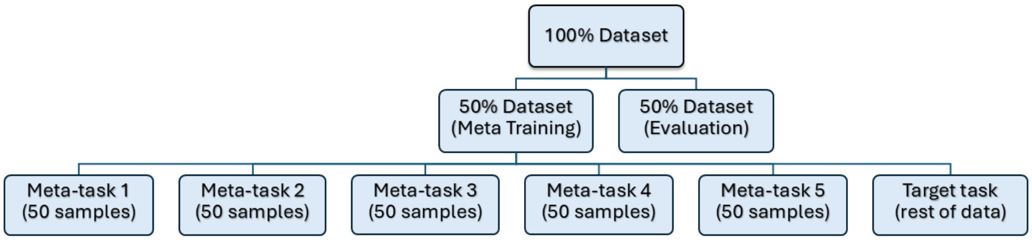


Figure 4.1: Task Split Protocol.

The evaluation set is used as the ultimate set considered for evaluation by all models. These include the baseline NMF-based models (NMF, GNMF, GNMF-NNDSVD, and NMF-NNDSVD), as well as their Meta models (Meta-NMF, Meta-GNMF, Meta-GNMF-NNDSVD, and Meta-NMF-NNDSVD). The key difference between Meta models and the baseline models lies in their initialization methods. This says that Meta-models use initial parameters learned by the meta-unsupervised method from the learning and Metas sets; the NMF and GNMF utilize random initialization, while the NMF-NNDSVD and GNMF-NNDSVD use Singular Value Decomposition (SVD) for initialization. To optimize the learning phase, we configure the following parameters: the factorization rank r varies from 2 to 18, the number of epochs is set to 50, and the maximum number of iterations for the NMF factorization is set to 100. For the base NMF models, all of them are limited to 100 iterations (default configuration of the original implementation). These settings are summarized in Table 4.2.

Table 4.2: Parameter Settings.

Parameter	Value
rank (r)	[2, 3, 5, 10, 14, 18]
outer epochs	50
inner epochs	100
number of *meta-task*s	5
size of *meta-task*s	50

4.3 Objective Analysis

Tables [4.3, 4.4, 4.5, 4.6] present the mean NMI scores over ranks for different models under noise-free conditions, whereas tables [4.7, 4.8, 4.10, 4.9] present the mean NMI scores across ranks for different models under noisy conditions. The analysis highlights key patterns in model performance, meta-learning effectiveness, and noise's impact on clustering results. Meta-learning generally enhances the performance of standard NMF models by improving initialization and generalization across tasks. However, the effectiveness of the meta-learning framework varies depending on the dataset and the initialization method used.

Dataset	NMF	Meta-NMF
Blobs	0.086	**0.869**
Coil20	0.718	**0.720**
Digits	0.570	**0.595**
Faces	0.665	**0.668**
FMNIST	**0.503**	0.499

Table 4.3: Clustering performance of NMF and Meta-NMF on data without noise, using mean NMI over ranks

Dataset	GNMF	Meta-GNMF
Blobs	0.073	**0.273**
Coil20	0.694	**0.719**
Digits	0.541	**0.601**
Faces	0.665	**0.675**
FMNIST	**0.490**	0.479

Table 4.4: Clustering performance of GNMF and Meta-GNMF on data without noise, using mean NMI over ranks

Dataset	GNMF NNDSVD	Meta-GNMF NNDSVD
Blobs	0.804	**0.849**
Coil20	0.691	**0.717**
Digits	0.523	**0.585**
Faces	0.646	**0.675**
FMNIST	0.488	**0.493**

Table 4.5: Clustering performance of GNMF NNDSVD and Meta-GNMF NNDSVD on data without noise, using mean NMI over ranks

Dataset	NMF NNDSVD	Meta-NMF NNDSVD
Blobs	0.759	0.759
Coil20	**0.702**	0.701
Digits	**0.520**	0.519
Faces	0.643	**0.646**
FMNIST	**0.491**	0.490

Table 4.6: Clustering performance of NMF NNDSVD and Meta-NMF NNDSVD on data without noise, using mean NMI over ranks

Dataset	NMF	Meta-NMF
Blobs	0.061	**0.798**
Coil20	0.615	**0.664**
Digits	0.117	**0.181**
Faces	0.557	**0.591**
FMNIST	0.270	**0.483**

Table 4.7: Clustering performance of NMF and Meta-NMF on data with added noise, using mean NMI over ranks

- In most cases, Meta-NMF outperforms NMF where it significantly improves performance on the Blobs dataset, achieving the highest NMI scores both with and without noise (0.869 vs. 0.086 in noise-free, 0.798 vs. 0.061 in noisy). The substantial improvement indicates that meta-learning successfully refines initialization, helping the model find a better factorization. However, on Fashion MNIST, Meta-NMF provides little to no improvement over standard NMF in the noise-free case (0.499 vs. 0.503).

- Meta-GNMF generally outperforms GNMF in both conditions with or without noise. Meta-GNMF performs best on the Digits dataset without noise, slightly outperforming all other models (0.601). This suggests that meta-learning helps refine graph-based NMF representations, leading to improved clustering accuracy. However, it struggles in some cases, as its advantage over GNMF diminishes on Fashion MNIST without noise (0.479 vs. 0.490) and Digits with noise (0.116 vs. 0.066).

- Meta-GNMF NNDSVD is the most robust meta-learning model in both conditions with or without noise where it outperforms its baseline model GNMF NNDSVD on all datasets. It achieves the highest performance on the Faces dataset without noise and on most datasets with noise. This indicates that meta-learning enhances generalization when dealing with complex or corrupted data.

- Meta-NMF NNDSVD performs similarly to its baseline NMF NNDSVD and fails to provide significant improvement unlike the other Meta-learning NMF-based

Dataset	GNMF	Meta-GNMF
Blobs	0.057	**0.093**
Coil20	0.608	**0.635**
Digits	**0.116**	0.066
Faces	0.555	**0.598**
FMNIST	0.274	**0.335**

Table 4.8: Clustering performance of GNMF and Meta-GNMF on data with added noise, using mean NMI over ranks

Dataset	NMF NNDSVD	Meta-NMF NNDSVD
Blobs	0.745	0.745
Coil20	0.611	0.611
Digits	**0.186**	0.185
Faces	**0.571**	0.570
FMNIST	**0.452**	0.450

Table 4.9: Clustering performance of NMF NNDSVD and Meta-NMF NNDSVD on data with added noise, using mean NMI over ranks

models, which consistently outperform their baselines. This performance can be explained that the NNDSVD initialization in the case of NMF seems to lead to poor local minima, making meta-learning have little room to explore alternative solutions.

Dataset	GNMF NNDSVD	Meta-GNMF NNDSVD
Blobs	0.712	**0.772**
Coil20	0.599	**0.674**
Digits	0.199	**0.200**
Faces	0.566	**0.599**
FMNIST	0.453	**0.491**

Table 4.10: Clustering performance of GNMF NNDSVD and Meta-GNMF NNDSVD on data with added noise, using mean NMI over ranks

Figures [4.2, 4.3, 4.4, 4.5, 4.6] present the NMI scores obtained across different rank settings on all datasets under noise-free conditions. The results indicate that, on average, the meta-learning-enhanced variants outperform their baseline counterparts by approximately $5 - 15\%$. Under noisy conditions, Figures [4.7, 4.8, 4.9, 4.10, 4.11] show that the performance gap becomes even more pronounced, with meta-learning variants achieving up to 20% higher NMI scores. Notably, Meta-NMF and Meta-GNMF-NNDSVD demonstrate superior performance, emphasizing the effectiveness of unsupervised meta-learning in capturing more nuanced data structures.

This improvement enhances the models' ability to learn meaningful representations while increasing robustness against noise. This may occur because meta-learning and meta-transfer of parameters help filter out noise from the components across learning epochs.

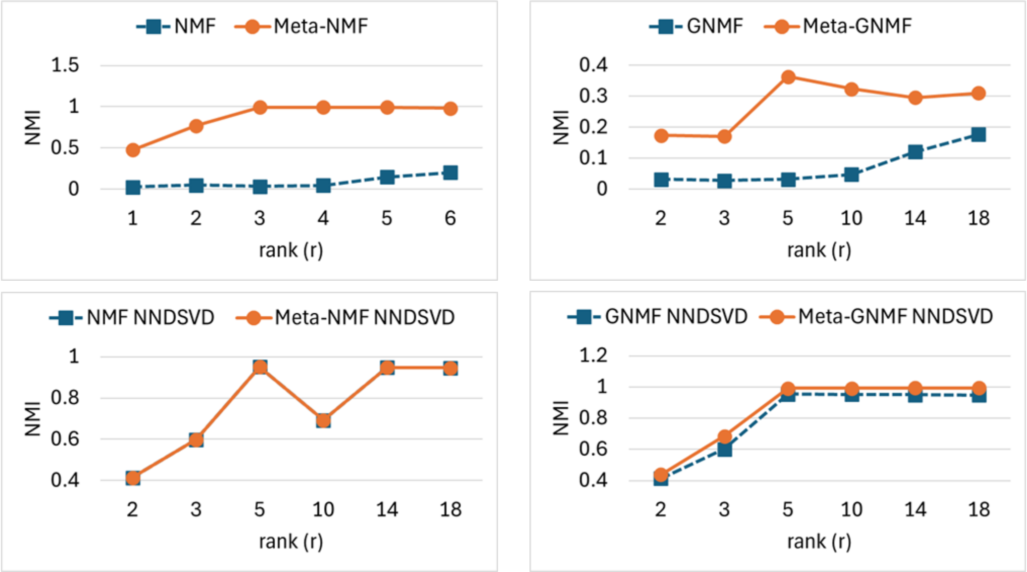


Figure 4.2: Comparison of NMF-based models with their Meta variants with no noise added to Blobs dataset.

The effectiveness of meta-learning models varies depending on the selected rank and noise conditions. The higher ranks ($r = 14, 18$) generally lead to better clustering performance across all datasets, with Meta-NMF and its variants showing the most substantial gains. The meta-learning framework also demonstrates improved noise sensitivity, as evidenced by the smaller performance degradation in noisy conditions compared to standard NMF and GNMF models. Among all methods, Meta-NMF and Meta-GNMF NNDSVD consistently achieve the highest NMI scores. The improvements achieved by the meta-learning framework can be attributed to its generalization and adaptability. By learning from smaller datasets, the meta-learning approach provides better parameter initialization, reducing sensitivity to initialization and guiding the optimization process toward more stable solutions. This adaptability allows them to maintain high performance under various conditions and datasets, making the framework particularly effective for clustering and dimensionality reduction tasks. However, there are some cases preventing meta-learning from making meaningful refinements like in the case of NMF NNDSVD. This further supports the idea that if a model already tends to fall in poor local minima limiting the flexibility of meta-learning to improve results.

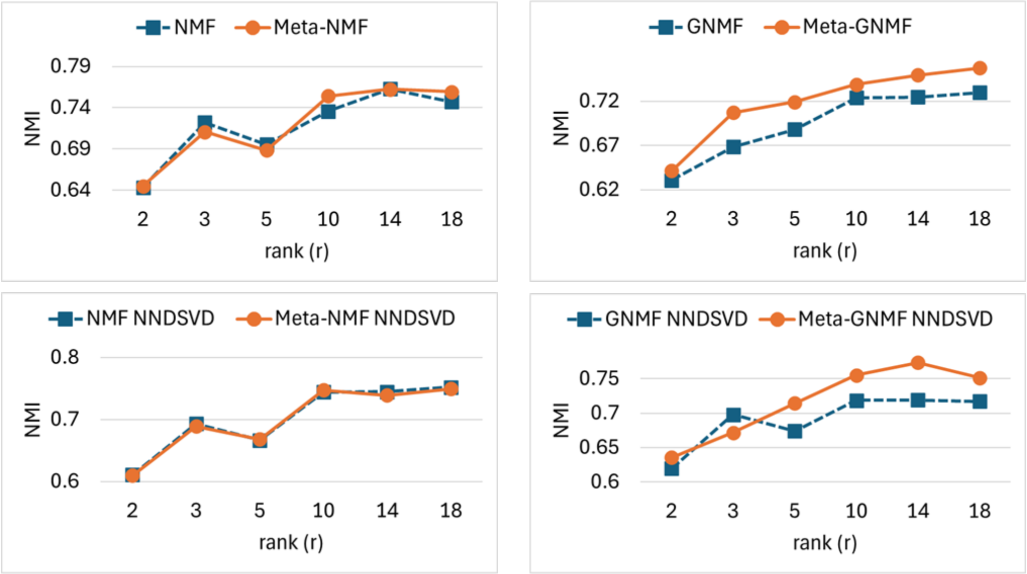


Figure 4.3: Comparison of NMF-based models with their Meta variants with no noise added to COIL20 dataset.

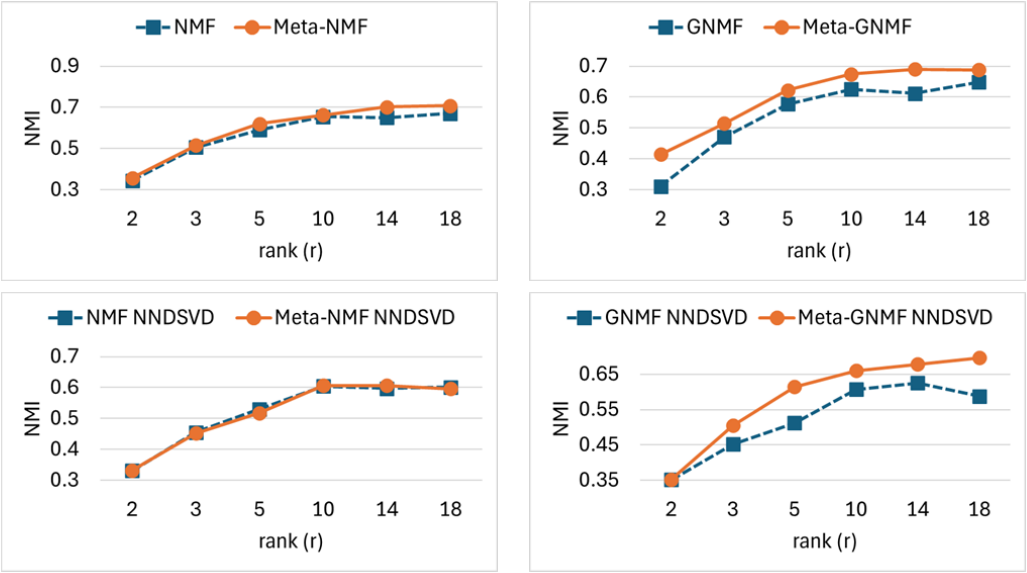


Figure 4.4: Comparison of NMF-based models with their Meta variants with no noise added to Digits dataset.

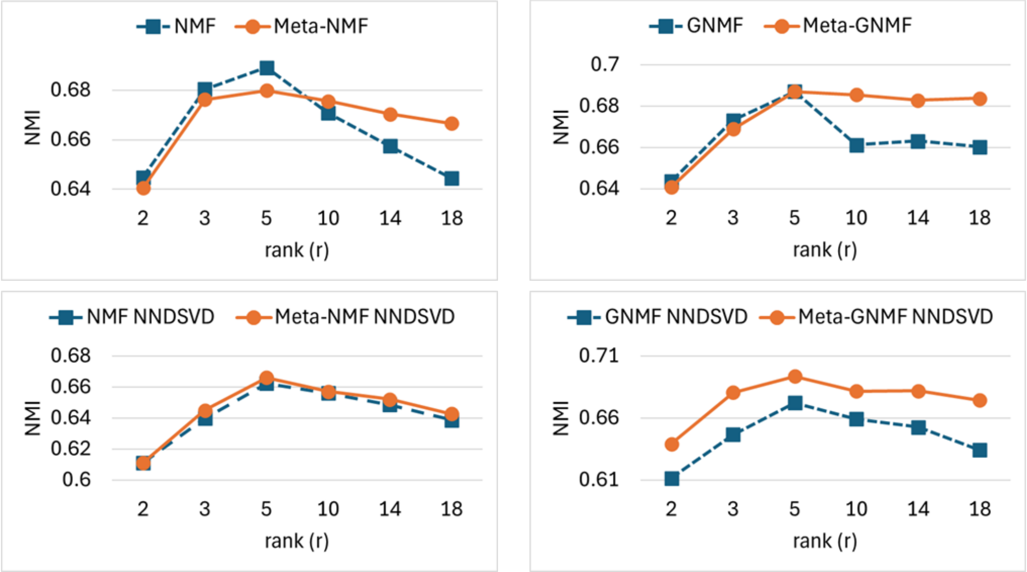


Figure 4.5: Comparison of NMF-based models with their Meta variants with no noise added to Faces dataset.

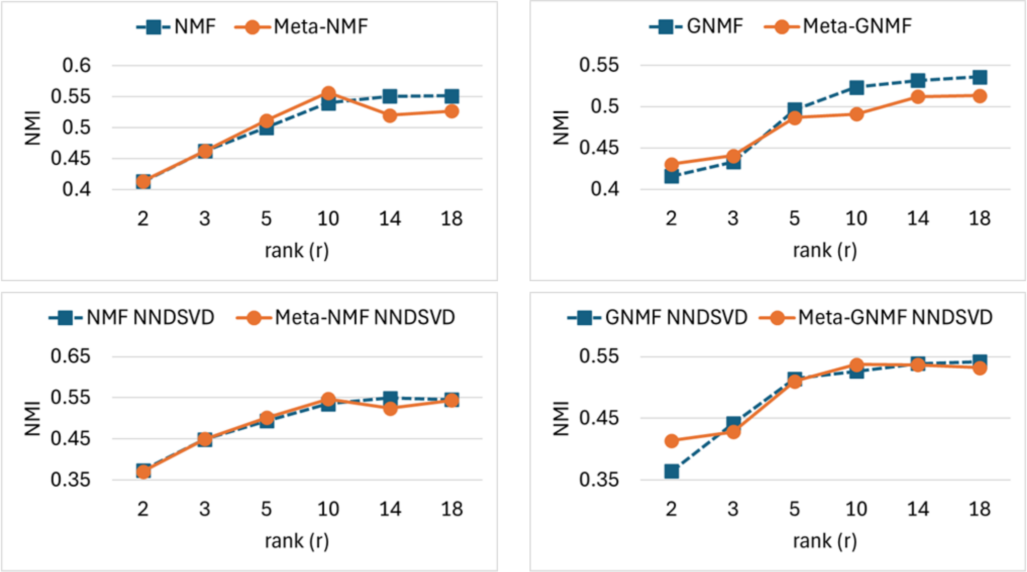


Figure 4.6: Comparison of NMF-based models with their Meta variants with no noise added to FMNIST dataset.

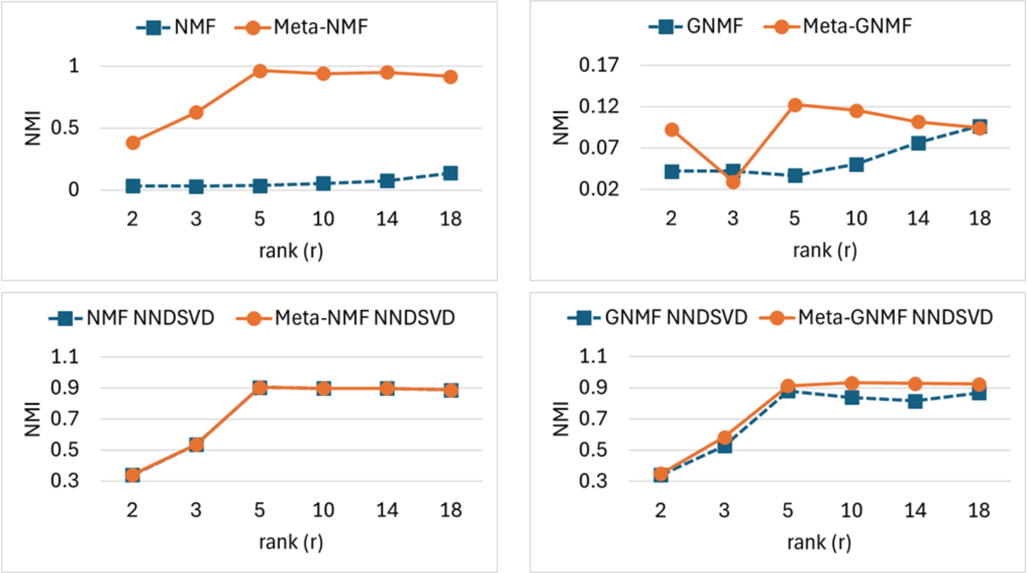


Figure 4.7: Comparison of NMF-based models with their Meta variants with noise added to Blobs dataset.

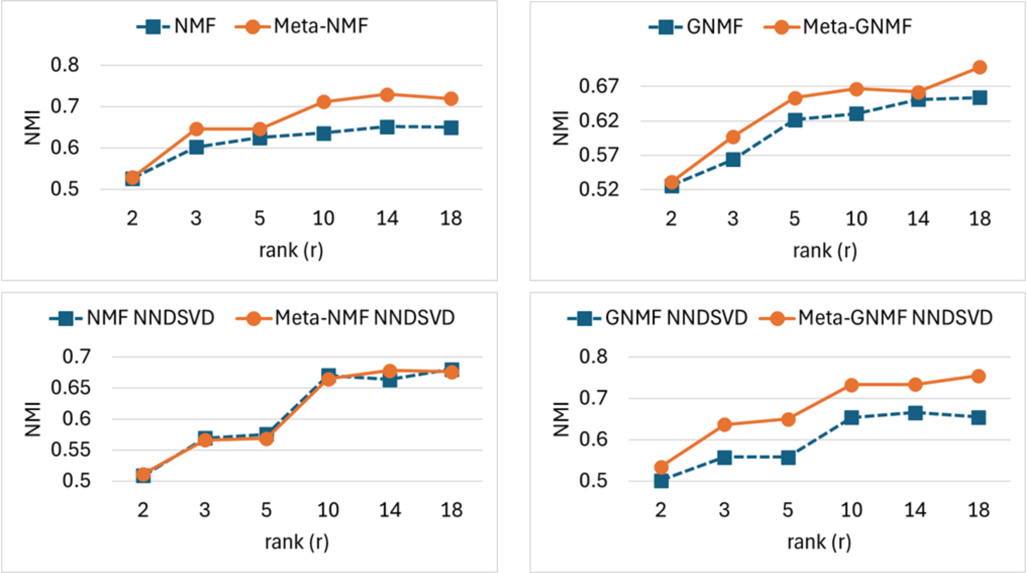


Figure 4.8: Comparison of NMF-based models with their Meta variants with noise added to COIL20 dataset.

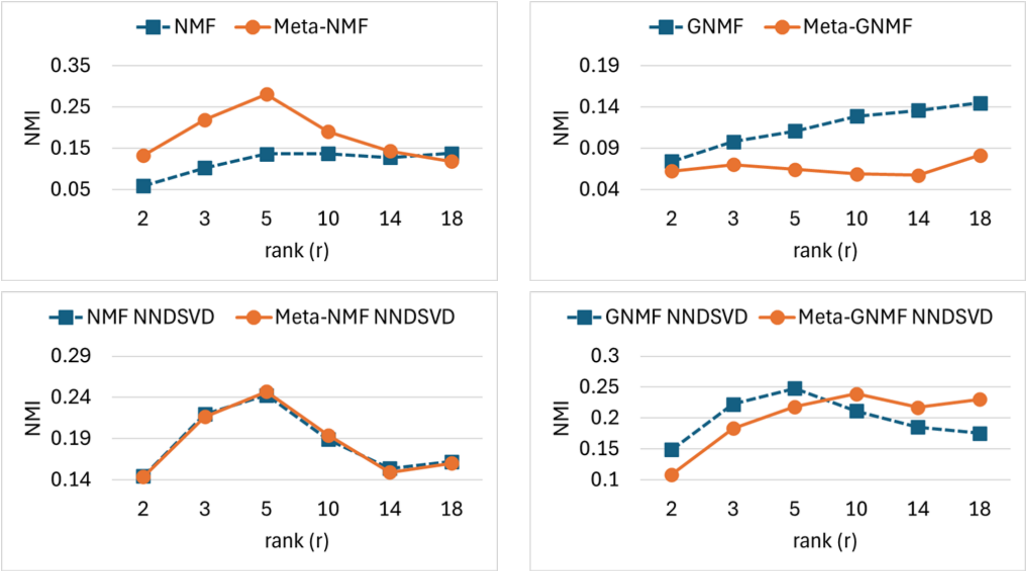


Figure 4.9: Comparison of NMF-based models with their Meta variants with noise added to Digits dataset.

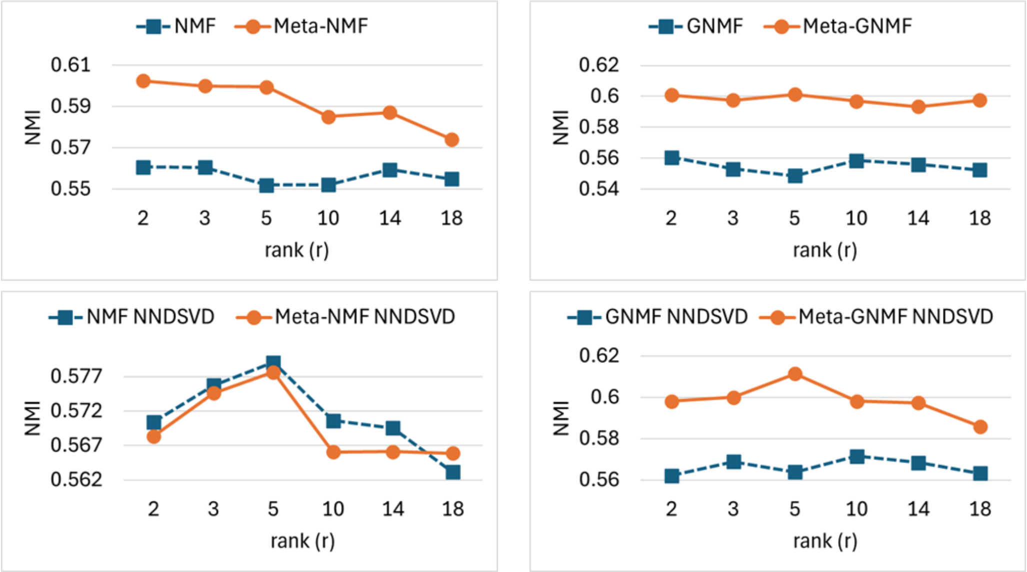


Figure 4.10: Comparison of NMF-based models with their Meta variants with noise added to Faces dataset.

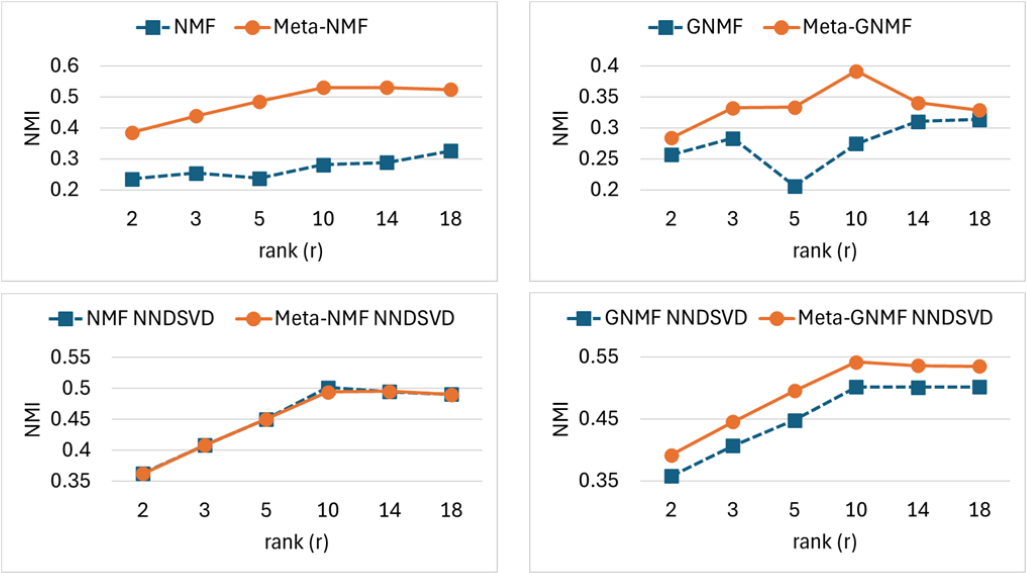


Figure 4.11: Comparison of NMF-based models with their Meta variants with noise added to FMNIST dataset.

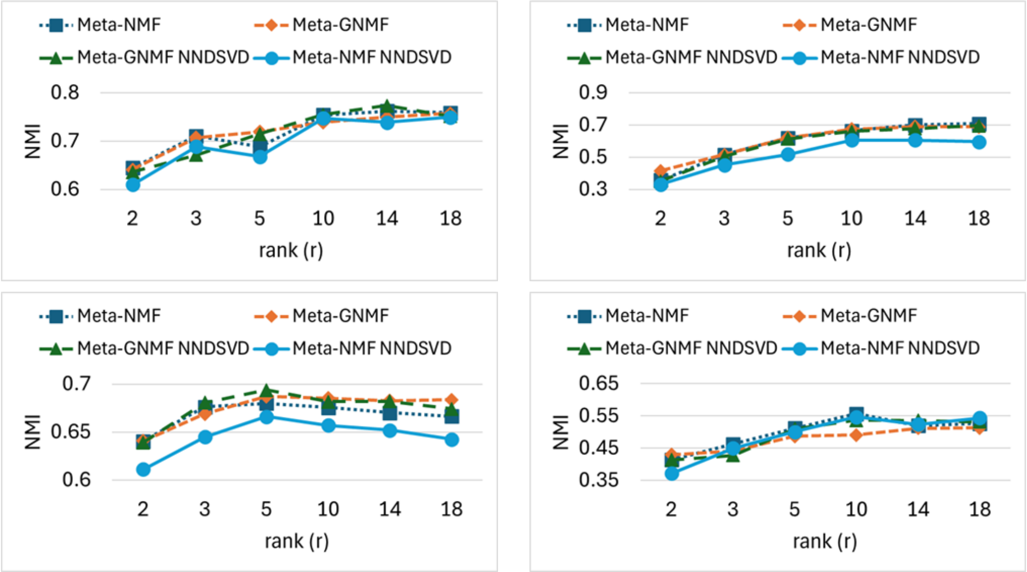


Figure 4.12: Comparison of Meta-based NMF models based on real-world benchmark datasets without noise (left to right), COIL20, Digits, Faces, FMNIST.

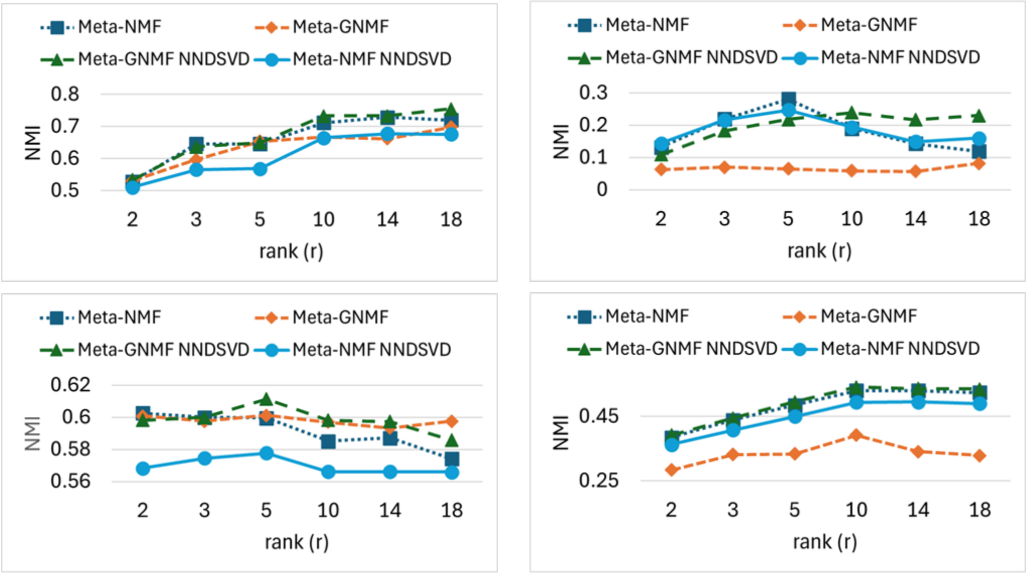


Figure 4.13: Comparison of Meta-based NMF models based on real-world benchmark datasets with noise (left to right), COIL20, Digits, Faces, FMNIST.

To investigate how each unsupervised learning model (NMF baseline) leverages the power of meta-learning, we present the performance of the meta-models in the same plots. Figures 4.12 and 4.13 compare the performance of various Meta-based NMF models across the real-life benchmark datasets under two conditions: noise-free (Fig. 4.12) and noisy (Fig. 4.13), using the NMI evaluation metric. In the noise-free setting, all meta-based models except Meta-NMF NNDSVD, which exhibits a slight deficiency yield similarly high-quality and interpretable latent representations. However, when noise is introduced, clear differences emerge. Meta-GNMF NNDSVD demonstrates the highest robustness, producing the clearest and most distinct components, followed by Meta-NMF. In contrast, Meta-GNMF and Meta-NMF NNDSVD consistently lag behind the others in noisy conditions. These results suggest that while the meta-learning models perform comparably under ideal conditions, Meta-GNMF NNDSVD and Meta-NMF show their superiority in noisy environments, highlighting their enhanced resilience to noise and ability to preserve feature clarity.

4.3.1 Subjective and Interpretability Analysis

To comprehensively evaluate the effectiveness of meta-learning in enhancing unsupervised learning for discovering meaningful facial components, we conducted a visual assessment using the Olivetti face dataset. This dataset enables empirical evaluation of how well different models extract interpretable facial parts.

We performed comparisons between standard NMF-based approaches and their meta-learning-enhanced counterparts under both noise-free and noisy conditions to quantify the impact of meta-learning on representation quality and robustness.

Our analysis of the extracted components reveals that meta-learning significantly improves the quality of facial part representation. As shown in Fig. 4.14, standard NMF (top) tends to extract more holistic facial representations with less distinct separation between facial features. In contrast, Meta-NMF (bottom) demonstrates superior capability in isolating specific facial components such as eyes, mouth, and nose regions with greater clarity. When noise is introduced to the dataset (Fig. 4.15), standard NMF (top) exhibits considerable degradation in extraction quality, while Meta-NMF (bottom) maintains relatively consistent feature extraction, suggesting enhanced robustness against data corruption.

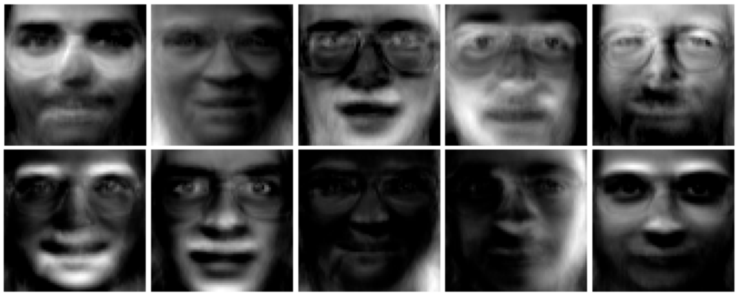


Figure 4.14: Face parts extracted by NMF (top) and Meta-NMF (bottom), without added noise.

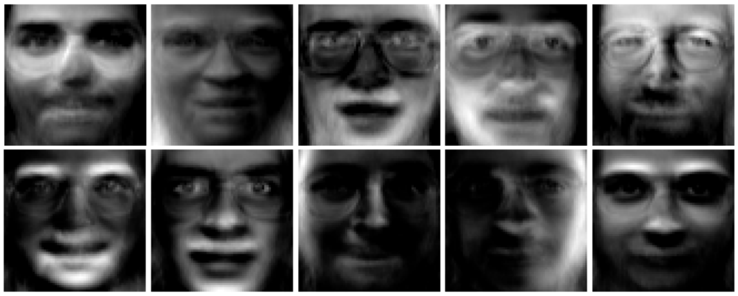


Figure 4.15: Face parts extracted by NMF (top) and Meta-NMF (bottom), with added noise.

The integration of graph regularization into NMF further highlights the benefits of meta-learning. In Fig. 4.16, standard GNMF (top) extracts facial components with reasonable quality but contains visible artifacts and less distinct feature separation. Meta-GNMF (bottom) demonstrates more refined extraction capability, producing cleaner representations of facial features like eyes, glasses, and mouth regions with improved visual interpretability. The robustness advantage becomes even more apparent when noise is introduced (Fig. 4.17), where GNMF (top) shows significant degradation while Meta-GNMF (bottom) maintains the integrity of extracted facial components despite the presence of noise.

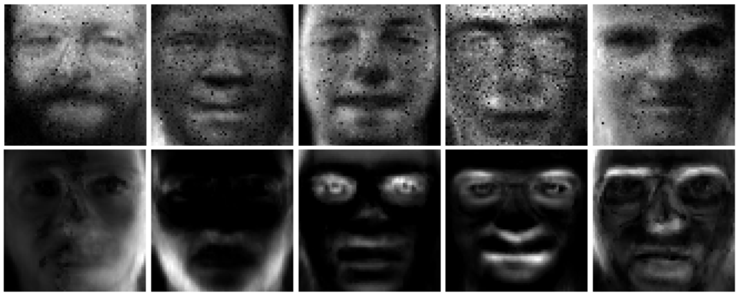


Figure 4.16: Face parts extracted by GNMF (top) and Meta-GNMF (bottom), without added noise.

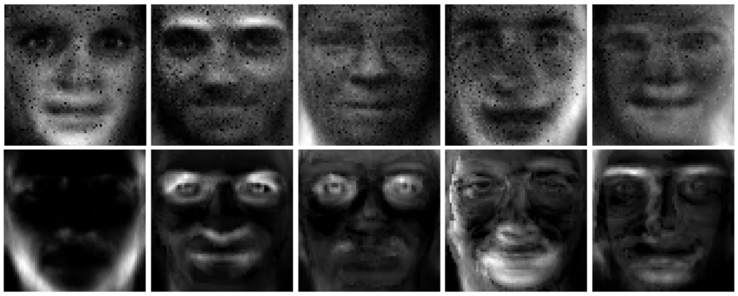


Figure 4.17: Face parts extracted by GNMF (top) and Meta-GNMF (bottom), with added noise.

The incorporation of Non-Negative Double Singular Value Decomposition (NNDSVD) initialization provides additional improvements in component extraction. As illustrated in Fig. 4.18, NMF NNDSVD (top) extracts facial components with improved definition compared to standard NMF, but Meta-NMF NNDSVD (bottom) achieves even greater distinction in isolating specific facial elements. When subjected to noise (Fig. 4.19), NMF NNDSVD (top) shows noticeable degradation with increased artifacts, while Meta-NMF NNDSVD (bottom) preserves the integrity of facial component extraction remarkably well.

The combination of graph regularization with NNDSVD initialization represents the most sophisticated approach examined in our study. In Fig. 4.20, GNMF NNDSVD (top) extracts facial components with good quality but still exhibits some artifacts and less precise feature isolation. Meta-GNMF NNDSVD (bottom) demonstrates superior performance in extracting clean, distinct facial features with minimal artifacts and greater interpretability. Fig. 4.21 provides compelling evidence for the robustness of meta-learning approaches in the presence of noise, where Meta-GNMF NNDSVD (bottom) maintains remarkably consistent quality in preserving distinctive facial components such as eyes with glasses, mouth, and face contours with minimal corruption from noise, while GNMF NNDSVD (top) shows considerable degradation.

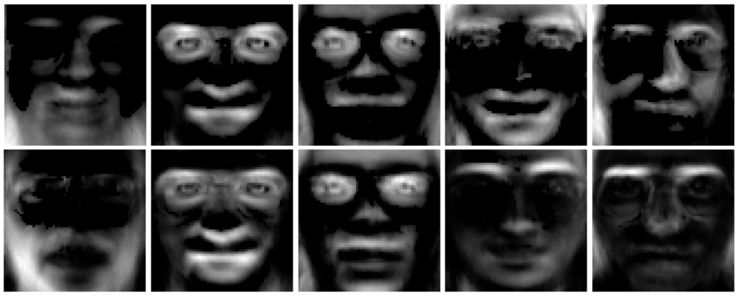


Figure 4.18: Face parts extracted by NMF NNDSVD (top) and Meta-NMF NNDSVD (bottom), without added noise.

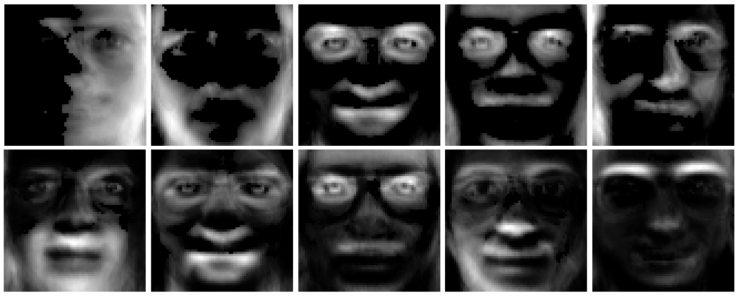


Figure 4.19: Face parts extracted by NMF NNDSVD (top) and Meta-NMF NNDSVD (bottom), with added noise.

These findings strongly support our hypothesis that meta-learning facilitates more effective noise filtering over successive training epochs, resulting in more stable and interpretable part-based representations. The meta-learning framework demonstrates particularly strong robustness in noisy conditions, where standard approaches often fail to maintain extraction quality. By enabling knowledge transfer across tasks, meta-learning significantly enhances the capability of NMF-based models to extract meaningful facial components, even in challenging environments with data corruption. The consistent improvement observed across different model configurations suggests that meta-learning offers a generalizable enhancement to unsupervised learning for part-based representation discovery.

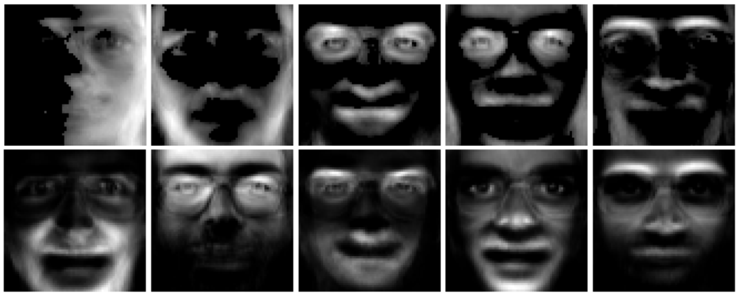


Figure 4.20: Face parts extracted by GNMF NNDSVD (top) and Meta-GNMF NNDSVD (bottom), without added noise.

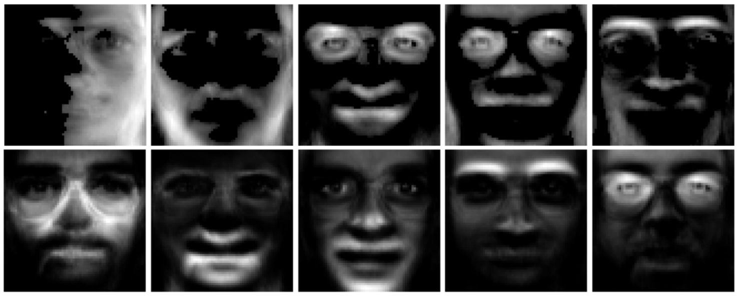


Figure 4.21: Face parts extracted by GNMF NNDSVD (top) and Meta-GNMF NNDSVD (bottom), with added noise.

Chapter 5

Conclusion and Future Directions

In this study, we have explored the integration of meta-learning with unsupervised learning, using NMF as a case study to address key challenges inherent in unsupervised tasks. By applying meta-unsupervised learning, we demonstrated that NMF could leverage prior knowledge from smaller factorization tasks to enhance its performance on larger, more complex datasets. The proposed Meta-NMF approach offers several key benefits: it reduces sensitivity to initialization, improves convergence stability, and enhances the generalizability of the model, enabling it to adapt more effectively to diverse datasets without requiring extensive parameter tuning. Additionally, the meta-learning approach allows for the filtering of noise over the course of learning epochs through meta-learning and meta-transfer of parameters, leading to the identification of more interpretable and meaningful object parts. This capability makes the model more robust to noise, ensuring that even in noisy conditions, the factorization remains clear and distinct. The results suggest that meta-learning has a significant impact on unsupervised learning tasks, such as clustering and dimensionality reduction, by improving optimization efficiency and reducing computational demands. Overall, our work underscores the potential of meta-learning to address critical limitations in unsupervised learning, offering a path toward more adaptive, resilient, and autonomous models that can extract meaningful representations from unlabeled data across a variety of domains.
Future will extend the meta-unsupervised learning framework to other unsupervised techniques like clustering and dimensionality reduction, improving noise robustness and stability, and enhancing scalability for large-scale datasets. Further research could focus on integrating domain-specific knowledge for more targeted learning, enhancing interpretability through explainable AI methods, and incorporating multi-task or dynamic learning capabilities. These directions would strengthen the framework's generalization across diverse applications, improve its adaptability, and make it more efficient in real-world settings.

Although this study has demonstrated the effectiveness of NMF across different domains, some limitations should be addressed while proceeding to future work.

Sensitivity to initialization is one major limitation of NMF, where incorrect initialization can result in inconsistent factorization results. Future research needs to explore improved initialization techniques or adaptive learning mechanisms that change parameters dynamically during factorization. Moreover, the computational complexity of NMF is still considered a problem, especially for high-scale datasets. Future studies could use other more efficient optimization algorithms, such as parallel computing approaches or a hybrid model of NMF and deep learning frameworks to scale it further.

Furthermore, the usage of NMF in dynamic or real-world environments has not been extensively explored. Future research could apply NMF for streaming applications with evolving data structures, real-time signal processing, and financial modeling. NMF can be further applied in interdisciplinary fields like quantum computing and materials science to open new doors and show the NMF's versatility in solving complex scientific and engineering problems.

References

[1] S. Alberuni and S. Ray. "Integration of biological data via NMF for identification of human disease-associated gene modules through multi-label classification". In: *PLoS One* 19.12 (2024), e0305503.

[2] T. Aonishi et al. "Imaging Data Analysis Using Non-Negative Matrix Factorization". In: *Neuroscience Research* 179 (2022), pp. 51–56.

[3] Fiorenzo Artoni, Arnaud Delorme, and Scott Makeig. "Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition". In: *NeuroImage* 175 (2018), pp. 176–187.

[4] J. Biswas, P. Kayal, and D. Samanta. "Reducing Approximation Error with Rapid Convergence Rate for Non-Negative Matrix Factorization (NMF)". In: *Mathematics and Statistics* 9.3 (2021), pp. 285–289.

[5] J. Bobadilla et al. "Recommender systems clustering using Bayesian non negative matrix factorization". In: *IEEE Access* 6 (2017), pp. 3549–3564.

[6] C. Boutsidis and E. Gallopoulos. "SVD-based initialization: A head start for nonnegative matrix factorization". In: *Pattern Recognition* 41.4 (2008), pp. 1350–1360.

[7] Jean-Philippe Brunet et al. "Metagenes and molecular pattern discovery using matrix factorization". In: *Proceedings of the national academy of sciences* 101.12 (2004), pp. 4164–4169.

[8] Juan José Burred. "Detailed derivation of multiplicative update rules for NMF". In: *Paris, France* (2014).

[9] Davide Cacciarelli and Murat Kulahci. "Hidden dimensions of the data: PCA vs autoencoders". In: *Quality Engineering* 35.4 (2023), pp. 741–750.

[10] Deng Cai et al. "Graph regularized nonnegative matrix factorization for data representation". In: *IEEE transactions on pattern analysis and machine intelligence* 33.8 (2010), pp. 1548–1560.

[11] Zongjie Cao et al. "The feasibility analysis of applying NMF in SAR target recognition". In: *2015 IEEE International Conference on Digital Signal Processing (DSP)*. IEEE. 2015, pp. 721–725.

[12] B. Chen. "Partial Multi-View Clustering via Meta-Learning and Contrastive Feature Alignment". In: *arXiv preprint arXiv:2411.09758* (2024).

[13] W.-S. Chen et al. "Symmetric Nonnegative Matrix Factorization: A Systematic Review". In: *Neurocomputing* (2023), p. 126721.

[14] Y. Chen et al. "Experimental explorations on short text topic mining between LDA and NMF based Schemes". In: *Knowledge-Based Systems* 163 (2019), pp. 1–13.

[15] Seungjin Choi. "Algorithms for orthogonal nonnegative matrix factorization". In: *2008 ieee international joint conference on neural networks (ieee world congress on computational intelligence)*. IEEE. 2008, pp. 1828–1832.

[16] Andrej Čopar, Blaž Zupan, and Marinka Zitnik. "Fast optimization of non-negative matrix tri-factorization". In: *PloS one* 14.6 (2019), e0217994.

[17] G. Cui et al. "Semi-Supervised Multi-View Clustering Based on NMF with Fusion Regularization". In: *ACM Transactions on Knowledge Discovery from Data* 18.6 (2024), pp. 1–26.

[18] K. Devarajan. "Nonnegative matrix factorization: an analytical and interpretive tool in computational biology". In: *PLoS Computational Biology* 4.7 (2008), e1000029.

[19] R. Dubroca, C. Junor, and A. Souloumiac. "Weighted NMF for high-resolution mass spectrometry analysis". In: *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*. 2012, pp. 1806–1810.

[20] J. F. Einsle et al. "All mixed up: Using machine learning to address heterogeneity in (natural) materials". In: *Microscopy and Microanalysis* 24.S1 (2018), pp. 562–563.

[21] F. Esposito. "A Review on Initialization Methods for Nonnegative Matrix Factorization: Towards Omics Data Experiments". In: *Mathematics* 9.9 (2021), p. 1006.

[22] Xiao Fu et al. "Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications." In: *IEEE Signal Process. Mag.* 36.2 (2019), pp. 59–80.

[23] V. Gallego et al. "Applying Machine Learning in Marketing: An Analysis Using the NMF and k-Means Algorithms". In: *Information* 15.7 (2024), p. 368.

[24] J. Gan et al. "Non-Negative Matrix Factorization: A Survey". In: *The Computer Journal* 64.7 (2021), pp. 1080–1092.

[25] Nicolas Gillis and François Glineur. "On the geometric interpretation of the nonnegative rank". In: *Linear Algebra and its Applications* 437.11 (2012), pp. 2685–2712.

[26] Michael Greenacre et al. "Principal component analysis". In: *Nature Reviews Methods Primers* 2.1 (2022), p. 100.

[27] B. Gyenes and A. E. Brown. "Deriving shape-based features for C. elegans locomotion using dimensionality reduction methods". In: *Frontiers in Behavioral Neuroscience* 10 (2016), p. 159.

[28] Basna Mohammed Salih Hasan and Adnan Mohsin Abdulazeez. "A review of principal component analysis algorithm for dimensionality reduction". In: *Journal of Soft Computing and Data Mining* 2.1 (2021), pp. 20–30.

[29] Haonan Huang et al. "Comprehensive multiview representation learning via deep autoencoder-like nonnegative matrix factorization". In: *IEEE Transactions on Neural Networks and Learning Systems* 35.5 (2023), pp. 5953–5967.

[30] T. Iwata. "Meta-Learning for Matrix Factorization without Shared Rows or Columns". In: *arXiv preprint arXiv:2106.15133* (2021).

[31] Weikuan Jia et al. "Feature dimensionality reduction: a review". In: *Complex & Intelligent Systems* 8.3 (2022), pp. 2663–2693.

[32] Ian T Jolliffe and Jorge Cadima. "Principal component analysis: a review and recent developments". In: *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences* 374.2065 (2016), p. 20150202.

[33] I. Jung, J. Choi, and H. Chae. "A Non-Negative Matrix Factorization-Based Framework for the Analysis of Multi-Class Time-Series Single-Cell RNA-Seq Data". In: *IEEE Access* 8 (2020), pp. 42342–42348.

[34] R. Kannan et al. "Deep data analysis via physically constrained linear unmixing: universal framework, domain examples, and a community-wide platform". In: *Advanced Structural and Chemical Imaging* 4 (2018), pp. 1–20.

[35] Ameer Ahmed Khan et al. "Meta-Unsupervised Representation Learning: A New Approach to Factorization and Interpretability". In: *Proceedings of Canadian AI 2025*. Canadian AI 2025, 2025.

[36] Z. Khan et al. "Enriching Non-Negative Matrix Factorization with Contextual Embeddings for Recommender Systems". In: *Neurocomputing* 380 (2020), pp. 246–258.

[37] Siavash Khodadadeh, Ladislau Boloni, and Mubarak Shah. "Unsupervised meta-learning for few-shot image classification". In: *Advances in neural information processing systems* 32 (2019).

[38] Jingu Kim and Haesun Park. *Sparse nonnegative matrix factorization for clustering*. Tech. rep. Georgia Institute of Technology, 2008.

[39] S. Latif, F. Shafait, and R. Latif. "Analyzing LDA and NMF Topic Models for Urdu Tweets via Automatic Labeling". In: *IEEE Access* 9 (2021), pp. 127531–127547.

[40] T. Laudadio et al. "NMF in MR spectroscopy". In: *Non-negative Matrix Factorization Techniques: Advances in Theory and Applications* (2016), pp. 161–177.

[41] Jonathan Le Roux, Felix J Weninger, and John R Hershey. "Sparse NMF–half-baked or well done?" In: *Mitsubishi Electric Research Labs (MERL), Cambridge, MA, USA, Tech. Rep., no. TR2015-023* 11 (2015), pp. 13–15.

[42] Daniel Lee and H. Sebastian Seung. "Algorithms for Non-negative Matrix Factorization". In: *Advances in Neural Information Processing Systems 13 (NeurIPS)*. 2000.

[43] Daniel D Lee and H Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization". In: *nature* 401.6755 (1999), pp. 788–791.

[44] T. Lekshmi Priya and H. Sandhya. "Matrix factorization for recommendation system". In: *International Conference on Artificial Intelligence and Data Engineering*. Springer Nature Singapore, 2019, pp. 267–280.

[45] Chengcai Leng et al. "Graph regularized Lp smooth non-negative matrix factorization for data representation". In: *IEEE/CAA Journal of Automatica Sinica* 6.2 (2019), pp. 584–595.

[46] Mingming Li et al. "Nonnegative matrix factorization with Wasserstein metric-based regularization for enhanced text embedding". In: *PloS one* 19.12 (2024), e0314762.

[47] Tao Li and Cha-charis Ding. "Nonnegative matrix factorizations for clustering: A survey". In: *Data Clustering* (2018), pp. 149–176.

[48] Zhao Li, Xindong Wu, and Hong Peng. "Nonnegative matrix factorization on orthogonal subspace". In: *Pattern Recognition Letters* 31.9 (2010), pp. 905–911.

[49] Qing Liao and Qian Zhang. "Local coordinate based graph-regularized NMF for image representation". In: *Signal processing* 124 (2016), pp. 103–114.

[50] Y. Lin et al. "Meta Matrix Factorization for Federated Rating Predictions". In: *Proceedings of the IEEE Conference*. 2020, pp. 981–990.

[51] X. Luo et al. "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems". In: *IEEE Transactions on Industrial Informatics* 10.2 (2014), pp. 1273–1284.

[52] B. M'sik and B. M. Casablanca. "Topic modeling coherence: A comparative study between LDA and NMF models using COVID'19 corpus". In: *International Journal* 9.4 (2020).

[53] Ondřej Mokrý et al. "Algorithms for audio inpainting based on probabilistic nonnegative matrix factorization". In: *Signal Processing* 206 (2023), p. 108905.

[54] Nicolas Nadisic, Nicolas Gillis, and Christophe Kervazo. "Smoothed separable nonnegative matrix factorization". In: *Linear Algebra and its Applications* 676 (2023), pp. 174–204.

[55] Salifu Nanga et al. "Review of dimension reduction methods". In: *Journal of Data Analysis and Information Processing* 9.3 (2021), pp. 189–231.

[56] Melanie Nijs et al. "A mathematical comparison of non-negative matrix factorization related methods with practical implications for the analysis of mass spectrometry imaging data". In: *Rapid Communications in Mass Spectrometry* 35.21 (2021), e9181.

[57] A. Pascual-Montano et al. "bioNMF: a versatile tool for non-negative matrix factorization in biology". In: *BMC Bioinformatics* 7 (2006), pp. 1–9.

[58] Kasun Pathirage et al. "Interpretable dimensionality reduction and classification of mass spectrometry imaging data in a visceral pain model via non-negative matrix factorization". In: *Plos one* 19.10 (2024), e0300526.

[59] Robert Peharz and Franz Pernkopf. "Sparse nonnegative matrix factorization with 0-constraints". In: *Neurocomputing* 80 (2012), pp. 38–46.

[60] J. Qiang et al. "Clustering Sparse Data with Feature Correlation with Application to Discover Subtypes in Cancer". In: *IEEE Access* 8 (2020), pp. 67775–67789.

[61] Jérémy Rapin et al. "NMF with sparse regularizations in transformed domains". In: *SIAM journal on Imaging Sciences* 7.4 (2014), pp. 2020–2047.

[62] Nadia Rashid et al. "ASAD: A Meta Learning-Based Auto-Selective Approach and Tool for Anomaly Detection". In: *IEEE Access* (2024).

[63] Israr ur Rehman et al. "CAML: Contextual augmented meta-learning for cold-start recommendation". In: *Neurocomputing* 533 (2023), pp. 178–190.

[64] F. Saberi-Movahed et al. "Nonnegative Matrix Factorization in Dimensionality Reduction: A Survey". In: *arXiv preprint arXiv:2405.03615* (2024).

[65] Y.E. Salehani et al. "MSdB-NMF: MultiSpectral Document Image Binarization Framework via Non-Negative Matrix Factorization Approach". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 9099–9112.

[66] Yaser Esmaeili Salehani and Saeed Gazor. "Smooth and sparse regularization for NMF hyperspectral unmixing". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10.8 (2017), pp. 3677–3692.

[67] M.D. Sweeney et al. "FaStaNMF: A Fast and Stable Non-Negative Matrix Factorization for Gene Expression". In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2023).

[68] Y. Tian, X. Zhao, and W. Huang. "Meta-Learning Approaches for Learning-to-Learn in Deep Learning: A Survey". In: *Neurocomputing* 494 (2022), pp. 203–223.

[69] Laurens Van Der Maaten, Eric O Postma, H Jaap Van Den Herik, et al. "Dimensionality reduction: A comparative review". In: *Journal of machine learning research* 10.66-71 (2009), p. 13.

[70] Mikolaj Wojnicki. "Bayesian Non-Negative Matrix Factorization with Applications in Genetics". In: (2024).

[71] Han-Jia Ye, Lu Han, and De-Chuan Zhan. "Revisiting unsupervised meta-learning via the characteristics of few-shot tasks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.3 (2022), pp. 3721–3737.

[72] Hongjing Zhang, Tianyang Zhan, and Ian Davidson. "A self-supervised deep learning framework for unsupervised few-shot learning and clustering". In: *Pattern Recognition Letters* 148 (2021), pp. 75–81.

[73] Lin Zhang et al. "Adaptive graph regularized nonnegative matrix factorization for data representation". In: *Applied Intelligence* 50 (2020), pp. 438–447.

[74] Yan Zhang et al. "Unsupervised nonnegative adaptive feature extraction for data representation". In: *IEEE Transactions on Knowledge and Data Engineering* 31.12 (2018), pp. 2423–2440.

[75] Yue Zhao, Ryan A Rossi, and Leman Akoglu. "Automating outlier detection via meta-learning". In: *arXiv preprint arXiv:2009.10606* (2020).

[76] Guoxu Zhou et al. "Nonnegative matrix and tensor factorizations: An algorithmic perspective". In: *IEEE Signal Processing Magazine* 31.3 (2014), pp. 54–65.

[77] Andrei Zinovyev et al. "Blind source separation methods for deconvolution of complex signals in cancer biology". In: *Biochemical and biophysical research communications* 430.3 (2013), pp. 1182–1187.