Dissertation

Nonnegative Matrix Factorization
Theory, Algorithms and Applications

by
Pascal Fernsel

A dissertation submitted in fulfillment of the requirements for the degree of
— Dr. rer. nat. —

Supervisor: Prof. Dr. Dr. h.c. Peter Maaß
Second Reviewer: Prof. Dr. Simon Arridge
Submission Date: January 2022
Date of Defense: 07.03.2022

Center for Industrial Mathematics
University of Bremen
To Micky — A good friend.
Abstract

This work is devoted to the application-oriented design of Nonnegative Matrix Factorization (NMF) models along with the development of solution algorithms, their application to real-world problems and the analysis of some theoretical aspects. The thesis is divided into two main parts and additionally contains a short introduction in the beginning, which includes a description of the thesis, the featured research papers and the contributions of the author of this thesis.

The first part of this work contains preliminary information on basic concepts of inverse problems and NMF as well as a short overview of solution strategies for NMF and the considered applications. The second part contains the five research papers to which the author contributed.

The first two papers have a rather theoretical focus on the considered NMF models. The first survey article gives an overview on construction principles of multiplicative algorithms for generalized NMF models by considering a large variety of penalty terms, while the second paper focuses on regularized orthogonal NMF models and their relation to generalized $K$-means clustering approaches.

The remaining three research articles are mostly application-oriented. The first of these proposes a joint reconstruction and low-rank decomposition method based on tailored NMF models for the general application to dynamic inverse problems. The considered methods are applied to the problem of dynamic computed tomography. The second one introduces spatially coherent clustering approaches based on Total Variation (TV) regularized orthogonal NMF models with a focus on datasets coming from hyperspectral imaging applications. The proposed frameworks are applied and evaluated on a Matrix-Assisted Laser Desorption/Ionization (MALDI) imaging dataset. The last paper analyzes supervised NMF models as a prior feature extraction method for classification tasks, which are specifically designed for mass spectrometry imaging datasets in clinical applications. The considered approaches are applied to MALDI imaging data for the extraction of tumor specific spectral patterns and the classification of tumor types.
Zusammenfassung


Der erste Teil des Werkes beinhaltet Informationen zu den grundlegenden Konzepten von Inversen Problemen und der NMF sowie eine kurze Übersicht zu Lösungsstrategien für die NMF und den betrachteten Anwendungsgebieten. Der zweite Teil umfasst die fünf Forschungsartikel, zu denen der Autor maßgeblich beitrug.


Acknowledgements

First, I would like to thank my supervisor Peter Maaß for his continuous support during my PhD, for the possibility to be part of his working group for such a long time as well as for the numerous research trips, which would not have been possible without his help. His ideas and scientific advices are a central part of this work.

Furthermore, I am very grateful to Simon Arridge and Andreas Hauptmann for giving me the possibility to do a research stay at the University College London, for their kindness during my stay and for our joint work with all the fruitful discussions on different topics. Even though my uncertainties and the fact that I did not know what we were going to do, we managed to be quite productive during my stay.

Moreover, I really much appreciated the secretaries from our working group Dörte and Judith for all their helpful assistance on technical as well as organizational matters.

Apart from that, I would like to thank all my friends in Bremen who were constantly supportive during my time as a PhD student. I am very grateful for my colleagues and friends Alex and Kosta with our numerous discussions and gaming nights. Furthermore, I would especially like to mention Robert for always giving me positive energy and for our nice online gaming sessions. Even though the fact that we have not found much time together during the last years, for which I am very sorry, we never have lost any affection to each other. Finally, I want to thank Artur for his constant mental support and our countless funny “discussions”. I am glad that you belong to one of my closest friends.

Above all, I would like to thank Miriam for her endless love and for constantly being there for me even during the hardest times. Without you, I would not have made it.

Finally, I would like to thank my mother Christine and my brother Marc for their love and support during all this time.
## Contents

Abstract ................................................................. iii  
Zusammenfassung ...................................................... iv  
Acknowledgements ..................................................... v  

<table>
<thead>
<tr>
<th>Contents</th>
<th>vii</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1. Structure of the Thesis</td>
<td>4</td>
</tr>
<tr>
<td>1.2. Research Papers and Contributions of the Author</td>
<td>5</td>
</tr>
<tr>
<td>1.3. Notation</td>
<td>9</td>
</tr>
<tr>
<td>I. Preliminaries</td>
<td>11</td>
</tr>
<tr>
<td>2. Inverse Problems and Nonnegative Matrix Factorization</td>
<td>13</td>
</tr>
<tr>
<td>2.1. Inverse Problems</td>
<td>13</td>
</tr>
<tr>
<td>2.2. Nonnegative Matrix Factorization</td>
<td>20</td>
</tr>
<tr>
<td>2.2.1. Variational Formulation</td>
<td>22</td>
</tr>
<tr>
<td>2.2.2. Orthogonal NMF and Clustering</td>
<td>26</td>
</tr>
<tr>
<td>3. Solution Strategies for NMF</td>
<td>29</td>
</tr>
<tr>
<td>3.1. Overview</td>
<td>29</td>
</tr>
<tr>
<td>3.2. Iterative Algorithms</td>
<td>30</td>
</tr>
<tr>
<td>3.2.1. Multiplicative Algorithms</td>
<td>31</td>
</tr>
<tr>
<td>3.2.2. Proximal Alternating Linearized Minimization</td>
<td>38</td>
</tr>
<tr>
<td>3.2.3. Initialization Methods and Stopping Criteria</td>
<td>47</td>
</tr>
<tr>
<td>4. Applications</td>
<td>49</td>
</tr>
<tr>
<td>4.1. MALDI Imaging</td>
<td>49</td>
</tr>
<tr>
<td>4.2. Dynamic Computed Tomography</td>
<td>51</td>
</tr>
<tr>
<td>5. Conclusion and Outlook</td>
<td>55</td>
</tr>
</tbody>
</table>
Bibliography 57

II. Research Papers 63

A Survey on Surrogate Approaches to Nonnegative Matrix Factorization . . 65
Regularized Orthogonal Nonnegative Matrix Factorization and K-means Clus-
tering . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 101
Joint Reconstruction and Low-rank Decomposition for Dynamic Inverse Prob-
lems . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 113
Spatially Coherent Clustering Based on Orthogonal Nonnegative Matrix Fac-
torization . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 155
Supervised Nonnegative Matrix Factorization Methods for MALDI Imaging
Applications . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 187
1 Introduction

“Is perception of the whole based on perception of its parts?”
— Daniel D. Lee and Sebastian Seung, [34]

In today’s digital world, data mining, data analysis and signal processing are ubiquitous throughout science and engineering and have already become part of our daily routine. It starts with gathering and analyzing datasets from measurements and observations for scientific purposes and ends with oftentimes questionable data collections of user information and customer behaviour in mobile phones, search engines, social media platforms and online shops from companies around the world. The volume of created, captured, copied and consumed data worldwide is predicted to grow exponentially over time with an estimated amount of 74 zettabyte for the year 2021 [55]. This number is comparable to the estimated number of stars in the observable universe, which is currently estimated to lie roughly within the range of $10^{22}$ to $10^{24}$ [17]. Extracting meaningful information and discovering the structure of these oftentimes high dimensional datasets have become important objectives in science and industry. Furthermore, datasets coming from complex phenomena oftentimes consist of a combination of latent components or factors, which motivate the application of suitable methods to decompose the data and to extract these characteristic features.

One general framework, which includes many classical feature extraction approaches, is the so-called Linear Dimensionality Reduction (LDR) technique. For some given data, this approach tries to approximate each data sample based on a linear combination of a few extracted basis elements. Mathematically, this can be formulated as follows: For a given dataset \( \{x_m \in \mathbb{R}^N \mid m = 1, \ldots, M\} \), find \( K \ll \min\{M, N\} \) basis vectors \( \{v_k \in \mathbb{R}^N \mid k = 1, \ldots, K\} \) and corresponding weights \( \{u_{mk} \in \mathbb{R} \mid m = 1, \ldots, M, \, k = 1, \ldots, K\} \), so that each data sample \( x_m \) can be approximated as

\[
x_m \approx \sum_{k=1}^{K} u_{mk} v_k.
\] (1.1)
1. Introduction

This is illustrated in Figure 1.1 for $N = 3$, $K = 2$ and $M = 72$.

To obtain a more compact formulation of the LDR framework, we can write the data samples $x_m$ and the basis vectors $v_k$ row-wise into a data matrix $X \in \mathbb{R}^{M \times N}$ and a basis matrix $V \in \mathbb{R}^{K \times N}$ respectively, so that $X := [x_1, \ldots, x_M]^\top$ and $V := [v_1, \ldots, v_K]^\top$. Furthermore, we define the coefficient matrix $U \in \mathbb{R}^{M \times K}$ to be $U_{mk} := u_{mk}$. In this way, (1.1) can be written as

$$X \approx UV.$$  

(1.2)

Hence, LDR is equivalent to Low-Rank Matrix Approximation (LRMA).

Usually, it is meaningful to approximate the data samples by just a small number of basis vectors, so that $K$ is chosen much smaller than the dimensions $M$ and $N$. Hence, LRMA can also be used for compressing data, since the data matrix $X$ contains $M \cdot N$ entries, whereas the factorization $UV$ only contains $K \cdot (M + N)$ entries. Furthermore, since each data sample can be reconstructed just by a small number of basis vectors, the extracted basis vectors $v_k$ can be seen as characteristic components of the given dataset.

To actually solve the LRMA problem in (1.2), the usual procedure is to introduce a suitable error measure $D(\cdot, \cdot)$ (see also Section 2.2.1). It is worth noting that for the standard Frobenius norm as discrepancy term $D(X, UV) := \|X - UV\|_F^2 =$
\[
\sum_{mn}(X_{mn} - (UV)_{mn})^2,
\]
it can be shown that LRMA is equivalent to the well-known Principal Component Analysis (PCA) \cite{30,23}, which can be computed based on the Singular Value Decomposition (SVD) of \(X\).

Even though the fact that PCA has been known for more than 100 years, LRMA models gained momentum within the last two decades due to the massive growth of global data acquisition and storage as well as the fact, that high dimensional data can be oftentimes well approximated by matrices with a low rank. LRMA approaches are analyzed and applied in many different research fields including numerical linear algebra, machine learning, data analysis, image processing, chemistry and meteorology, where they are used for example to solve linear systems of equations or to perform regression, clustering, classification, feature extraction and denoising (see \cite{10,23} and the references therein).

Moreover, it is oftentimes useful to include additional constraints to the factorization matrices in (1.2). In the case of a clustering method, it is typical to enforce the matrix \(U\) to have a single non-zero entry in each of its rows to ensure the clustering interpretability of the obtained matrix factorization, so that the rows of \(V\) can be interpreted as the centroids of the classical \(K\)-means problem (see also Section 2.2.2). Other examples are sparse PCA or sparse component analysis, where the rows of \(U\) are constrained to contain as many zeros as possible (sparsity) in order to ensure that each data sample is approximated by just a few of the extracted basis vectors.

However, this work is devoted to the specific case of the so-called Nonnegative Matrix Factorization (NMF), which also belongs to the LRMA framework and can be seen as an additive linear model for nonnegative datasets. In its classical form, NMF constrains the entries of the factorization matrices \(U\) and \(V\) to be nonnegative and is typically only used for nonnegative data matrices. Different from the above techniques, these constraints allow to approximate the data samples by a nonnegative superposition of the basis vectors and hence allow a parts-based representation of the whole dataset based on just a few basis vectors. Furthermore, the nonnegativity constraint simplifies the interpretation of the matrices \(U\) and \(V\) for nonnegative data. NMF was introduced by Paatero and Tapper in 1994 and gained much popularity by the papers of Lee and Seung \cite{34,35} with the introduction of the classical multiplicative algorithms (see Section 3.2.1). A more thorough introduction to the NMF gives Section 2.2.

This thesis focuses on three major aspects of the NMF:

- The design of new application-oriented NMF models together with some theoretical studies on their relation to other methods
• The development of suitable solution algorithms along with several convergence results

• The final application of the constructed NMF models to real-world problems

For a more detailed overview of the research papers and the contributions of the author, we refer the reader to Section 1.2. In the following section, we give a short overview on the structure of the present thesis.

1.1. Structure of the Thesis

This thesis is divided into two main parts, namely the Preliminaries (Part I) and the Research Papers (Part II).

In the preliminaries, we lay the foundations and describe the fundamental prerequisites for the presented research papers in Part II, which include a basic understanding of inverse problems and nonnegative matrix factorization along with an overview on the theory and solution techniques of NMF as well as a description of the considered applications.

In Chapter 2, we describe shortly some basic principles of the classical theory of linear inverse problems as well as dynamic inverse problems (see Section 2.1). This will help us to get a better understanding of the needed regularization procedures for NMF problems and of the application case of dynamic computed tomography in Section 4.2. Section 2.2 begins with a general introduction to the NMF, short discussions on rather theoretical topics like uniqueness and ill-posedness as well as a description of the typical variational formulation of the NMF problem in Section 2.2.1. Lastly, Section 2.2.2 describes the special case of orthogonal NMF and its relation to clustering problems.

Since the majority of the presented research papers in Part II have a major focus on the algorithm development for the different application cases, Chapter 3 is completely devoted to general solution methods with a focus on iterative algorithms for NMF problems. After obtaining an overview on possible solution approaches in Section 3.1, Section 3.2 describes common iterative optimization techniques including the classical optimization method based on multiplicative algorithms (Section 3.2.1) as well as so-called proximal alternating linearized minimization schemes (Section 3.2.2). Finally, Section 3.2.3 gives a short overview on initialization methods and stopping criteria for NMF.

Most of the NMF models discussed in the research papers in Part II are designed for specific application cases. Chapter 4 is devoted to the considered application fields of
this work and gives a short introduction to MALDI imaging (Section 4.1) and dynamic computed tomography (Section 4.2).

Finally, Chapter 5 gives an outlook for future possible research directions.

1.2. Research Papers and Contributions of the Author

In this section, we give a short overview and description of the featured research papers in this work. From a total of six papers that have been published in a journal or uploaded as a preprint with me being at least a co-author (see [20, 21, 5, 3, 18, 36]), only five of them will be included in this thesis, since [5] deals about the analysis of the invertibility of deep neural networks, which is not the focus of this work.

In the following, a list of these five research papers is presented together with the contributions of the author ordered by the appearance in this thesis. Furthermore, the publication status and a short topic description for each research paper is provided. Four of the featured research papers are published in journals [20, 3, 18, 36], whereas [21] is uploaded as a preprint in arXiv.

<table>
<thead>
<tr>
<th>A Survey on Surrogate Approaches to NMF [20]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Contribution</strong></td>
</tr>
<tr>
<td>This paper is based on my master thesis. Besides contributing to all aspects of the paper, I also performed the presented numerical experiments.</td>
</tr>
<tr>
<td><strong>Publication Status and Availability</strong></td>
</tr>
<tr>
<td>Published as open access in the Vietnam Journal of Mathematics.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>In this survey, the so-called Majorization-Minimization (MM) principle to derive multiplicative algorithms for solving NMF problems is reviewed. Furthermore, the known construction principles of the corresponding surrogate functions is extended by a general framework, which allows to add a large variety of penalty terms to the NMF cost function. Finally, generalized alternating multiplicative algorithms are derived and applied to MALDI imaging data. While this paper already contains introductory material to this topic, we also...</td>
</tr>
</tbody>
</table>
1. Introduction

refer the reader particularly to Section 2.2, 3.2.1 as well as Section 4.1 of this thesis.

---

### Regularized ONMF and $K$-means Clustering [21]


**Contribution**

The original idea of the proposed framework how to link orthogonal nonnegative matrix factorization to $K$-means clustering came from the co-author Peter Maaß, who also derived the first theoretical results. However, the finalization and the theoretical extensions as presented in [21] were done by me.

**Publication Status and Availability**

Uploaded in arXiv as a preprint.

**Description**

This paper focuses on relationships between $K$-means clustering approaches and regularized Orthogonal Nonnegative Matrix Factorization (ONMF) models. In particular, this work proposes a novel framework to extract the distance measure and the centroid of the $K$-means method based on first order conditions of the considered ONMF objective function. The proposed derivation technique exploits the typical alternating minimization schemes of NMF algorithms and generalizes to non-standard regularized ONMF models. The framework is applied on ONMF models with $\ell_1$ and $\ell_2$ discrepancy terms with an additional elastic net regularization on both of the factorization matrices. Finally, an intuitive view of the obtained results is given and special cases are examined and compared to the findings described in the literature.

For introductory material to this paper, we refer the reader particularly to Section 2.2 and 2.2.2 of this thesis.

---

### Joint Rec. and Low-Rank Decomposition for Dynamic IP [3]

### Contribution

The idea of this paper was evolved from discussions with Simon Arridge and Andreas Hauptmann during my research stay at the University College London from 16 February 2019 to 22 March 2019 and from 28 August 2019 to 06 September 2019. During my stay, we finalized the main idea of the paper as well as the development of suitable NMF models for solving dynamic inverse problems and chose the problem of dynamic computed tomography as an appropriate application case. In the subsequent part of this project, I did the major work on adapting the theoretical foundations, the algorithm development as well as the numerical experiments.

### Publication Status and Availability

Published as open access in Inverse Problems & Imaging. The used MATLAB® codes and datasets are freely available in the GitLab [4].

### Description

In this work, a joint reconstruction and low-rank decomposition method based on nonnegative matrix factorization for dynamic inverse problems is introduced. This approach is especially designed for the case, where the target can be represented by a decomposition of spatial and temporal basis functions and hence can be efficiently described by a low-rank decomposition. The proposed framework allows a flexible incorporation of separate regularisers for spatial and temporal features and is able to provide a substantial speed-up for the stationary case. Multiplicative update rules are explicitly derived for different NMF models along with theoretical guarantees of the monotone decrease of the cost function. Finally, the proposed framework is applied to the specific problem of dynamic computed tomography and evaluated on three different simulated phantoms, where the combined approach is compared to a separated low-rank reconstruction and decomposition approach based on the widely used principal component analysis. For introductory material, we refer the reader particularly to Section 2.1, 2.2, 3.2.1 as well as Section 4.2.
1. Introduction

Spatially Coherent Clustering Based on Orthogonal NMF [18]


Contribution

I contributed to all aspects of the paper.

Publication Status and Availability

Published as open access in the Journal of Imaging. The used MATLAB® codes are freely available in the GitLab [19].

Description

Many application fields like mass spectrometry imaging or Earth remote sensing naturally lead to datasets with spatially coherent areas. In these cases, classical approaches in cluster analysis, which do not take into account the additional spatial information, typically do not lead to satisfactory reconstructions of the spatially coherent regions. This work introduces clustering models based on Orthogonal Nonnegative Matrix Factorization (ONMF), which include an additional Total Variation (TV) regularization term for the cluster membership matrix to induce the needed spatial coherence in the clusters. The obtained ONMF models are specifically designed for datasets coming from hyperspectral imaging applications but can also be applied to general nonnegative data, which contain additional spatial information. Several ONMF models with different optimization techniques are proposed, where the TV regularization is either performed as a subsequent postprocessing step or included into the clustering algorithm. A short convergence analysis of the proposed combined methods is provided. Finally, a numerical evaluation of 12 different TV regularized ONMF methods on a MALDI imaging measurement is provided, which leads to significantly better clustering results compared to classical clustering models.

For introductory material, we refer the reader particularly to Section 2.2, 2.2.2, 3.2.1, 3.2.2 as well as Section 4.1.

Supervised NMF Methods for MALDI Imaging Applications [36]

Johannes Leuschner, Maximilian Schmidt, Pascal Fernsel, Delf Lachmund, Tobias Boskamp and Peter Maaß: Supervised non-negative matrix factorization
1.3. Notation

Matrices play a major role in this work and are denoted, unless otherwise stated, by capital bold Greek or Latin letters (e.g. $X, U, V, \Psi, A, \Phi, X_i, \Psi_i, \ldots$). Moreover, a description of the rows and columns of matrices is needed. To do so, we write $X_{i,\bullet}$ and $X_{\bullet,j}$ for the $i$-th row and $j$-th column of a matrix $X$. The entries in the $i$-th row

---

**Contribution**

The theoretical part of this paper is based on my previous work [20] and includes in particular the results on the algorithm development of the considered NMF models. However, I was not involved in the numerical experiments of this work, which are due to Johannes Leuschner and Maximilian Schmidt.

**Publication Status and Availability**

Published as open access in Bioinformatics. The used MATLAB® codes are freely available in the GitLab [37].

**Description**

In this work, a supervised NMF workflow is introduced, which is able to perform a combined feature extraction and classification task. This is done by incorporating the classification labels into the NMF and adding appropriate supervised penalty terms into the NMF cost function in order to guide the NMF algorithms towards the extraction of data patterns, which are relevant for discriminating the respective classes. The considered NMF models are specifically designed for mass spectrometry imaging datasets with a focus on clinical applications such as tumor typing and the extraction of tumor specific spectral patterns but can be also used for general nonnegative data. Different NMF models are analyzed and multiplicative algorithms as well as ADADELTA update rules are derived. Finally, numerical experiments on a MALDI imaging dataset are provided and confirm, that the proposed supervised NMF methods lead to significantly better classification accuracy and stability as compared to other standard approaches. For introductory material, we refer the reader particularly to Section 2.2, 3.2.1 as well as Section 4.1.
and $j$-th column of $X$ and of a matrix product $UV$ are indicated as $X_{ij}$ and $(UV)_{ij}$ respectively. Regarding iterative algorithms, we denote the $i$-th iteration of a matrix $X$ by $X^{[i]}$. Vectors in $\mathbb{R}^n$ will be denoted, unless otherwise states, by bold small Greek or Latin letters (e.g. $\mathbf{x, u, v, \theta, x_i, \theta_i, \ldots}$).

Furthermore, norms of matrices and vectors are used throughout this work. The Frobenius norm of a matrix $X$ and the Euclidean norm of the column vector $X_{\bullet,j}$ are indicated as $\|X\|_F$ and $\|X_{\bullet,j}\|_2$ respectively. Moreover, we define the 1-norm of a matrix $X$ to be $\|X\|_1 := \sum_{i,j} |X_{ij}|$. What is more, the inner product and the norm of a Hilbert space $\mathcal{H}$ is denoted as $\langle x, y \rangle_{\mathcal{H}}$ and $\|x\|_{\mathcal{H}}$ respectively for $x, y \in \mathcal{H}$. In the case of the standard inner product in $\mathbb{R}^n$, it is written as $\langle x, y \rangle$.

Finally, nonnegativity constraints on the matrices are commonly used in this work. We use the shorthand notation $X \geq 0$ to indicate that all entries of $X$ are nonnegative. Furthermore, we define the set $\mathbb{R}_{\geq 0} := \{ x \in \mathbb{R} \mid x \geq 0 \}$. The notation of the NMF problem introduced in Section 2.2 is reused throughout Part I of this work.
Part I.

Preliminaries
2 | Inverse Problems and Nonnegative Matrix Factorization

In this chapter, we first give a short introduction to the basic concepts of the classical theory of linear inverse problems in Section 2.1. This part includes the definition of ill-posed problems along with the introduction of the mathematical framework and some basics on regularization theory with a focus on Tikhonov regularization. The latter will be used extensively regarding the variational formulation of the considered NMF models in this work. Finally, a short treatise on the theory of dynamic inverse problems is included, since the featured article [3] deals about the specific application case of dynamic computed tomography, which is further described in Section 4.2. However, we will keep Section 2.1 rather short since the theory of inverse problems is not the focus of this work.

The following Section 2.2 introduces the problem of nonnegative matrix factorization and discusses some basic properties. Section 2.2.1 describes the typical variational formulation of the NMF as a minimization problem and gives an overview of the considered regularization terms in this work along with their use cases.

Finally, Section 2.2.2 introduces shortly the special case of Orthogonal NMF along with its relation to clustering methods to ensure a basic understanding of this type of NMF problems, which are treated in four of the featured articles [21, 18, 36, 20].

2.1. Inverse Problems

In many real-world applications, direct measurements of specific properties of an object are technically not possible and have to be determined based on indirect observations. This problem, i.e. the determination of the cause of an observed effect, is referred to as an inverse problem. To determine the effect with the knowledge of the cause is referred to as the direct problem and is typically much easier to handle.

Inverse problems occur in many different application fields such as medical imaging with classical examples such as Electrical Impedance Tomography (EIT) or Computed
Tomography (CT), image processing problems (e.g. image denoising) or engineering with the identification of parameters in PDEs. Further classical examples of inverse problems are inverse scattering problems, where the characteristics of an object have to be determined based on the scattering of incoming waves, as well as several types of inverse problems of the heat equation, where, for example, the heat distribution at time \( t < T \) must be computed based on the heat distribution at time \( T \). In this work, we will take a closer look at the specific problem of dynamic CT, which is described in more detail in Section 4.2.

Linear inverse problems are typically written as an operator equation

\[ Ax^\dagger = y^\dagger, \tag{2.1} \]

where \( A : \mathcal{X} \to \mathcal{Y} \) is a linear and bounded operator modeling the considered problem with \( \mathcal{X} \) and \( \mathcal{Y} \) being Hilbert spaces, \( x^\dagger \in \mathcal{X} \) the true solution of the inverse problem and \( y^\dagger \in \mathcal{R}(A) := \{ Ax \mid x \in \mathcal{X} \} \subset \mathcal{Y} \) the true data, where \( \mathcal{R}(A) \) denotes the range or image of \( A \). For simplicity, we consider in this work only Hilbert spaces and note that the theory of inverse problems can also be extended to the case of Banach spaces [53].

In most cases, it is only possible to obtain noisy measurements \( y^\delta \in \mathcal{Y} \) with \( \| y^\dagger - y^\delta \|_{\mathcal{Y}} < \delta \) for some \( \delta > 0 \). Based on these noisy measurements, the aim is to find a suitable approximation \( x^\delta \in \mathcal{X} \) of the true solution \( x^\dagger \). However, it could be the case that \( y^\delta \not\in \mathcal{R}(A) \) if \( A \) is not surjective or that the nullspace \( \mathcal{N}(A) := \{ x \in \mathcal{X} \mid Ax = 0 \} \) is nonempty leading to non-unique solutions of (2.1). Furthermore, it is possible that the solution does not depend continuously on the measured data. This property is especially critical and makes inverse problems hard to solve, since even small noise in the measurements \( y^\delta \) can lead to huge reconstruction errors of \( x^\delta \). These properties lead to the notion of well-posedness in the sense of Hadamard for linear problems, which he introduced in 1923 [25].

**Definition 2.1** (Well-Posedness (Hadamard, [25])). Let \( A : \mathcal{X} \to \mathcal{Y} \) be a given linear and continuous mapping with Hilbert spaces \( \mathcal{X} \) and \( \mathcal{Y} \). The linear problem \( (A, \mathcal{X}, \mathcal{Y}) \) given by the operator equation \( Ax = y \) is well-posed, if the following conditions are satisfied:

(i) For all \( y \in \mathcal{Y} \), there exists a solution \( x \in \mathcal{X} \).

(ii) The solution is unique.

(iii) The solution \( x \) depends continuously on the data \( y \), i.e. the inverse operator \( A^{-1} : \mathcal{R}(A) \to \mathcal{X} \) is continuous.
Moreover, the problem $(A, X, Y)$ is ill-posed in the sense of Hadamard, if one of the above conditions is not fulfilled.

In 1987, Zuhair Nashed suggested a new notion of well-posedness [42]. This concept also leads to the definition of the so-called Moore-Penrose inverse: For the operator equation $Ax = y$ for a given $y \in Y$, we introduce a generalized solution concept by considering the minimization problem $\min_{x \in X} \| Ax - y \|_Y$. Hence, we look for a $x^+ \in X$ that minimizes the residual, so that $\| Ax^+ - y \|_Y \leq \| Ax - y \|_Y$ for all $x \in X$. It can be shown that $x^+$ minimizes the residual if and only if it also solves the normal equation $A^*Ax = A^*y$ with $L(y) := \{ x \in X \mid A^*Ax = A^*y \}$ being the set of solutions. Furthermore, it can be shown that $L(y)$ is non-empty if and only if $y \in R(A) \oplus R(A)^\perp \subset Y$ and that $L(y)$ contains a unique element $x^+$ with minimal norm. This defines the Moore-Penrose inverse $A^+: D(A^+) \to X$ with $D(A^+) := R(A) \oplus R(A)^\perp \subset Y$, which assigns each $y \in D(A^+)$ to the minimum norm solution $x^+ \in L(y).

Furthermore, it can be shown that $A^+$ is continuous if and only if the range of $A$ is closed: $R(A) = \overline{R(A)}$. If this is the case, $D(A^+) = Y$ and for every $y \in Y$, there is a unique $x^+$, which solves the normal equation and hence minimizes the residual.

However, if the range of $A$ is not closed, we have that $D(A^+) \subsetneq Y$. Hence, the supposed solution $A^+y$ is only defined for $y \in D(A^+)$ and is additionally unsuitable since $A^+$ is not continuous, so that small deviations in the measurements can lead to huge reconstruction errors.

Therefore, the only important condition of the ill-posedness of an inverse problem is the case when the range of $A$ is not closed and leads to the definition of ill-posedness in the sense of Nashed [42].

**Definition 2.2 (Ill-Posedness (Nashed, [42])))**. The problem $(A, X, Y)$ is called ill-posed in the sense of Nashed if the range of $A$ is not closed. Otherwise, it is called well-posed in the sense of Nashed.

Typical operators, which lead to ill-posed inverse problems are compact operators with an infinite dimensional image. A linear operator $A : X \to Y$ is called compact, if for every bounded sequence $\{ x_n \}_{n \in \mathbb{N}} \subset X$, the sequence $\{ Ax_n \}_{n \in \mathbb{N}} \subset Y$ has a cluster point in $Y$. Accordingly, we define the set $K(X, Y) := \{ A : X \to Y \mid A \text{ is linear and compact} \}$ as well as $L(X, Y) := \{ A : X \to Y \mid A \text{ is linear and bounded} \}$. It can be shown that every linear compact operator is also bounded, i.e. it holds that $K(X, Y) \subset L(X, Y)$.

A classical example of a compact operator are integral operators $A : L^2(\Omega) \to L^2(\Omega)$.
of the form

\[ Af(x) := \int_\Omega k(x, y)f(y) \, dy \]

with \( \Omega \subset \mathbb{R}^n \) being a nonempty and compact set, \( L^2(\Omega) \) being the usual Hilbert space of square-integrable functions and with the kernel \( k \in L^2(\Omega \times \Omega) \). Another famous example is the Radon transform \( R : L^2(\Omega) \to L^2(\Gamma) \) with \( \Omega := \{ x \in \mathbb{R}^2 \mid x_1^2 + x_2^2 \leq 1 \} \) and \( \Gamma := [0, 2\pi] \times [-1, 1] \), which is defined by

\[ Rf(\varphi, s) := \int_{\langle x, \theta(\varphi) \rangle = s} f(x) \, d\sigma(x) \]

with \( \theta(\varphi) := [\cos(\varphi), \sin(\varphi)]^\top \) and leads to the inverse problem of computed tomography. A more detailed description of this problem is given in Section 4.2.

The ill-posedness of compact operators can be directly seen by analyzing their Singular Value Decomposition (SVD). It can be shown that for each \( A \in \mathcal{K}(\mathcal{X}, \mathcal{Y}) \), there exist singular values \( \{\sigma_j\}_{j \in J} \subset (0, \infty) \) with \( \sigma_1 \geq \sigma_2 \geq \cdots > 0 \) and orthonormal bases \( \{u_j\}_{j \in J} \subset \mathcal{X} \) and \( \{v_j\}_{j \in J} \subset \mathcal{Y} \) of \( \mathcal{N}(A)^\perp \) and \( \mathcal{R}(A) \) respectively with

\[ \text{span}\{u_j \mid j \in J\} = \mathcal{N}(A)^\perp, \quad \text{span}\{v_j \mid j \in J\} = \mathcal{R}(A), \]

so that \( Ax \) can be written as the series expansion

\[ Ax = \sum_{j \in J} \sigma_j \langle x, u_j \rangle \mathcal{X} v_j \]

for all \( x \in \mathcal{X} \) and some index set \( J \). With this, it can be shown that the Moore-Penrose inverse has the form

\[ A^+ y = \sum_{j \in J} \sigma_j^{-1} \langle y, v_j \rangle \mathcal{Y} u_j \]  \hspace{1cm} (2.2)

for \( y \in \mathcal{D}(A^+) \). If the image of \( A \) is finite-dimensional, then the index set \( J \) is finite and \( A^+ \) is continuous leading to well-posed problems; otherwise, \( A^+ \) is not continuous due to \( \sigma_j \to 0 \) for \( j \to \infty \), which is the reason for the ill-posedness of compact operators with infinite-dimensional image. Furthermore, the degree of the ill-posedness can be measured by how fast the singular values converge to zero.

In the case of ill-posed inverse problems, it is hence necessary to transform them into well-posed problems in order to compute reasonable solutions. This is typically done by a suitable approximation of the Moore-Penrose inverse with a family of continuous operators \( \{R_t\}_{t>0} \) and leads to the so-called regularization theory of inverse problems.
Definition 2.3 (Regularization). For $A \in \mathcal{L}(\mathcal{X}, \mathcal{Y})$, let $\{R_t\}_{t>0}$ be a family of continuous operators from $\mathcal{Y}$ to $\mathcal{X}$ with $R_t(0) = 0$. If there exists a map $\gamma : (0, \infty) \times \mathcal{Y} \rightarrow (0, \infty)$, so that for every $y \in \mathcal{R}(A)$ it holds
\[
\lim_{\delta \to 0} \sup_{\|y^\delta\|_{\mathcal{Y}} \leq \delta} \|A^+y - R_{\gamma(\delta,y^\delta)}y^\delta\|_{\mathcal{X}} = 0,
\]
then the pair $\{\{R_t\}_{t>0}, \gamma\}$ is called a regularization for $A^+$. The map $\gamma$ is called parameter choice and fulfills
\[
\lim_{\delta \to 0} \sup_{\|y^\delta\|_{\mathcal{Y}} \leq \delta} \gamma(\delta, y^\delta) = 0.
\]

The regularization parameter $\gamma(\delta, y^\delta)$ depends in general on the noise level $\delta$ and the noisy measurement $y^\delta$ and goes to zero for $\delta \to 0$. Hence, the regularization consists of a family of parameterized operators $\{R_t\}_{t>0}$ with a suitable parameter choice $\gamma$, which try to approximate the Moore-Penrose inverse so that $R_{\gamma(\delta,y^\delta)}y^\delta \to A^+y$ for $\delta, \gamma(\delta, y^\delta) \to 0$ for all $y \in \mathcal{R}(A)$ and $y^\delta \in \mathcal{Y}$ with $\|y - y^\delta\|_{\mathcal{Y}} \leq \delta$.

An upper bound of the overall reconstruction error $\|A^+ y - R_t y^\delta\|_{\mathcal{X}}$ is given by the sum of the approximation error and the data error, so that
\[
\|A^+ y - R_t y^\delta\|_{\mathcal{X}} \leq \underbrace{\|A^+ y - R_t y\|_{\mathcal{X}}}_{\text{Approximation error}} + \underbrace{\|R_t y - R_t y^\delta\|_{\mathcal{X}}}_{\text{Data error}}.
\]

For $t \to 0$, the approximation error tends to zero according to Definition 2.3, whereas the data error tends to infinity, since the operators $R_t$ are not uniformly bounded if the range of $A$ is not closed. Hence, an optimal parameter choice is needed to minimize the overall reconstruction error.

A typical example is the regularization by so-called filter functions, which are motivated by the SVD of the Moore-Penrose inverse in (2.2) and the fact, that $1/\sigma_j \to \infty$ for $j \to \infty$. These filter functions $F_t : (0, \infty) \to [0, \infty]$ try to resolve this problem by defining the corresponding family of operators $R_t : \mathcal{Y} \to \mathcal{X}$ so that
\[
R_t(y) := \sum_{j \in J} F_t(\sigma_j)\sigma_j^{-1}(y, v_j)y v_j.
\]

One typical approach is to discard the summands with the small singular values leading to the so-called truncated Singular Value Decomposition (tSVD). This is done by
defining the filter functions to be

\[ F_t(\sigma) := \begin{cases} 
1 & \text{if } \sigma > t, \\
0 & \text{otherwise}, 
\end{cases} \]

where the regularization parameter \( t \) determines the threshold to truncate the singular values. It can be shown that the resulting family of operators \( \{R_t\}_{t>0} \) indeed leads to a regularization according to Definition 2.3.

Another classical example is the so-called Tikhonov regularization, which is defined by the filter \( F_t(\sigma) := \sigma^2 / (\sigma^2 + t) \). It can be shown, that the corresponding regularization \( R_t(y^\delta) \) as defined in (2.3) can be also obtained by minimizing the objective function

\[ J_t(x) := \|Ax - y^\delta\|_2^2 + t\|x\|_\chi^2, \tag{2.4} \]

leading to \( R_t(y^\delta) = \text{arg min } J_t(x) \), where \( \|Ax - y^\delta\|_2^2 \) is typically referred to as the discrepancy term, which measures the error between the noisy measurement \( y^\delta \) and the true data \( Ax \). The other term \( \|x\|_\chi^2 \) is the regularization term or penalty term along with the regularization parameter \( t \), which denotes in this case the classical \( L^2 \) regularization.

The aspect of the regularization based on a cost function motivates to consider more general regularization terms and leads to the generalized Tikhonov regularization framework based on the minimization of cost functions which have the form

\[ J_t(x) := \|Ax - y^\delta\|_2^2 + tR(x), \tag{2.5} \]

where \( R(x) \) is a suitably chosen regularization term incorporating some prior knowledge about the structure of the solution.

Besides the classical Tikhonov regularization with the \( L^2 \) penalty term, the concept of sparsity in the solution became popular in many applications in the last years. An element \( x \in \chi \) of a separable Hilbert space with a countable orthonormal basis \( \{u_j\}_{j \in J} \) is called sparse with respect to \( \{u_j\}_{j \in J} \), if there exists a finite index set \( I \subset J \), so that \( x \) can be written as \( x = \sum_{i \in I} (x, u_i) \chi u_i \). The authors of the pioneering work in [11] considered regularization terms of the form \( R(x) := \sum_{j \in J} |(x, u_j)|_\chi^q \) for \( 1 \leq q \leq 2 \) and could show that the choice of \( q = 1 \) leads to sparse minimizers of the corresponding objective function.

Another type of regularization, which is also used in many of the featured articles of this thesis, is the so-called Total Variation (TV) regularization. In many imaging
applications, the considered images contain flat regions with sharp edges. As in the case of the classical Tikhonov regularization with the $L^2$ penalty term, the TV approach promotes smooth solutions but also additionally preserves the sharp edges in the image and thus makes a better use of the a priori knowledge. The general definition of the total variation is given for locally integrable functions $u \in L^1_{\text{loc}}(\Omega)$ for an open and bounded set $\Omega \subset \mathbb{R}^n$ and is defined as

$$\text{TV}(u) := \sup \left\{ -\int_{\Omega} u \, \text{div} \, \phi \, dx : \phi \in C_c^\infty(\Omega, \mathbb{R}^n) \text{ with } \|\phi(x)\|_2 \leq 1 \text{ } \forall \, x \in \Omega \right\},$$

where $C_c^\infty(\Omega, \mathbb{R}^n)$ is the set of all smooth functions with compact support. However, the much more typical form of the TV penalty term can be obtained for Sobolev functions $u \in W^{1,1}(\Omega)$ and is given by

$$\text{TV}(u) = \int_{\Omega} \|\nabla u(x)\|_2 \, dx.$$

TV regularization methods are primarily used for image denoising, deblurring and zooming (see [9] and the references therein).

These and more regularization terms are used in the context of the application of the nonnegative matrix factorization throughout the research papers in Part II of this thesis. An overview of the used penalty terms is given in Table 2.1 of Section 2.2.1. In this context, the sparsity and $L^2$ regularization will be denoted with the sequence spaces $\ell_1$ and $\ell_2$ respectively. For more information on other techniques to tackle the ill-posedness of inverse problems including iterative regularizations or regularization methods based on discretization, we refer the reader to [49].

Finally, we would like to introduce very shortly the extended concept of dynamic inverse problems, since one of the featured papers [3] consider the application of different nonnegative matrix factorization models for dynamic computed tomography. Different from the classical framework, dynamic inverse problems assume that the measurement process takes a certain amount of time and allows the forward model as well as the examined object, which needs to be reconstructed, to change over time. Hence, this can be described by linear operators $A_i$ with $i \in \mathbb{N}$ specifying the current time step $t_i$ of the performed measurement, which map the solutions $x_i^\dagger$ to the true data $y_i^\dagger$ (see also Equation (2.1)). However, as in the case of classical inverse problems, only noisy measurements $y_i^\delta$ are available. A typical mathematical framework (see [51]) is the consideration of linear and bounded operators $A_i \in \mathcal{L}(\mathcal{H}, \mathcal{G}_i)$ for Hilbert spaces $\mathcal{H}$ and
\( \mathcal{G}_i \) for all \( i \) leading to the dynamic problem

\[
A_i x_i = y_i \quad \text{for all } i.
\]

Typical further examples of dynamic inverse problems include EEG/MEG measurements or dynamic Electrical Impedance Tomography. A usual assumption as a priori knowledge for these kind of problems is the temporal smoothness of the solutions leading to penalty terms of the form

\[
\sum_i \frac{\|x_{i+1} - x_i\|_F^2}{(t_{i+1} - t_i)^2}.
\]

For further information on the theory of dynamic inverse problems along with the derivation of solution algorithms and their application to EEG/MEG measurements, dynamic EIT and dynamic CT, we refer the reader to the papers [51, 52]. Furthermore, we refer the reader to Section 4.2 for further details on dynamic CT and to the featured work [3], where several nonnegative matrix factorization models are designed to build a joint reconstruction and feature extraction framework for dynamic inverse problems.

\section*{2.2. Nonnegative Matrix Factorization}

Nonnegative Matrix Factorization (NMF), originally introduced in 1994 as positive matrix factorization by Paatero and Tapper in [43], is, in its classical form, an unsupervised learning tool and belongs to the family of the classical linear dimensionality reduction methods, which are equivalent to low-rank matrix approximations (see Chapter 1). Different from other types of LDR approaches like the well-known principal component analysis, the NMF constraints the entries of the factorization matrices to be nonnegative. This key property makes the NMF a specialized tool for nonnegative data in the wide field of data analysis. Besides its capability to extract meaningful information out of given datasets, which does not distinguish the NMF from other feature extraction methods like the PCA, the additional nonnegativity constraint allows a parts-based representation of the dataset. In other words, NMF makes it possible to approximate each data sample by a nonnegative superposition of the extracted basis vectors. Furthermore, the nonnegativity constraint significantly simplifies the interpretability of the factor matrices, for example in the case when the entries correspond to measurements of nonnegative physical quantities.

This makes the NMF the ideal tool for nonnegative data, which has been extensively
used for data representation, feature extraction, compression, blind source separation of mixed systems, noise removal, model reduction, clustering and similar tasks (see [10, 23] and the references therein). Possible application fields include hyperspectral image analysis for blind hyperspectral unmixing, music analysis for audio source separation, text mining for topic recovery and document classification, gene expression analysis and other fields (see [23, Chapter 9], [10, Chapter 8] and the references therein for an overview of possible application fields). In the case of the featured articles of Part II of this work, the focus lies on medical imaging applications including dynamic Computed Tomography (CT) (see Section 4.2) and MALDI imaging (see Section 4.1), which is a mass spectrometry imaging method. Dynamic CT is considered in [3], where several NMF models are introduced in order to construct a joint reconstruction and feature extraction method. MALDI imaging is considered in three of the presented research papers, where different NMF models are used for a prior feature extraction step for classification [36], data representation [20] and spatially coherent clustering [18].

There are two main variants of the NMF problem, which are defined below. The notation for the matrices and their dimensions in the following definitions are used throughout Part I of this work.

**Definition 2.4** (Exact NMF). For a given nonnegative data matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, find, if possible, two matrices $U \in \mathbb{R}^{M \times K}_{\geq 0}$ and $V \in \mathbb{R}^{K \times N}_{\geq 0}$ with the factorization rank $K \ll \min\{M, N\}$, such that

$$X = UV.$$  \hfill (2.6)

**Definition 2.5** (NMF). For a given nonnegative data matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, find two matrices $U \in \mathbb{R}^{M \times K}_{\geq 0}$ and $V \in \mathbb{R}^{K \times N}_{\geq 0}$ with the factorization rank $K \ll \min\{M, N\}$, such that

$$X \approx UV.$$  \hfill (2.7)

Hence, NMF problems can be subdivided into two main groups: **Exact NMF**, where an exact factorization of $X$ is needed and **Approximate NMF** (or just **NMF**), where $X$ is only approximated. Throughout the literature, Exact NMF is much less studied than the NMF and is also not the main focus of this work. However, theoretical studies on its geometrical interpretation, the uniqueness of the problem as well as the so-called nonnegative rank can lead to interesting insights also for the general NMF problem. For more details on these topics, we refer the reader to Part I of the book [23].

This work mainly focuses on the approximate NMF along with the development of suitable solution algorithms, its relation to clustering methods and the design of suitable NMF models for real world applications. From the relation in (2.7), we can conclude
that every column $X_{\bullet,n}$ and row $X_{m,\bullet}$ can be written as a nonnegative superposition
of just a few basis vectors, so that $X_{\bullet,n} \approx \sum_k V_{kn} U_{\bullet,k}$ and $X_{m,\bullet} \approx \sum_k U_{mk} V_{k,\bullet}$. In
the same sense, the whole matrix $X$ can be approximated as $X \approx \sum_{k=1}^K U_{\bullet,k} V_{k,\bullet}$.
Hence, NMF can also be seen as a basis learning tool with additional nonnegativity
constraints.

NMF problems are typically NP-hard, non-linear, ill-conditioned and suffer from
the nonuniqueness of the solutions [23, 44, 32]. Furthermore, the NMF problem in Defini-
tion 2.5 usually leads to a non-convex problem when reformulating it as an optimization
problem. Hence, the question arises how to construct suitable solution techniques for
NMF problems. The following sections give a short overview on this topic with a focus
on iterative algorithms.

Finally, we would like to refer the interested reader to two books, which are currently
published and solely devoted to NMF and Nonnegative Tensor Factorizations (NTF).
First, there is the book by Cichocki et al. [10] from 2009, which focuses on iterative
algorithms and applications for NMF as well as for NTF. Finally, there is the very
recent book of Nicolas Gillis [23] from 2020 published in SIAM, which provides more
theoretical aspects of the NMF.

2.2.1. Variational Formulation

In the case of approximative NMF factorizations, the original NMF problem in Defini-
tion 2.5 is reformulated as an optimization problem together with a discrepancy term
$D$, which measures the error between the data matrix $X$ and the factorization $UV$,
as well as additional regularization terms $R_i$ to regularize the NMF problem and to
enforce additional properties to the matrices.

For a given NMF problem, the discrepancy term is chosen based on the statistical
distribution of the noise and can be obtained based on the corresponding maximum like-
lihood estimators. The derivation of the common error measures shall not be discussed
here, since this can be found in standard literature [23, Section 5.1]. However, we will
shortly overview the most common discrepancy terms for NMF problems and focus on
the $\beta$-divergences, which are a special case of the more general Bregman divergences.

**Definition 2.6 (Divergence [27]).** Let $\mathcal{X}$ be an arbitrary set. A function $D : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$
is called a *divergence*, if it fulfills the following properties:

(i) $D(x, y) \geq 0 \quad \forall x, y \in \mathcal{X},$

(ii) $D(x, y) = 0 \iff x = y.$
2.2. Nonnegative Matrix Factorization

Definition 2.7 ($\beta$-Divergence [27]). The $\beta$-divergence for $\beta \in \mathbb{R}$ is the function $d_\beta : \mathbb{R}_{>0} \times \mathbb{R}_{>0} \to \mathbb{R}_{\geq 0}$ defined as

$$d_\beta(x, y) := \begin{cases} x^\beta \beta (\beta - 1) + y^\beta \beta - \frac{x y^{\beta-1}}{\beta - 1} & \text{for } \beta \in \mathbb{R} \setminus \{0, 1\}, \\ x \log \left(\frac{x}{y}\right) - x + y & \text{for } \beta = 1, \\ \frac{x}{y} - \log \left(\frac{x}{y}\right) - 1 & \text{for } \beta = 0. \end{cases}$$ (2.8)

Accordingly, the $\beta$-divergence between matrices $A$ and $B$ is defined as

$$D_\beta(A, B) := \sum_{i,j} d_\beta(A_{ij}, B_{ij}).$$

Typical examples of $\beta$-divergences, which are typically used for NMF problems, are the Frobenius norm with $\beta = 2$ for Gaussian noise or the Kullback-Leibler (KL) divergence with $\beta = 1$ for Poisson noise. Both examples are further analyzed as part of a generalized framework for the derivation of so-called multiplicative update rules (see Section 3.2.1) in the featured survey article [20] in Part II of this thesis. The other featured papers primarily discuss the standard case of the Frobenius norm. Note that a correct choice of the discrepancy term for the considered NMF problem together with the corresponding application field is crucial for the performance of the solution algorithm. For a review on this topic and further error measures, we refer the reader to [23, Section 5.1] and [10, Chapter 2].

Furthermore, NMF problems are ill-posed due to the non-uniqueness of the solution [32, 44]. Hence, they usually require regularization techniques such as the ones based on the generalized Tikhonov framework discussed in Section 2.1, which is typically done by including suitable regularization terms $R_i$ into the NMF model. By using prior information of the factorization matrices, it is possible to regularize the NMF problem, to enforce some additional properties to the matrices for the specific application and to obtain in general better estimates of the solution. Choosing the right NMF model for the considered application is an important part when using NMF in practice.

Hence, the general regularized NMF optimization problem can be formulated as

$$\min_{U, V \geq 0} D(X, UV) + \sum_{i=1}^l \phi_i R_i(U, V) = \min_{U, V \geq 0} F(U, V),$$ (2.9)

where $F$ is the objective function and $\phi_i \geq 0$ are the regularization parameters, which
control the influence of the penalty terms $R_i$ to the NMF model. Note that it is possible to extend the minimization with respect to additional auxiliary variables to facilitate the minimization or to enable specifically designed penalty terms.

This generalized NMF problem can be seen as a base for the majority of the considered NMF models in the featured articles. Table 2.1 provides a list of the regularization terms, which are considered in the presented research papers of Part II.

Typical regularization terms used in these works range from standard $\ell_1$ and $\ell_2$ regularization (see Section 2.1) to more problem specific terms such as TV regularization or penalty terms which enforce orthogonality constraints to the matrices. We would like to note that in [18], further non-differentiable versions of TV regularization terms are considered, which are not listed in Table 2.1. Similarly, in [36], further regularization terms leading to a logistic regression scheme are considered. We will not go into the details regarding the properties and use cases of the regularization strategies listed in Table 2.1 and refer instead to the featured papers of Part II. However, since four of the featured research papers [36, 20, 18, 21] consider penalty terms for orthogonality leading to a unique NMF model with a strong relationship to clustering approaches, we will discuss shortly some main properties of the so-called orthogonal NMF in the

<table>
<thead>
<tr>
<th>Name</th>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tikhonov</td>
<td>$R(U, V) = |U|^2_F + |V|^2_F$</td>
<td>Standard $\ell_2$ regularization [36, 20, 3, 21].</td>
</tr>
<tr>
<td>Sparsity</td>
<td>$R(U, V) = |U|_1 + |V|_1$</td>
<td>Standard $\ell_1$ regularization [36, 20, 3, 21].</td>
</tr>
<tr>
<td>Orthogonality</td>
<td>$R(U, V) = |U^\top U - I|^2_F$, $R(U, V) = |VV^\top - I|^2_F$</td>
<td>Orthogonality constraints to enforce uncorrelated features [36] or to design a clustering framework [21, 18].</td>
</tr>
<tr>
<td>Total Variation</td>
<td>$R(U, V) = TV(U)$, $TV(U) := \sum_{m,k} \varepsilon_{TV} + \sum_{\ell \in N_m} (U_{mk} - U_{\ell k})^2$, $N_m$ index of neighbouring pixels, $\varepsilon_{TV} &gt; 0$</td>
<td>Standard TV regularization for image processing tasks [20], regularization of inverse problems [3] or the design of ONMF models for spatially coherent clusterings [18].</td>
</tr>
<tr>
<td>Supervised NMF</td>
<td>$R(U, V) = |u - XV^\top \beta|^2_2$, $u$ class labels, $\beta$ model parameters</td>
<td>Penalty term, which allows supervised classification schemes, where the NMF is used as a prior feature extraction step [36].</td>
</tr>
</tbody>
</table>

Table 2.1: Majority of the considered regularization strategies in the featured articles of Part II together with the corresponding definition, a short description and the references to the respective papers.
A further main aspect of the whole NMF workflow is an optimal choice of the occurring hyperparameters of the NMF model including the factorization rank \( K \) and the various regularization parameters \( \varphi_i \) when working with generalized NMF models as in (2.9). Typically used approaches are based on cross validation, approximative or heuristic techniques such as residual analysis or core consistency diagnostics, the use of expert knowledge or other statistically motivated methods. We will not go into further details and refer instead to both books on NMF [10, 23] and in particular to the featured work [3] of Part II, where the optimal choice of the model-order \( K \) is based on analyzing the importance of the rank-one matrices \( U_{\bullet,k} V_{k,*} \).

Furthermore, we should note that by changing the structure or the constraints of the general NMF problem in (2.9), it is possible to construct various NMF models which are different from the standard NMF approach. For example, there exists the so-called near separable NMF, which is defined by the minimization problem

\[
\min_{\mathcal{I} \subseteq \{1, \ldots, N\}} \min_{V \in \mathbb{R}^{K \times N} \geq 0} D(X, X_{\bullet,\mathcal{I}} V) \text{ such that } |\mathcal{I}| = K.
\] (2.10)

Hence, it is assumed in this problem that there exist some index set \( \mathcal{I} \), such that \( U = X_{\bullet,\mathcal{I}} \) with \( U \) being the factorization matrix in the standard NMF problem in Definition 2.5. Moreover, there exist many other variants including semi-NMF, which only constraints \( V \) to be nonnegative, symmetric NMF, which considers the special case \( V = U^T \) and hence the NMF problem \( X \approx UU^T \) or nonnegative matrix trifactorization, which considers the problem of finding an additional matrix \( W \in \mathbb{R}^{K_1 \times K_2} \) along with \( U \in \mathbb{R}^{M \times K_1} \) and \( V \in \mathbb{R}^{K_2 \times N} \), such that \( X \approx UWV \). Linking these NMF models and finding relationships to existing problems in the literature is an important knowledge to choose the most appropriate model for the given data. However, providing an overview on the existing variants of NMF problems goes beyond the scope of this work. Hence, we refer the reader to the nice overviews in [23, 10].

Finally we note that there exist further theoretical studies on the computational complexity of the NMF problem in the standard case of an approximate factorization. One major result regarding this topic is that, as in the case of Exact NMF, the standard NMF problem is also NP-hard if the factorization rank \( K \) is part of the input. However, the analysis of the computational complexity of NMF problems is not the focus of this thesis and we refer the reader for a more detailed treatise to [23, Chapter 6].
2.2.2. Orthogonal NMF and Clustering

A specific variant of the NMF problem in (2.9) is the so-called *Orthogonal NMF* (ONMF), which introduces an additional orthogonality constraint on one or both of the factorization matrices. In this section, we introduce the ONMF problem and discuss its basic relation to clustering methods, since many of the featured papers deal with a variety of different ONMF models [36, 20, 18, 21].

There exist numerous different versions of ONMF models throughout the literature, such as combinations of orthogonality constraints on one or both matrices $U$ and $V$ together with tri-factorizations or symmetric factorizations [15, 38]. However, we focus in this work only on the so-called *uni-orthogonal* NMF together with the Frobenius norm as discrepancy term, since this is also the focus of the research articles of Part II. This ONMF model corresponds to the standard NMF with an additional orthogonality constraint on one of the matrices $U$ or $V$. In the following, we choose the optimization problem

$$
\min_{U, V \geq 0, \quad U^T U = I_{K \times K}} \| X - UV \|^2_F, \quad (2.11)
$$

where $I_{K \times K}$ is the identity matrix of size $K \times K$. It can be shown that this problem and ONMF models in general have a strong relationship to clustering approaches.

The main objective of a clustering framework is to partition a given set of objects into groups, so that objects within a group are more similar to each other than objects, which come from different groups. Mathematically, the clustering task can be formalized as follows: For a given set of indices $\{1, \ldots, M\}$ of a corresponding dataset $\{x_m \in \mathbb{R}^N \mid m = 1, \ldots, M\}$, find $K$ disjoint index sets $\mathcal{I}_k \subset \{1, \ldots, M\}$, so that $\bigcup_{k=1,\ldots,K} \mathcal{I}_k = \{1, \ldots, M\}$. The index sets $\mathcal{I}_k$ are typically referred to as *clusters*.

However, this mathematical formulation still lacks a proper notion of similarity to compare the samples $x_m$ to each other. In order to understand the connection between ONMF models and clustering approaches, we consider in the following $K$-means clustering. In this case, a similarity measure $\text{dist}(\cdot, \cdot)$ is introduced to measure the similarity between two data samples. A default choice for such a measure is the squared Euclidean distance $\text{dist}(x_i, x_j) := \|x_i - x_j\|^2_2$.

Furthermore, so-called *centroids* $c_k \in \mathbb{R}^N$ are computed to obtain a representative for each of the clusters $\mathcal{I}_k$. The computation of both the centroids and the clusters is based on the minimization of the within-cluster variation, which is given by the objective function $F := \sum_{k=1}^K \sum_{m \in \mathcal{I}_k} \text{dist}(x_m, c_k)$. As it is the case for NMF problems, this minimization task is NP-hard [40] and motivates to seek for approximate solutions.
One classical approach is the $K$-means algorithm and is based on an alternating minimization approach similar to the iterative algorithms for NMF problems (see Section 3.2).

The relationship between $K$-means clustering and ONMF can be seen by varying some constraints of both problems. In order to compare $K$-means to an NMF model, it is needed to additionally constrain the data samples to be nonnegative, such that $x_m \in \mathbb{R}^N_{\geq 0}$ for all $m \in \{1, \ldots, M\}$. Furthermore, the data samples $x_m$ are written row-wise to a data matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, so that $X = [x_1, \ldots, x_M]^\top$. Moreover, we define the so-called *cluster membership* matrix $\tilde{U} \in \{0,1\}^{M \times K}$ to be

$$
\tilde{U}_{mk} := \begin{cases}
0 & \text{if } m \notin I_k, \\
1 & \text{if } m \in I_k,
\end{cases}
$$

as well as the *centroid matrix* $\tilde{V} := [c_1, \ldots, c_K]^\top$, which contains row-wise the centroids of the $K$-means algorithm. With this, it is easy to see that the objective function $F$ of the $K$-means algorithm can be rewritten as $F = \|X - \tilde{U} \tilde{V}\|_F^2$ and hence has the same structure as a usual NMF discrepancy term (see [21]).

Note that it is necessary to assume the above structure of the matrix to obtain this relationship and that the columns of $\tilde{U}$ are not orthonormal to each other due to the missing normalization constraint. However, it holds that the columns of $\tilde{U}$ are orthogonal to each other, i.e. $\langle U_{\bullet, k}, U_{\bullet, \tilde{k}} \rangle = 0$ for $k \neq \tilde{k}$. This ensures that every row $\tilde{U}_{m, \bullet}$ has at most one positive entry $\tilde{U}_{mk}$ indicating the association of the data sample $x_m$ to the cluster $I_k$. The same holds true for the matrix $U$ in the ONMF model (2.11) together with the combination of the nonnegativity and orthogonality constraints. Hence, the orthogonality constraint without any additional structure as in the cluster membership matrix already provides the basic clustering interpretability, since this prevents that data samples are associated to multiple clusters. However, note that this constraint could also lead to empty clusters.

It can be shown that the NMF problem (2.9) with $D = D_2$, no additional regularization terms and $U$ satisfying the conditions of the weighted cluster membership matrix

$$
U_{mk} := \begin{cases}
0 & \text{if } m \notin I_k, \\
\frac{1}{\sqrt{|I_k|}} & \text{if } m \in I_k,
\end{cases}
$$

is equivalent to the classical $K$-means clustering problem [14, 46]. Furthermore, it can be shown that the standard ONMF problem in (2.11) with the more relaxed constraints
is equivalent to a weighted variant of spherical $K$-means [46]. For more results regarding the relationship between ONMF problems and $K$-means clustering, we refer to the featured article [21] and the references therein. In this work, a short review on the existing results on the connections between ONMF and $K$-means clustering is provided. Furthermore, regularized ONMF models are studied with respect to their connections to generalized $K$-means models. In particular, the effect of the regularization terms in the ONMF problem on the distance measures and centroids of the $K$-means framework is analyzed.

In order to simplify the derivation of suitable solution algorithms for the ONMF models, the hard orthogonality constraint in (2.11) is oftentimes written as a penalty term in the objective function of the ONMF problem. For a general discrepancy term $D$, this yields

$$\min_{U, V \geq 0} D(X, UV) + \sigma \frac{1}{2} \| U^T U - I_{K \times K} \|_F^2,$$

(2.14)

where $\sigma \geq 0$ is the corresponding regularization parameter. These kind of ONMF models are used throughout many of the featured research articles in Part II. [20] derives multiplicative update rules for ONMF models involving additional penalty terms based on the Majorization-Minimization framework (see Section 3.2). The work [36] use ONMF models for extracting more uncorrelated features for the use of a supervised classification scheme. Finally, [18] introduces spatially coherent clustering methods based on ONMF models such as (2.14) with an additional total variation regularization procedure.

For an overview of possible solution methods for ONMF problems, we refer the reader to the following chapter on general solution approaches for NMF and to [18, Section 1.1] of Part II for a more specific overview on algorithms for ONMF throughout the literature.
3 | Solution Strategies for NMF

The algorithm development of NMF problems constitutes a major part of the featured research articles. This section is solely devoted to solution strategies for NMF problems and gives a short overview on common optimization techniques.

Section 3.1 gives a small overview on common solution methods for NMF problems including techniques, which do not belong to the large family of iterative algorithms. However, the latter constitutes the main focus of this thesis and are described in Section 3.2 with a specific focus on two different types of algorithms. Section 3.2.1 is devoted to the classical multiplicative update rules for NMF along with the so-called majorization-minimization principle and some convergence results, which is part of many of the presented research articles in Part II [20, 3, 18, 36]. The following Section 3.2.2 discusses the so-called proximal alternating linearized minimization scheme together with some basic convergence results, which is a central part of the featured article in [18]. Finally, Section 3.2.3 gives a short overview on possible initialization techniques and stopping criteria for these kind of algorithms.

3.1. Overview

Due to the huge amount of possible solution techniques for NMF problems throughout the literature, this thesis does not intend to give an exhaustive and detailed overview on each possible approach. However, this section intends to give a rough summary and a categorization on some of the most common methods. In the following section 3.2, we then focus on two different iterative algorithms for the standard case of an approximative NMF factorization, since both approaches will be extensively used in the featured research papers of Part II.

In order to categorize the different solution techniques for NMF properly, it is meaningful to subdivide in a first step the different NMF problems into exact and approximate factorizations (see Definition 2.4 and 2.5). Besides some theoretical analysis regarding the geometric interpretation and computational complexity in the case of the exact NMF [23], some few heuristic approaches to obtain a solution have been
analyzed throughout the literature (see [23, Chapter 2]).

Much more work has been done for the NMF problem with an approximate factor-
ization. In the special case of the near-separable NMF defined by the optimization
problem in (2.10), several different classes of algorithms have been proposed including
sequential algorithms along with successive projection schemes, which try to construct
the solution $I$ by identifying one index at a time. Further approaches include heuristic
techniques or convex-optimization-based algorithms. For a detailed overview of these
types of approaches, we refer the reader to [23, Chapter 7].

However, the vast majority of the algorithm development throughout the literature
has been done for the standard NMF problem presented in Section 2.2 and is based on
iterative algorithms, which are also in the focus of this work. A short introduction and
overview on these methods is provided in the following section with a subsequent focus
on multiplicative update rules and proximal alternating minimization schemes.

### 3.2. Iterative Algorithms

Most of the iterative update schemes for the standard NMF problem in Section 2.2
are based on an alternating two-block coordinate descent framework. Hence, in the
case of the standard NMF problem given by the optimization problem with the general
objective function in (2.9), the algorithm optimizes with respect to $U$ for fixed $V$ and
with respect to $V$ for fixed $U$ yielding the update rules

$$U^{[i+1]} = \arg \min_{U \geq 0} F(U, V^{[i]}),$$  \hspace{1cm} (3.1)

$$V^{[i+1]} = \arg \min_{V \geq 0} F(U^{[i+1]}, V).$$  \hspace{1cm} (3.2)

The main motivation behind this minimization scheme is the fact that for usual choices
of the objective function, $F$ is usually convex in each of the variables $U$ and $V$
but non-convex in $(U, V)$. Hence, the subproblems in (3.1) and (3.2) are convex
and therefore efficiently solvable. Furthermore, in the case that $F$ only consists of
the discrepancy term $D$ (see Section 2.2.1), where $D$ fulfills the symmetry condition
$D(X, UV) = D(X^\top, V^\top U^\top)$, it is hence possible to get the update rule of $V$
free by only deriving the algorithm for $U$ and vice versa. Note that the two-block coordi-
nate descent framework can be easily generalized to $n$ blocks in the case that multiple
auxiliary variables are needed in the considered NMF model (see also the featured ar-
ticles [20, 3, 18, 36]). For the very few papers throughout the literature, which derive
nonalternating update rules for NMF, we refer the reader to [23, Chapter 8] and the
There exist many different types of iterative algorithms to find a solution to the above minimization problems including the multiplicative update rules and the proximal alternating linearized minimization scheme, which will be discussed in more detail in the following two sections since they constitute a major part in several of the featured articles of Part II [20, 3, 18, 36]. Further typical approaches, which are partially also used as comparative methods in the presented articles, are for instance based on classical Alternating Least Squares (ALS) methods along with their modifications Alternating Nonnegative Least Squares (ANLS) and Hierarchical Alternating Least Squares (HALS), which can be used when the discrepancy term \( D \) is the Frobenius norm (see [23, Chapter 8] and [10, Chapter 4]). Further methods include projected gradient methods, which consist of a gradient descent step and a subsequent projection step to ensure nonnegativity, the alternating direction method of multipliers, which decouples the least squares term and the nonnegativity constraint and quasi-Newton Methods, which belong to second-order optimization techniques. For a detailed review of these and also other approaches, we refer the reader to [23, Chapter 8] and [10, Chapter 3-6].

Along with these numerous algorithms, there also exist many different approaches to derive the corresponding update rules. Some of the main methods, by which many of the above mentioned first-order optimization algorithms can be constructed, include the analysis of the first-order optimality conditions together with the Karush-Kuhn-Tucker (KKT) conditions and the Majorization-Minimization (MM) framework. The latter approach is also often used for the derivation of multiplicative update rules, which are a major part of several featured articles in Part II of this thesis. Hence, the MM framework will be introduced in more detail in the next Section along with the description of multiplicative algorithms for NMF.

### 3.2.1. Multiplicative Algorithms

Multiplicative update rules (MU) belong to the most well-known algorithms for NMF and have been proposed for the first time in the context of NMF by the work of Lee and Seung [34] published in Nature in 1999, which popularized the NMF for the analysis of nonnegative data in general. However, MU algorithms were already derived much earlier in 1972 in [48] for the Kullback-Leibler divergence before even being classified as multiplicative update rules in 1993 by [13]. After the famous papers of Lee and Seung [34, 35], further MU algorithms for other discrepancy terms including the Itakura-Saito (IS) divergence [22] and regularized NMF models [12, 33, 20] were derived. For a deeper historical background on MU algorithms in context with the NMF, we refer the reader...
to [23, Section 8.2] and the references therein.

The principle of MU algorithms is based on the fact that they consist only of summations as well as entrywise multiplications and divisions of matrices. Hence, they ensure the nonnegativity constraint of the NMF problem automatically without any additional projection step as long as the matrices are initialized with nonnegative entries. One of the key characteristics of MU algorithms is that they can be easily implemented due to their simple structure described above. Furthermore, together with the MM framework, they can be easily derived in most cases and are extremely adaptable to many different NMF models with multiple penalty terms. This can be seen in the featured articles of Part II, where multiplicative update rules are derived for a variety of application-oriented NMF models [20, 36, 3, 18].

Finally, in some cases, they are comparable to state-of-the-art algorithms regarding their convergence speed in particular if the objective function is the KL divergence [28]. However, in many other cases and especially for dense datasets with the Frobenius norm as discrepancy term, they perform rather poorly compared to other optimization methods [23, Section 8.4].

In the following, we will describe the MM principle for the derivation of multiplicative algorithms and derive with this workflow the classical MU algorithm proposed by Lee and Seung in [35] as an example. However, since Part II already features a survey paper on this topic including a detailed introduction on the MM framework and MU algorithms [20], we try to avoid too many details and just provide a short overview. Finally, we note that MU algorithms can also be derived via the KKT conditions (see e.g. [10, Chapter 3]). However, this thesis together with the research articles in Part II focus on the case of the MM principle.

The basic idea of the MM framework is to replace the objective function $F$ for each minimization problem in (3.1) and (3.2) with a corresponding majorizing auxiliary function, whose specific properties lead to a simplified minimization, a monotone decrease of the original cost function $F$ and the desired multiplicative update rules due to its tailored construction.

**Definition 3.1 (Auxiliary Function).** Let $\Omega \subset \mathbb{R}^n$ be an open subset and $F : \Omega \rightarrow \mathbb{R}$ a given objective function. A function $Q_F$ is called a majorizer, a surrogate function or an auxiliary function, if it fulfills the following two conditions:

(i) $Q_F(x, \tilde{x}) \geq F(x)$ for all $x, \tilde{x} \in \Omega$,

(ii) $Q_F(x, x) = F(x)$ for all $x \in \Omega$. 

3.2. Iterative Algorithms

The minimization scheme of the MM framework is defined with the update rule

$$x^{[i+1]} := \arg\min_{x \in \Omega} Q_F(x, x^{[i]})$$  \hspace{1cm} (3.3)

with the assumption that $\arg\min_{x \in \Omega} Q_F(x, \tilde{x})$ exists for all $\tilde{x} \in \Omega$. Hence, instead of optimizing $F$ directly, the surrogate function with respect to the first argument is minimized. Together with the characteristic properties of the majorizer given in Definition 3.1, this leads to the monotone decrease of the objective function, since

$$F(x^{[i+1]}) \leq Q_F(x^{[i+1]}, x^{[i]}) \leq Q_F(x^{[i]}, x^{[i]}) = F(x^{[i]}).$$  \hspace{1cm} (3.4)

This principle is illustrated in Figure 3.1. Typical majorizers are additionally strictly convex with respect to the first variable to ensure the unique existence of the solution of (3.3) and to ensure the convergence of the iterates (see Section 3.2.1.1). In addition, this allows the solution of the minimization problem based on the zero gradient condition $\nabla x Q_F(x) = 0$. Furthermore, an explicit solution to this equation can often be obtained if the surrogate function is separable with respect to the first variable, i.e. it can be written as a sum, where each term only depends on one component of $x$. Finally, using tailored construction techniques for majorizers ensure a multiplicative structure of the update rule given in (3.3).

Figure 3.1.: Illustration of two iterations of the MM principle for a nonconvex objective function $F$ with bounded curvature and a majorizing surrogate function $Q_F$, which is strictly convex in the first argument.
Possible construction techniques for auxiliary functions are based on the Quadratic Upper Bound Principle (QUBP), Jensen’s inequality or other theoretical frameworks. These frameworks are extremely versatile for different NMF models with multiple penalty terms leading to the high flexibility of the MU algorithms. A detailed review of possible techniques is given in the featured work [20], so we do not intend to give an overview at this point. Instead, we aim to derive exemplarily the MU algorithm by Lee and Seung in [35] for the standard NMF problem

$$\min_{U,V \geq 0} \frac{1}{2}\|X - UV\|^2_F =: F(U, V)$$

(3.5)

based on the QUBP. The latter is based on the fact, that for twice continuously differentiable functions $G : \Omega \rightarrow \mathbb{R}$ with bounded curvature and $\Omega \subset \mathbb{R}^n$ an open and convex subset, there exists a matrix $A \in \mathbb{R}^{n \times n}$, so that

$$Q_G(x, \tilde{x}) := G(\tilde{x}) + \nabla G(\tilde{x})^\top (x - \tilde{x}) + \frac{1}{2} (x - \tilde{x})^\top A (x - \tilde{x})$$

is a quadratic auxiliary function for $G$. Furthermore, it can be shown that the zero gradient condition $\nabla_x Q_G(x, \tilde{x}) = 0$ gives the unique minimizer

$$x = \tilde{x} - A^{-1} \nabla G(\tilde{x})$$

(3.6)

if the matrix $A$ is additionally symmetric and positive definite. Hence, the MU algorithms in combination with the QUBP framework can also be seen as a classical gradient descent method with a specific chosen step size $A$. For quadratic objective functions, which is also the case in (3.5), typical choices for $A$ are diagonal matrices of the form

$$A(\tilde{x})_{ii} := \left(\frac{\nabla^2 G(\tilde{x}) \tilde{x}_i}{\tilde{x}_i} + \kappa_i\right)$$

and are dependent on the second argument of the majorizer. The additional parameters $\kappa_i \geq 0$ are usually chosen depending on regularization terms of the NMF model. The diagonal structure of $A$ ensures its simple inversion and the separability of the corresponding surrogate function.

In the case of the NMF problem in (3.5) the minimization scheme is a two-block coordinate descent framework as described in (3.1) and (3.2). Hence, for both minimization problems, an auxiliary function has to be determined via the QUBP. However, it is possible to derive the update rules for $U$ based on the QUBP and obtain the algorithm for $V$ simply by symmetry arguments as described in Section 3.2.
For the minimization with respect to $U$, we consider the objective function $F(U) = \frac{1}{2}\|X - UV\|_2^2$, use its separability with respect to the rows of $U$ and write

$$F(U) = \sum_{m=1}^{M} \frac{1}{2}\|X_{m,\bullet} - U_{m,\bullet}V\|_2^2 = \sum_{m=1}^{M} F_m(U_{m,\bullet}).$$

By computing the gradients $\nabla_U F_m(U) = (U_{m,\bullet}V - X_{m,\bullet})V^\top$ and $\nabla^2_{UU} F_m(U) = VV^\top$, setting $\kappa_i = 0$ and inserting them in (3.6) and (3.7) gives

$$U_{m,\bullet} = \bar{U}_{m,\bullet} - \Lambda^{-1}((\bar{U}_{m,\bullet}V - X_{m,\bullet})V^\top) = \bar{U}_{m,\bullet} \odot \frac{X_{m,\bullet}V^\top}{\bar{U}_{m,\bullet}VV^\top},$$

where $\odot$ is the element-wise (Hadamard) multiplication and the fraction line the element-wise division. Hence, the negative parts of the term cancel out leading to the desired multiplicative structure of the MU framework. Applying the MM framework of (3.3) and extending the equation to the whole of $U$, we get the classical update rule of Lee and Seung for $U$:

$$U^{[i+1]} = U^{[i]} \odot \frac{XV^\top}{U^{[i]}VV^\top}.$$  \hspace{1cm} (3.8)

By symmetry, as described in Section 3.2, we can obtain the algorithm for $V$, which shall not be discussed in detail.

Finally, we would like to mention that MU algorithms typically suffer from the so-called zero locking phenomenon (see [23, Section 8.2.5] and the references therein). In many applications such as for the extraction of spectra in MALDI imaging, sparse matrices are desirable and can be achieved by adding sparsity terms into the NMF model. However, in the case of the MU algorithms, zero entries in the matrices can prevent the convergence to a stationary point or lead to numerical instabilities if some entries of the denominator might become equal to zero.

In a first step, this can be partially prevented by initializing the matrices of the MU algorithm with strictly positive entries. Due to the multiplicative structure of the update rules, these entries should theoretically stay strictly positive even if some entries tend to zero at convergence during the iterations. However, due to the finite precision of the machine, some entries might be set to zero during the course of the algorithm. This can be prevented by using a lower bound and projecting the small entries of the matrices to a small positive number $\varepsilon > 0$, which is typically chosen as the machine epsilon. In the case of the derived update rule in (3.8), one would instead update $U$ as

$$U^{[i+1]} = \max \left( \varepsilon, U^{[i]} \odot \frac{XV^\top}{U^{[i]}VV^\top} \right).$$
3.2.1.1. Convergence Theory

In this section, we shortly discuss the convergence of the MU algorithms in combination with the $n$-block coordinate descent framework and the MM principle.

In a first step, it is easy to see that the values of the objective function converge. This is primarily due to the monotone decrease of the cost function showed in (3.4) based on the MM framework. In the case of the standard NMF problem in (2.9) for two variables $U$ and $V$ and the corresponding objective function $F$, this leads to

$$F(U^{[i]}, V^{[i]}) \geq F(U^{[i+1]}, V^{[i]}) \geq F(U^{[i+1]}, V^{[i+1]}),$$

where $(U^{[i]}, V^{[i]})$ are the iterations based on the MM framework. Since the objective is bounded from below, this leads to the convergence of the values of the cost function $F$.

However, the convergence of the iterates $(U^{[i]}, V^{[i]})$ is still unclear. To obtain some information on their convergence, it is needed to look at the so-called Block Successive Upper-bound Minimization (BSUM) framework [47]. Loosely speaking, the BSUM framework corresponds to the MM principle, where the majorizers, which are constructed for each block of variables, have to fulfill more properties than in the MM framework. If this is the case, the convergence of the iterates to a stationary point of the objective function can be ensured.

In the following, we introduce the BSUM framework and state the needed properties of the majorizers for the convergence of the iterates based on the work [47]. Let $X := X_1 \times \cdots \times X_n$ with $X_i \subset \mathbb{R}^{m_i}$ be closed convex sets and $m := \sum_i m_i$. Furthermore, we use the notation $X \ni x := (x_1, \ldots, x_n)$ for the variable blocks $x_i \in X_i$ for $i \in \{1, \ldots, n\}$. For a continuous function $F : X \to \mathbb{R}$, the following minimization problem is considered:

$$\min_{x \in X} F(x).$$

(3.9)

As in the MM framework, an alternating minimization scheme is used as in (3.1) and (3.2) and suitable majorizers $Q_i : X_i \times X \to \mathbb{R}$ are constructed for each of the variable blocks. Hence, instead of minimizing $F$ directly, the update scheme

$$x_i^{[j+1]} \in \arg \min_{x_i \in X_i} Q_i(x_i, x_i^{[j]}),$$

(3.10)

is considered. Note that without any additional requirements on the majorizers, the solution of the corresponding subproblems might not be unique. Before stating the
needed properties of the majorizers for the convergence, we need the notion of a
generalized directional derivative and a quasi-convex function.

**Definition 3.2** (Lower Dini Directional Derivative). Let $F : \mathcal{X} \to \mathbb{R}$ be a continuous function for a convex subset $\mathcal{X} \subset \mathbb{R}^m$. The so-called **lower Dini directional derivative** of $F$ at $x \in \mathcal{X}$ in direction $d \in \mathcal{X}$ is defined by

$$F'(x; d) := \lim \inf_{\lambda \searrow 0} \frac{F(x + \lambda d) - F(x)}{\lambda}.$$ 

**Definition 3.3** (Quasi-Convex Functions). Let $F : \mathcal{X} \to \mathbb{R}$ be a function with $\mathcal{X} \subset \mathbb{R}^m$ a convex set. $F$ is called **quasi-convex** if

$$F(\lambda x + (1 - \lambda)y) \leq \max\{F(x), F(y)\} \quad \forall \lambda \in (0, 1), \forall x, y \in \mathcal{X}.$$ 

Note that the limit of the lower Dini directional derivative always exists even for discontinuous functions although the limit is not necessarily finite.

Furthermore, we need the concept of a coordinatewise minimum of a function as well as the regularity of a function.

**Definition 3.4** (Coordinatewise Minimum of a Function). A point $z \in \text{dom } F \subset \mathbb{R}^m$ is a **coordinatewise minimum** of $F$ with respect to the coordinates $\mathbb{R}^{m_1}, \ldots, \mathbb{R}^{m_n}$ with $\sum_i m_i = m$, if

$$F(z) \leq F(z + d_k^1) \quad \forall d_k \in \mathbb{R}^{m_k} \quad \text{with} \quad d_k^0 = (0, \ldots, d_k, \ldots, 0),$$

where $z + d_k^0 \in \text{dom } F$ for all $k \in \{1, \ldots, n\}$.

**Definition 3.5** (Regularity of a Function). Let $F : \mathbb{R}^m \to \mathbb{R}$ be a continuous function. $F$ is called **regular** at the point $z \in \text{dom } F$ with respect to the coordinates $m_1, \ldots, m_n$ with $\sum_i m_i = m$, if $F'(z; d) \geq 0$ for all $d = (d_1, \ldots, d_n)$ with $F'(z; d_k^0) \geq 0$, where $d_k^0 = (0, \ldots, d_k, \ldots, 0)$ with $d_k \in \mathbb{R}^{m_k}$ for all $k \in \{1, \ldots, n\}$.

Hence, a coordinatewise minimum $z$ of a given function $F$ is a stationary point as long as $F$ is regular at $z$.

In order to set the basic fundament for the convergence of the BSUM iterates to a stationary point of the objective function, the following conditions on the auxiliary functions $Q_i$ are needed [47]:

(M1) $Q_i(x_i, x) = F(x) \quad \forall x \in \mathcal{X}, \forall i,$
3. Solution Strategies for NMF

\[ Q_i(x_i, y) \geq F(y_1, \ldots, y_{i-1}, x_i, y_{i+1}, \ldots, y_n) \quad \forall x_i \in X_i, \forall y \in X, \forall i, \]

(M3) \( Q_i \) is continuous for all \( i \),

(M4) \( Q'_i(x_i, y; d_i)|_{x_i=y_i} = F'(y; d) \quad \forall d = (0, \ldots, d_i, \ldots, 0), \text{ s.t. } y_i + d_i \in X_i \forall i. \]

The properties (M1)-(M2) corresponds to the ones of Definition 3.1. The condition in (M4) ensures that the first order behaviour of \( Q_i(\cdot, x^{[j-1]}_i) \) is locally the same as \( F(\cdot) \). While this technical property might be difficult to check, the authors in [47] provide sufficient conditions, which are easier to handle. Along with these conditions, [47] is able to show the following convergence results for the BSUM framework:

**Theorem 3.6** (Convergence of BSUM [47]).

(i) Assume that the conditions (M1)-(M4) hold and that \( Q_i(x_i, y) \) is quasi-convex in \( x_i \) for all \( i \in \{1, \ldots, n\} \). Furthermore, we suppose that the subproblem in (3.10) has a unique solution for any \( x^{[j]} \in X \). Then, every limit point \( x^* \) of the iterates generated by the algorithm of BSUM is a coordinatewise minimum of the minimization problem in (3.9). If \( F \) is additionally regular in \( z \), then \( z \) is a stationary point of (3.9).

(ii) Assume that the conditions (M1)-(M4) hold and that \( X^0 := \{ x \mid F(x) \leq F(x^{[0]}) \} \) is compact, i.e. the iterates of the BSUM algorithm lie in a compact set. Furthermore, we suppose that \( F \) is regular at any point in \( X^0 \) and that (3.10) has a unique solution for any \( x^{[j]} \in X \) for at least \( n-1 \) variable blocks. Then, the iterates of BSUM converge to the set of stationary points \( X^* \), i.e. it holds that

\[
\lim_{j \to \infty} \left( \inf_{x^* \in X^*} \| x^{[j]} - x^* \|_2 \right) = 0.
\]

However, the featured articles of this work do not provide a thorough convergence analysis of the derived MU algorithms based on the BSUM framework. Hence, Theorem 3.6 provides a promising basis for future research directions and makes it possible to check the MU algorithms for convergence in the papers of Part II.

### 3.2.2. Proximal Alternating Linearized Minimization

A rather recent development regarding iterative algorithms for NMF problems are so-called Proximal Alternating Linearized Minimization (PALM) schemes [6, 45, 56, 44, 16, 1], which play a major role in the featured research article [18].
3.2. Iterative Algorithms

Before introducing the objective function, we need to define the notion of a proper and a lower semi-continuous functions.

**Definition 3.7 (Proper Function [50])**. A function $F : \mathbb{R}^n \to \overline{\mathbb{R}} := [-\infty, \infty]$ is called *proper*, if

(i) $F(x) < \infty$ for at least one $x \in \mathbb{R}^n$,

(ii) $F(x) > -\infty$ for all $x \in \mathbb{R}^n$.

**Definition 3.8 (Lower Semi-continuity [50])**. A function $F : \mathbb{R}^n \to \mathbb{R}$ is called *lower semi-continuous at* $\bar{x}$, if $\lim \inf_{x \to \bar{x}} F(x) \geq F(\bar{x})$ and *lower semi-continuous (on)* $\mathbb{R}^n$, if this holds for all $x \in \mathbb{R}^n$.

For reasons of clarity, we consider first of all the minimization with respect to two variable blocks. The corresponding objective function, which is typically considered for PALM schemes, is of the form

$$\min_{x \in \mathbb{R}^{m_1}, y \in \mathbb{R}^{m_2}} \{G(x, y) := D(x, y) + R_1(x) + R_2(y)\}, \tag{3.11}$$

where $D : \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \to \mathbb{R}$ usually denotes the discrepancy term with a finite-sum structure given by

$$D(x, y) := \frac{1}{N} \sum_{n=1}^{N} D_n(x, y) \tag{3.12}$$

for $D_n : \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \to \mathbb{R}$. Such finite-sum structures often arise in image processing application and NMF problems, which will be further elaborated later in Section 3.2.2.3. However, we will see that this structural assumption is only needed for a stochastic extension of the PALM scheme (SPRING), which will be introduced in Section 3.2.2.1.

Furthermore, the functions $R_1 : \mathbb{R}^{m_1} \to \mathbb{R} \cup \{+\infty\}$ as well as $R_2 : \mathbb{R}^{m_2} \to \mathbb{R} \cup \{+\infty\}$ in (3.11) are used as regularization terms. The following conditions are imposed on the functions for the application of the PALM method (see [6]):

(P1) The functions $G, R_1$ and $R_2$ are bounded from below.

(P2) The function $D$ is finite-valued, differentiable and the gradient $\nabla D$ is Lipschitz continuous on bounded subsets of $\mathbb{R}^{m_1} \times \mathbb{R}^{m_2}$, i.e. for each bounded subset $B_1 \times B_2 \subset \mathbb{R}^{m_1} \times \mathbb{R}^{m_2}$, there exists $M > 0$ such that for all $(x_i, y_i) \in B_1 \times B_2$, $i \in \{1, 2\}$, it holds

$$\|\nabla_x D(x_1, y_1) - \nabla_x D(x_2, y_2), \nabla_y D(x_1, y_1) - \nabla_y D(x_2, y_2)\|_2 \leq M\|(x_1 - x_2, y_1 - y_2)\|_2.$$
(P3) The functions $R_1$ and $R_2$ are proper and lower semi-continuous.

(P4) The partial gradient $\nabla_x D$ is Lipschitz continuous with modulus $L_1(y)$, i.e. for a fixed $y \in \mathbb{R}^{m_2}$, there exists an $L_1(y) > 0$ such that
\[
\|\nabla_x D(x_1, y) - \nabla_x D(x_2, y)\|_2 \leq L_1(y)\|x_1 - x_2\|_2 \quad \forall x_1, x_2 \in \mathbb{R}^{m_1}.
\]
The same holds true for the partial gradient $\nabla_y D$ with the corresponding modulus $L_2(x)$.

(P5) For $i \in \{1, 2\}$ there exist $\lambda^-_i, \lambda^+_i > 0$ such that
\[
\begin{align*}
\inf\{L_1(y[j]) | j \in \mathbb{N}\} & \geq \lambda^-_1, & \inf\{L_2(x[j]) | j \in \mathbb{N}\} & \geq \lambda^-_2, \\
\sup\{L_1(y[j]) | j \in \mathbb{N}\} & \leq \lambda^+_1, & \sup\{L_2(x[j]) | j \in \mathbb{N}\} & \leq \lambda^+_2,
\end{align*}
\tag{3.13}
\]
\[
\begin{align*}
\sup\{L_1(y[j]) | j \in \mathbb{N}\} & \leq \lambda^+_1, & \sup\{L_2(x[j]) | j \in \mathbb{N}\} & \leq \lambda^+_2.
\end{align*}
\tag{3.14}
\]
where $j$ denotes the iteration index of the PALM algorithm.

We note that the technical assumption in (P5) can be omitted under slightly harder conditions, since the inequalities in (3.13) can be obtained based on the assumption in (P4). Furthermore, the inequalities in (3.14) are fulfilled if the function $D$ is $C^2$ and if the generated sequence $\{(x[j], y[j])\}_{j \in \mathbb{N}}$ by PALM is bounded.

We can also see that that no convexity assumption is used throughout (P1)–(P5). In addition, no differentiability of the functions $R_1$ and $R_2$ is needed allowing the use of non-differentiable regularization terms for this optimization method, which is different from the classical MM framework together with the QUBP discussed in Section 3.2.1. Furthermore, the numerical evaluation of the featured article [18], where different PALM algorithms were applied for ONMF models, shows that the PALM scheme has a faster convergence speed and performance stability in practice than the considered MU algorithms. Moreover, as we will discuss in Section 3.2.2.2, the PALM framework allows convergence guarantees for a wide range of objective functions. However, the derivation of algorithms based on PALM might not be as flexible as in the case of the MM framework for the MU algorithms, since the proximal operator with respect to the considered regularization terms has to be computed or be known a priori, which may restricts the use of a complex combination of penalty terms.

The PALM algorithm is based on an alternating minimization of the variables $x$ and $y$. For each variable, the main structure of the update step includes a gradient descent step with respect to the discrepancy term $D$ followed by the application of the proximal operator with respect to the corresponding penalty term $R_i$ for some $i \in \{1, 2\}$. Note
that due to the use of possible non-convex regularization terms, the proximal mapping is
given by the fundamental Moreau proximal map and its corresponding Moreau envelope
function [50, 6]. For a proper and lower semi-continuous function $J : \mathbb{R}^n \rightarrow (-\infty, \infty]$ 
with $\inf J > -\infty$ and a given $t \in (0, \infty)$, the set-valued proximal mapping $\text{prox}^J_t$ and
the corresponding Moreau envelope $m^J_t$ are defined by
\[
\text{prox}^J_t(x) := \arg \min_{y \in \mathbb{R}^n} \left\{ \frac{1}{2t} \|y - x\|^2_2 + J(y) \right\},
\]
\[
m^J_t(x) := \inf_{y \in \mathbb{R}^n} \left\{ \frac{1}{2t} \|y - x\|^2_2 + J(y) \right\} \leq J(x).
\]
Both the proximal mapping and the Moreau envelope are well-defined as long as $J$
fulfills the above conditions [50, 6].

Remark 3.9. For a proper and lower semi-continuous function $J : \mathbb{R}^n \rightarrow (-\infty, \infty]$ 
with $\inf J > -\infty$ and a given $x \in \mathbb{R}^n, t \in (0, \infty)$, the set $\text{prox}^J_t(x)$ is nonempty and compact,
$m^J_t(x)$ is finite and $m^J_t$ is continuous in $(x, t)$.

In this work and especially regarding the analyzed NMF models in the presented
article [18] in Part II, no non-convex regularization parameters are considered leading to
single-valued and continuous proximal mappings [50], so that the above generalization
of set valued functions is not needed. For more information on the properties of the
generalized proximal mappings, we refer the reader to [50]. However, the possibility to
consider non-differentiable regularization terms is exploited in [18] by analyzing total
variation regularization terms in the NMF models.

In the case of the minimization of the cost function $G$ in (3.11) with respect to $x$ for
a given $y \in \mathbb{R}^{m_2}$ via the PALM algorithm, this leads to the update rule
\[
x^{[i+1]} \in \text{prox}^{R_1}_{\eta_{x[i]}} \left( x^{[i]} - \eta_{x[i]} \nabla_x D \left( x^{[i]}, y^{[i]} \right) \right),
\]
where $\eta_{x[i]}$ is the step size of the gradient descent step. Following [6] to ensure conver-
gegence of the algorithm, $\eta_{x[i]}$ is chosen based on the Lipschitz constant $L_1(y^{[i]})$ and
is given by $\eta_{x[i]} = 1/((\gamma_1 L_1(y^{[i]}))$ for a $\gamma_1 > 1$. The update step for $y^{[i+1]}$ is structured anal-
ogously together with the step size $\eta_{y[i]} = 1/((\gamma_2 L_2(x^{[i+1]})))$. The whole PALM scheme
for the minimization problem in (3.11) is presented in Algorithm 1. The algorithm
terminates until a suitable stopping criterion is satisfied. Furthermore, note that it is
needed to compute the Lipschitz constants $L_1(y^{[i]})$ and $L_2(x^{[i+1]})$ for each iteration
step.
Algorithm 1 PALM for solving (3.11)

1: Initialize $x^{[0]} \in \mathbb{R}^{m_1}$, $y^{[0]} \in \mathbb{R}^{m_2}$, $\gamma_1, \gamma_2 > 0$, $i = 0$
2: repeat
3: $\eta_{x[i]} = 1/(\gamma_1 L_1(y^{[i]}))$
4: $x^{[i+1]} \in \text{prox}_{R_1}\eta_{x[i]}(x^{[i]} - \eta_{x[i]} \nabla_x D(x^{[i]}, y^{[i]}))$
5: $\eta_{y[i]} = 1/(\gamma_2 L_2(x^{[i+1]}))$
6: $y^{[i+1]} \in \text{prox}_{R_2}\eta_{y[i]}(y^{[i]} - \eta_{y[i]} \nabla_y D(x^{[i+1]}, y^{[i]}))$
7: $i \leftarrow i + 1$
8: until Stopping criterion satisfied

3.2.2.1. Possible Extensions of PALM

There are many possible extensions of the PALM workflow presented above throughout the literature [6, 45, 44, 16, 1]. However, to stay within the scope of this thesis, only the extensions of the PALM algorithm which are considered in the featured article [18] will be shortly described in more detail.

One important extension of the classical PALM workflow is the possibility to consider more than 2 variable blocks leading to minimization problems of the form

$$
\min_{x_1 \in \mathbb{R}^{m_1}, \ldots, x_p \in \mathbb{R}^{m_p}} \left\{ G(x_1, \ldots, x_p) := D(x_1, \ldots, x_p) + \sum_{j=1}^{p} R_j(x_j) \right\} \tag{3.15}
$$

with $p > 2$ blocks and the analogous assumptions on the functions $G, D$ and $R_j$ as described in (P1)–(P5). According to [6], the convergence theory of the PALM scheme easily extends to this generalized objective function (see also Section 3.2.2.2). Furthermore, we will see that this extension will be very useful to apply the PALM algorithm to ONMF problems.

To improve the performance of the classical PALM algorithm, the authors in [45] developed the so-called inertial PALM (iPALM) algorithm, which introduces an additional momentum term. Since it still follows the rough outline of Algorithm 1, we will not describe the full algorithm and refer the reader to [45]. While no theoretical improvement of the convergence rate could be proven, the authors show an increased practical performance compared to PALM.

Finally, [16] introduces the so-called Stochastic Proximal Alternating Linearized Minimization (SPRING) method. This approach is a stochastic extension of the classical PALM scheme and replaces the full partial gradients of the gradient descent steps in Algorithm 1 by random estimates by exploiting the finite-sum structure described in
Equation (3.12).

The main motivation for this approach is based on the fact that for large \( N \) in (3.12), the computation of the full partial gradients of \( D \) including all the \( D_n \) can be computationally expensive leading to ineffective minimization algorithms. The SPRING approach tries to reduce the computational complexity for each iteration by considering random gradient estimates, which are formed by just using a few indices \( n \in B_{i,\ell} \subset \{1, \ldots, N\} \), where \( B_{i,\ell} \) denotes the mini-batch with \( i \) being the iteration number of SPRING and \( \ell \in \{1, \ldots, s_r\} \) the index specifying the currently used subsample of indices with \( 1/s_r \) being the subsample ratio.

Many different gradient estimates exist throughout the literature. One classical example, which is also used in the featured article [18], is the so-called Stochastic Gradient Descent (SGD) estimator given by

\[
\tilde{\nabla}_x D(x, y) = \frac{1}{|B_{i,\ell}|} \sum_{n \in B_{i,\ell}} \nabla_x D_n(x, y).
\]

For other possible gradient estimates like the so-called SAGA and SARAH gradient estimator, we refer to the work [16] and the references therein. The whole SPRING algorithm for the minimization problem in (3.11) is depicted in Algorithm 2.

**Algorithm 2** SPRING for solving (3.11)

1: Initialize \( x^{[0]} \in \mathbb{R}^{m_1}, \ y^{[0]} \in \mathbb{R}^{m_2}, \ s_r \in \mathbb{N}, \ i = 0 \)
2: repeat
3: Sample mini-batches \( B_{i,\ell} \) for all \( \ell \in \{1, \ldots, s_r\} \), which are chosen uniformly at random from all subsets of \( \{1, \ldots, N\} \) with cardinality \( [N/s_r] \).
4: for \( \ell = 1, \ldots, s_r \) do
5: Compute \( \eta_{x,[i,\ell]} \)
6: \( x^{[i,\ell+1]} \in \text{prox}_{\eta_{x,[i,\ell]}} \left( x^{[i,\ell]} - \eta_{x,[i,\ell]} \tilde{\nabla}_x D(x^{[i,\ell]}, y^{[i,\ell]}) \right) \)
7: Compute \( \eta_{y,[i,\ell]} \)
8: \( y^{[i,\ell+1]} \in \text{prox}_{\eta_{y,[i,\ell]}} \left( y^{[i,\ell]} - \eta_{y,[i,\ell]} \tilde{\nabla}_y D(x^{[i,\ell+1]}, y^{[i,\ell]}) \right) \)
9: end for
10: \( x^{[i+1,1]} = x^{[i,s_r+1]}, \ y^{[i+1,1]} = y^{[i,s_r+1]} \)
11: \( i \leftarrow i + 1 \)
12: until Stopping criterion satisfied

larily, the step sizes \( \eta_{x,[i,\ell]} \) and \( \eta_{y,[i,\ell]} \) are chosen based on the Lipschitz constants of the stochastic estimates of the gradients. For more information, we refer the reader to [16] and to the featured article [18] in Part II.
3.2.2.2. Convergence Theory

The convergence properties of the PALM scheme are well-known throughout the literature and are heavily dependent on the properties of the considered objective function. Following the theoretical convergence results in [6] of the classical PALM method and considering the minimization problem in (3.11), the so-called Kurdyka-Łojasiewicz (KL) property of the function \( G \) plays a major role. To define this property properly, we need the concept of the subdifferential of a proper and lower semi-continuous function, which is in general non-convex [41, 6].

**Definition 3.10** (Subdifferential [6]). Let \( F : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\} \) be a proper and lower semi-continuous function.

(i) For a given \( x \in \text{dom} \ F := \{ y \in \mathbb{R}^n \mid F(y) < +\infty \} \), we define the Fréchet subdifferential \( \hat{\partial} F(x) \) of \( F \) at \( x \) as the set of all \( w \in \mathbb{R}^n \), which fulfill

\[
\liminf_{y \neq x \to x} \frac{F(y) - F(x) - \langle w, y - x \rangle}{\| y - x \|_2} \geq 0.
\]

For the case \( x \notin \text{dom} \ F \), we set \( \hat{\partial} F(x) = \emptyset \).

(ii) We define the limiting-subdifferential \( \partial F(x) \) of \( F \) at \( x \in \mathbb{R}^n \) as

\[
\partial F(x) := \left\{ y \in \mathbb{R}^n \mid \exists x_k \to x : F(x_k) \to F(x), \ \hat{\partial} F(x_k) \ni y_k \to y \text{ as } k \to \infty \right\}.
\]

(iii) The point \( x^* \) is called a critical point of \( F \), if \( 0 \in \partial F(x^*) \).

Furthermore, we denote by \( C_\eta \) for \( \eta \in (0, \infty) \) the set of all concave and continuous functions \( \varphi : [0, \eta) \to \mathbb{R}_{\geq 0} \), so that \( \varphi \in C^1 \) on \((0, \eta)\), continuous at 0, \( \varphi(0) = 0 \) and that for all \( t \in (0, \eta) \), it holds \( \varphi'(t) > 0 \). With this, we are able to define the KL property in the following.

**Definition 3.11** (Kurdyka-Łojasiewicz Functions [6]). Let \( F : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\} \) be a proper and lower semi-continuous function.

(i) \( F \) has the Kurdyka-Łojasiewicz (KL) property at \( \pi \in \text{dom} \partial F := \{ x \in \mathbb{R}^n \mid \partial F(x) \neq \emptyset \} \), if there exist \( \eta \in (0, \infty) \), a neighbourhood \( \mathcal{X} \subset \mathbb{R}^n \) of \( \pi \) and a function \( \varphi \in C_\eta \), such that for all

\[
x \in \mathcal{X} \cap \{ y \in \mathbb{R}^n \mid F(\pi) < F(y) < F(\pi) + \eta \},
\]
it holds
\[ \varphi'(F(x) - F(\bar{x})) \cdot \inf\{\|y\|_2 \mid y \in \partial F(x)\} \geq 1. \]

(ii) \( F \) is a KL function, if it satisfies the KL property for all \( x \in \text{dom} \partial F \).

While this property is rather technical, it is a major property for the convergence analysis of the PALM algorithm and leads, if further analyzed, to a geometrical feature, which has profound consequences in the analysis of first-order descent methods (see [6] and the references therein).

Regarding the convergence analysis of the classical PALM scheme described in Algorithm 1, the following can be shown [6].

**Theorem 3.12 (Convergence of PALM [6]).** Let the function \( G \) be given as defined in the minimization problem (3.11). Furthermore, let \( G \) be a KL function, so that the Assumptions (P1)-(P5) hold. Finally, let \( \{z_i\}_{i \in \mathbb{N}} \) with \( z_i := (x_i, y_i) \in \mathbb{R}^m \) be a bounded sequence generated by PALM in Algorithm 1 for some \( m \in \mathbb{N} \). Then, the sequence \( \{z_i\}_{i \in \mathbb{N}} \) converges to a critical point \( z^* := (x^*, y^*) \) of \( G \).

Note that the above convergence result can be also extended to \( p > 2 \) variable blocks [6] for the corresponding minimization problem in (3.15), which is also relevant for the considered ONMF model in the featured article [18].

The question remains how to check the technical KL property for a considered objective function. However, KL functions are ubiquitous across many applications. Furthermore, there exist strong relationships between KL functions and other classes of functions, such as semi-algebraic or subanalytic functions.

In the following, we shortly describe the connection between semi-algebraic functions and KL functions, since this plays a major role in the short convergence analysis of the PALM algorithms in the featured article [18]. First, we define the notion of a semi-algebraic function, which is taken from [6].

**Definition 3.13 (Semi-algebraic Sets and Functions [6]).**

(i) A subset \( S \subset \mathbb{R}^n \) is called a semi-algebraic set, if there exists a finite number of polynomial functions \( \varphi_{ij}, \psi_{ij} : \mathbb{R}^n \to \mathbb{R} \) for \( j \in \{1, \ldots, p\}, i \in \{1, \ldots, q\} \), so that
\[
S = \bigcup_{j=1}^p \bigcap_{i=1}^q \{x \in \mathbb{R}^n \mid \varphi_{ij}(x) = 0 \text{ and } \psi_{ij}(x) < 0\}.
\]

(ii) A function \( F : \mathbb{R}^n \to (-\infty, +\infty] \) is called semi-algebraic if its graph \( \{(x, t) \in \mathbb{R}^{n+1} \mid F(x) = t\} \) is a semi-algebraic subset of \( \mathbb{R}^{n+1} \).
The following property of KL functions is taken from [6] and helps in the work [18] of Part II to check the KL property of the considered NMF objective functions.

**Lemma 3.14.** Let $F : \mathbb{R}^n \to (-\infty, +\infty]$ be a proper and lower semi-continuous function. If $F$ is semi-algebraic, then $F$ satisfies the KL property at any point of dom $F$.

The class of semi-algebraic functions include, inter alia, real polynomial functions and is closed under finite sums, products and compositions of semi-algebraic functions [6]. Hence, it covers a wide range of possible objective functions in many applications and provides an intuitive method to identify KL functions, which is also used in the work [18] of Part II with the considered ONMF models (see also Section 3.2.2.3).

Furthermore, we note that further results on convergence rates of the PALM algorithm exist for the case of a semi-algebraic function $G$ and are based on the analysis of its so-called desingularizing function. However, since the featured work [18] does not contain any analysis on convergence rates, we refer the reader for more information to [6].

For iPALM, similar convergence results as in Theorem 3.12 can be obtained [45]. For the same conditions as in Theorem 3.12 with additional assumptions on the parameters of the momentum terms, the convergence of the iPALM iterates to a critical point of $G$ is ensured.

Finally, further similar convergence results for the stochastic extension SPRING are obtained in [16] for the choice of the SAGA or SARAH gradient estimator. However, since the featured work [18] only consider the classical SGD estimator for which no convergence guarantees are ensured by [16], we omit the corresponding details at this point.

### 3.2.2.3. Application to NMF

A classical application field for optimization methods based on PALM are NMF problems [6, 45, 16, 18]. The authors in [6] apply their proposed classical PALM scheme to an NMF problem with an $\ell_0$ sparsity regularization term, which is nonconvex and nondifferentiable. However, by using the PALM minimization procedure and by showing the needed properties of the NMF cost function, they derive the update rules by computing the needed gradients as well as the proximal operator of the regularization term and show the convergence of the iterates to a stationary point of the cost function.

In the presented article [18] of Part II, the ONMF model in (2.14) is considered along with a nondifferentiable total variation regularization, for which the minimization scheme based on PALM, iPALM and SPRING with the SGD estimator is analyzed.
To ensure the needed properties for convergence, an additional auxiliary variable is introduced to avoid fourth order terms in the objective function and to ensure the Lipschitz continuity of the partial gradients of $F$. Hence, the extended version of the PALM scheme with $p = 3$ variable blocks is considered. For the derivation of the update rules, the corresponding gradients of the occurring functions are calculated as well as the needed Lipschitz constants, which are estimated based on a power method as it has been done in [16]. Furthermore, a short convergence analysis of the considered PALM methods is provided, where the Assumptions (P1)-(P5) as well as the KL property of the whole objective function is analyzed. Based on these findings, the convergence of the PALM and iPALM algorithm to a stationary point of the cost function is obtained. For SPRING, the SGD estimator is used for which [16] does not provide any convergence guarantees. Hence, we note that the claimed convergence result of the SPRING iterates in [18] cannot be guaranteed. Finally, a thorough numerical evaluation of comparative as well as the proposed algorithms on a hyperspectral imaging dataset is performed, where the PALM algorithms achieved the best clustering outcomes and performance stability.

3.2.3. Initialization Methods and Stopping Criteria

Regarding the description of the above multiplicative update rules and the PALM scheme, the question remains on how to compute suitable initial matrices and to set appropriate stopping criteria for the iterative algorithms.

The initialization method of the matrices plays a key role for the performance of the algorithms. Poor initializations oftentimes lead in practice to a poor convergence of the iterates or even to incorrect solutions. Furthermore, the usual NMF problem is non-convex in both variables leading to different local minima, so that the iterative algorithms presented above are very sensitive to the used initialization techniques. Finally, the computation of appropriate initial matrices can become more complicated if constraints on the matrices are included in the NMF problem.

A good overview on possible initialization strategies is provided by the two books on NMF and Nonnegative Tensor Factorizations (NTF) [10, 23]. One classical approach, which is still used throughout the literature as a standard procedure, is the random initialization of the matrices, where the entries are drawn from the interval $[0, 1]$ based on a uniform distribution. This straightforward procedure may have a comparatively low computational cost, but more sophisticated initialization methods, which make use of the given data matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, oftentimes allows to identify better local minima of the objective function. Furthermore, the initial matrices thus obtained are usually
closer to a suitable solution of the problem, so that the considered algorithm is able to converge faster.

One common approach follows the results of the papers [26, 7], which is also mostly used in the research articles of Part II [20, 3, 18, 37]. The initialization method is based on a truncated Singular Value Decomposition (tSVD) of the data matrix following a Krylov method [26] with a subsequent normalization as well as projection steps to ensure the nonnegativity of the matrices [7]. In the presented research article [18], where ONMF models are introduced for clustering applications, further initialization techniques based on K-means++ are considered to take into account the additional orthogonality contraint of the matrix $U$. Note that for multiplicative algorithms, it is recommended to perform additional projection steps to strictly positive matrices $U$ and $V$ to avoid the zero locking phenomenon described in Section 3.2.1.

As for any other iterative optimization method, a suitable stopping criterion has to be chosen for the considered algorithms. In the following, we outline some of the common strategies. A very common technique is to set an upper bound for the number of iterations, which is usually between 200 and 5000 [23] and used in many of the featured research articles [20, 3, 18, 37].

A further method, which is also used in the presented research articles [3, 37], monitors the relative error of the iterates $U^{[i]}$ and $V^{[i]}$. In the case of $U^{[i]}$, the stopping criterion can be written as

$$\|U^{[i]} - U^{[i-i^*]}\|_F \leq \varepsilon \|U^{[i-i^*]}\|_F,$$

for some $i^* \in \mathbb{N}$ and a small $\varepsilon > 0$. Typical choices are $i^* = 10$ and $\varepsilon = 10^{-4}$ in practice [23]. A similar stopping criterion can be obtained by monitoring the relative error of the objective function. However, the computation of the value of the objective function can lead to additional computational cost. Independent from the above described stopping criteria, it is also recommended to set another upper bound for the computational time to ensure that the algorithm will stop in a reasonable amount of time [23].

For more information on possible initialization methods and stopping criteria, we refer the reader to [10, 23] and the references therein.
4 | Applications

This chapter gives a short introduction to the two main application fields discussed in the featured research articles of Part II. Section 4.1 introduces the Matrix-Assisted Laser Desorption/Ionization (MALDI) imaging technique and motivates the application of NMF models to this mass spectrometry imaging method. Three of the featured articles [20, 18, 36] use MALDI imaging data for the numerical evaluation of several considered NMF models.

Section 4.2 gives a short overview of Computed Tomography (CT) with a focus on the specific case of dynamic CT. The corresponding basic principles on dynamic inverse problems can be found in Section 2.1. In the featured paper [3], the considered NMF models are designed for this specific application case to develop a joint reconstruction and low-rank decomposition method for dynamic inverse problems and are evaluated on three simulated phantoms.

4.1. MALDI Imaging

The so-called Matrix Assisted Laser Desorption/Ionization imaging, first described in 1994 [54], belongs to the big family of Mass Spectrometry Imaging (MSI) methods and allows to provide a spatial molecular profile of a given analyte. Several of the featured articles in Part II of this thesis apply specifically designed NMF models to MALDI Imaging for data representation, clustering or classification [20, 18, 36]. In all these research articles, a short introduction to MALDI imaging is provided. To avoid repetitions of this topic, we only provide a short description of this application in this section.

The basic measurement process with a mass spectrometer can usually be subdivided into three main steps: the ionization of the analyte, the separation of the ions with respect to their mass-to-charge ratio (m/z-value) and the final detection of the separated particles.

The ionization of the material is fundamental for the whole measurement process and allows, in the case of a typical Time-Of-Flight (TOF) mass spectrometer, to accelerate
Figure 4.1.: Figure 4.1a shows an illustration of the typical structure of a MALDI dataset with three obtained mass spectra and a selection of three channels (slightly changed from the original image in the featured article [20]). Figure 4.1b illustrates the basic MALDI ionization process.

the extracted ions via an electric field so that all particles have the same kinetic energy after the acceleration phase leading to different velocities for different m/z-values of the ions. After the acceleration phase, the particles travel a certain distance force-free until they reach the detector, which allows to determine their specific m/z-value via the equations of energy conservation.

The abbreviation MALDI refers to the used ionization method, which allows the measurement of a wider mass range of the extracted particles compared to other ionization techniques like the so-called Desorption Electrospray Ionization (DESI) or the Secondary-Ion Mass Spectrometry (SIMS) [2]. This is possible due to the application of a matrix on the analyte, which allows the extraction of the molecules from the analyte and an optimized transfer of the ionization energy (see also Figure 4.1b). The latter is provided by a pulsed laser with pulse durations of 1 ns to 100 ns to prevent a too strong fragmentation of the particles due to the thermal energy. Another major part of the whole process is the preparation of the analyte before the actual measurement, which follows standardized protocols. For more information on typically used matrices, the sample preparation and MALDI imaging in general, we refer the reader to [29].

The laser shoots the analyte on multiple points of the analyte by following a grid pattern and acquires for each point a whole mass spectrum (see Figure 4.1a). Hence, the whole MALDI dataset can be interpreted as a collection of mass spectra for each spatial position of the analyte or as a collection of intensity maps for each m/z-value, which are usually referred to as channels or m/z-images. The whole dataset is typically stored in
a matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, where each row corresponds to a mass spectrum and each column to a channel for a specific m/z-value. Typical data sizes of MALDI data range from $10^4$ to one million spectra and m/z-images respectively. Furthermore, note that MALDI datamatrices are naturally nonnegative, since the entries are based on the number of the detected ions. This property makes the NMF an ideal tool for analyzing MALDI data by decomposing the data matrix into two matrices $U \in \mathbb{R}^{M \times K}_{\geq 0}$ and $V \in \mathbb{R}^{K \times N}_{\geq 0}$ with $X \approx UV$. Together with the assumption, that each measured spectra can be represented by a nonnegative superposition of a few characteristic spectra, the NMF is able to extract these spectra within the rows of $V$ along with the corresponding pseudo channels in $U_{:, k}$ making the NMF an ideal tool for data representation as shown in the featured article [20]. Furthermore, [18] designs spatially coherent clustering models based on ONMF by including a TV regularization term and applies them to labelled MALDI imaging datasets. Finally, [36] introduces supervised NMF models as a prior feature extraction step for the classification of tumor subtypes and the extraction of corresponding biomarkers.

4.2. Dynamic Computed Tomography

Computed Tomography or X-ray Tomography is a classical inverse problem and can be found in the beginning of many introductory books on linear inverse problems as an initial application example [39, 49, 31]. Similar to MALDI Imaging in Section 4.1, it belongs to the big field of medical imaging applications. However, different from MALDI Imaging, CT goes back to at least 1917 with the introduction of the Radon transform and allows insights into the human body without any surgical procedures.

The basic principle is as follows: Multiple, thin beams of X-rays are shot through the human body and the number of outgoing photons for each X-ray are counted with a detector. In such a way, it is possible to measure how much the intensity of the X-ray is attenuated by going through the tissue. Due to the different densities of the various tissue types, the incoming X-rays are absorbed in different intensities. If there are sufficiently enough X-ray beams, which are equally distributed throughout a two-dimensional $\mathbf{x} = (x_1, x_2)$ plane, it is possible to reconstruct the density distribution of the human body within this plane. Hence, the inverse problems of CT consists of determining the density distribution of the human body based on the attenuation of the X-rays by going through the tissue.

In the following, we derive shortly the mathematical model of the classical CT problem in two dimensions with $\mathbf{x} := (x_1, x_2)$. First, we parametrize the straight X-rays
4. Applications

\[ X(x) = (x_1, x_2) \]

\[ Y(\theta, s_1, t) \]

\[ Y(\theta, s_2, t) \]

\[ x = (x_1, x_2) \]

\[ Y(\theta, s_1, t) \]

\[ Y(\theta, s_2, t) \]

\[ R \langle x, \theta \rangle = s \]

\[ \phi_1 \]

\[ \phi_2 \]

\[ (a) \text{ Detailed Projection} \]

\[ (b) \text{ Two Projections} \]

**Figure 4.2.** Figure 4.2a shows a detailed illustration of a projection during a dynamic CT measurement for the two-dimensional case. Figure 4.2b illustrates two projections for two different angles $\phi_1$ and $\phi_2$. Both illustrations consider the case of the parallel-beam geometry.

via the Hesse normalform based on the unit normal vector of the X-ray $\theta(\phi) = [\cos(\phi), \sin(\phi)]^\top \in S^1$ and the distance from the origin $s \in \mathbb{R}$, where $S^1$ is the unit circle (see Figure 4.2a). Hence, for a given $\theta \in S^1$ and a distance $s \in \mathbb{R}$, the X-ray can be described by $\langle x, \theta \rangle = s$, where $x$ is the corresponding position vector. Based on the physical laws, it can be derived that the mean of the density function $X(x)$ along one of these lines corresponds to the logarithm of the ratio of the input signal intensity $y_0$ to the output signal intensity $y_1$. This leads to the relationship

\[
\int_{\langle x, \theta \rangle = s} X(x) \, d\sigma(x) = \ln \left( \frac{y_0}{y_1} \right), \quad (4.1)
\]

where the integral on the left side is the line integral along the line $\langle x, \theta \rangle = s$ with the corresponding hypersurface measure $\sigma$. This gives rise to the definition of the so-called Radon transform $R$, which assigns each function $F$ to its line integral and is given by

\[
RF(\theta, s) := \int_{\langle x, \theta \rangle = s} F(x) \, d\sigma(x). \quad (4.2)
\]

Regarding the inverse problem of Equation (4.1) and the reconstruction of the density function $X$, the right hand side is given by noisy measurements for each $s \in \mathbb{R}$ and
\( \theta \in S^1 \) denoted by \( Y(\theta, s) \). Hence, the whole CT problem can be formulated via a short operator equation given by
\[
RX(\theta, s) = Y(\theta, s), \tag{4.3}
\]
where \( Y(\cdot, \cdot) \) are usually referred to as sinograms. As described in Section 2.1, it can be shown that the Radon transform between suitably defined \( L^2 \) spaces is a compact operator and hence leads indeed to an ill-posed inverse problem.

However, the featured article in [3] discusses NMF models which are specifically designed for dynamic inverse problems in general and are applied to the problem of dynamic computed tomography. In this generalized case, the attenuation \( X \) is assumed to be additionally time dependent. This makes it possible to model dynamic processes in the human body and can be helpful to reconstruct spatial motion or to determine the blood flow by injecting a contrast agent to the patient’s blood stream for applications in angiography. The time dependence of \( X(x, t) \) leads in turn to a time dependence of \( Y(\theta, s, t) \) (see Figure 4.2a and 4.2b) and hence to the operator equation
\[
(R_{I(t)}X(x, t))(\theta, s) = \int_{(x, \theta)} X(x, t) d\sigma(x) = Y(\theta, s, t), \tag{4.4}
\]
where \( R_{I(t)} \) denotes the time dependent forward operator, which is the Radon transform with respect to the set of given measurements \( I(t) \ni (\theta, s) \) at time \( t \) with \( I(t) \subset \{(\theta, s) \mid \theta \in S^1, s \in \mathbb{R}\} \) (see also [8] for more details). Hence, for each time \( t \), \( X(\cdot, t) \) corresponds to a reconstruction of the density function based on the sinogram \( Y(\cdot, \cdot, t) \).

Furthermore, a specific number of angles \( \theta \in S^1 \) and distances \( s \in \mathbb{R} \) are measured for each point in time leading to the time dependent Radon transform \( R_{I(t)} \). In the setting considered in the work [3], we assume that the number of angles measured at different points in time remains constant, i.e. \( |I(t)| = c \) in a slight abuse of notation, while still allowing that different angles are measured at different times. This ensures that when discretizing the time dependent Radon transform, the resulting matrix \( R_t \) has the same dimension for all \( t \), which simplifies the analysis of the NMF models. The discretization of Equation (4.4) leads finally to the linear system
\[
R_t X_{\cdot,t} = Y_{\cdot,t} \quad \text{for} \quad 1 \geq t \geq T, \tag{4.5}
\]
with \( R_t \in \mathbb{R}_{\geq 0}^{M \times N} \) being the discretized Radon transform for each point in time \( t \), where \( N \) denotes the number of pixels in the original image and \( M \) is given by the product \( M = |I(t)| n_s \) with the number of detection points \( n_s \). Furthermore, \( X \in \mathbb{R}_{\geq 0}^{N \times T} \) is the discretized attenuation, which contains in its columns \( X_{\cdot,t} \) the CT reconstructions for
each point in time. Finally, $Y \in \mathbb{R}_{\geq 0}^{M \times T}$ denotes the data matrix, which contains the vectorized sinograms in its columns $Y_{\bullet,t}$. Note that due to the definition of the Radon transform by line integrals and the physical conditions of the CT measurements, all the above matrices are nonnegative, which plays a central role regarding the construction of suitable NMF models in the featured work [3]. In this paper, several NMF models are designed leading to a joint reconstruction and low-rank decomposition method for the dynamic CT problem. The basic idea behind this approach is to perform the reconstruction by solving Equation (4.5) and to compute in parallel an NMF of the matrix $X$ to extract spatial and temporal features of the obtained reconstructions. For more information on this workflow, we refer the reader the corresponding research paper [3] presented in Part II of this thesis.
In this work, various application-oriented NMF models were constructed by considering different regularization schemes including standard $\ell_1$ and $\ell_2$ regularization, total variation, orthogonality constraints and further penalty terms to enforce the needed properties of the factorization matrices. While the considered NMF models can be used for a broad range of different applications, the main focus of this work lies on medical imaging technologies including MALDI imaging and dynamic CT. In the case of MALDI imaging, the proposed NMF models were used for data representation [20], clustering [18] and supervised classification schemes [36]. Regarding dynamic CT, several NMF models were designed to introduce a joint reconstruction and low-rank decomposition framework for dynamic inverse problems [3]. For both applications, various datasets were used for numerical evaluations of the considered NMF models, which showed that combined methods, such as the joint reconstruction and low-rank decomposition framework in [3], the spatially coherent clustering models in [18] or the supervised classification approach in [36], typically lead to significantly improved results.

Furthermore, suitable solution algorithms were developed and several convergence results were derived with a focus on PALM schemes and multiplicative update rules based on the majorization-minimization principle. Finally, theoretical studies on the relation between regularized ONMF models and generalized $K$-means approaches were performed [21].

Several further research directions regarding the theory, the algorithm development along with their convergence theory and the application of the considered NMF models could be of interest.

The majority of the NMF models in the presented research papers of Part II need an a priori choice of the hyperparameters including the regularization parameters as well as the factorization rank $K$, which is typically based on heuristic approaches. Hence, a useful extension of the presented framework would be the development of improved methods for making an optimal choice of these parameters. Regarding initial approaches for an optimal choice of the factorization rank, we refer the reader to the featured work [3] and the references therein.
Moreover, a deeper convergence analysis of the proposed NMF algorithms based on
the presented requirements in Section 3.2.1 and 3.2.2 for the multiplicative update rules
as well as the PALM framework constitutes a further interesting research direction.

Furthermore, a more thorough numerical evaluation of the presented NMF models
with different types of datasets from a broad range of application fields is needed
to confirm the obtained performance results. This can be done along with a deeper
analysis of the SPRING optimization framework considered in [18], which resulted in
significantly better results than the classical multiplicative update rules.

Another possible research direction could be the analysis of other discrepancy terms
or more advanced NMF models in general, which are tailored to the considered ap-
lication. In the specific case of the proposed joint reconstruction and low-rank de-
composition model based on NMF for dynamic inverse problems in [3], a combination
with other approaches like morphological motion models [24] could be possible to allow
reconstructions of spatial movements of the target.

Finally, a more theoretical research direction could be the investigation of spatially
coherent clustering models with continuous factorization models in the infinite dimen-
sion space (see [21]). The analysis of first order conditions could lead to connections
to $K$-means clustering models and partial differential equations, whose solutions give
information about the corresponding centroids, clusters and the distance measures.
Bibliography


[17] ESA. *How many stars are there in the Universe?* (accessed on 26 October 2021). European Space Agency Website. url: https://www.esa.int/Science_Exploration/Space_Science/Herschel/How_many_stars_are_there_in_the_Universe.


Part II.

Research Papers
A Survey on Surrogate Approaches to Non-negative Matrix Factorization

Pascal Fernsel1 · Peter Maass1

Received: 16 February 2018 / Accepted: 9 August 2018 / Published online: 12 October 2018
© The Author(s) 2018

Abstract
Motivated by applications in hyperspectral imaging, we investigate methods for approximating a high-dimensional non-negative matrix $Y$ by a product of two lower-dimensional, non-negative matrices $K$ and $X$. This so-called non-negative matrix factorization is based on defining suitable Tikhonov functionals, which combine a discrepancy measure for $Y \approx KX$ with penalty terms for enforcing additional properties of $K$ and $X$. The minimization is based on alternating minimization with respect to $K$ and $X$, where in each iteration step one replaces the original Tikhonov functional by a locally defined surrogate functional. The choice of surrogate functionals is crucial: It should allow a comparatively simple minimization and simultaneously its first-order optimality condition should lead to multiplicative update rules, which automatically preserve non-negativity of the iterates. We review the most standard construction principles for surrogate functionals for Frobenius-norm and Kullback–Leibler discrepancy measures. We extend the known surrogate constructions by a general framework, which allows to add a large variety of penalty terms. The paper finishes by deriving the corresponding alternating minimization schemes explicitly and by applying these methods to MALDI imaging data.

Keywords Non-negative matrix factorization · Multi-parameter regularization · Majorize-minimization algorithms · Imaging mass spectrometry

Mathematics Subject Classification (2010) 15A23 · 68W25 · 65F22

Hans-Georg is always a source of scientific inspiration. This paper is dedicated to him on the occasion of his 70th birthday.

✉ Pascal Fernsel
pfernsel@math.uni-bremen.de

Peter Maass
pmaass@math.uni-bremen.de

1 Center for Industrial Mathematics, University of Bremen, D-28359 Bremen, Germany
# 1 Introduction

Matrix factorization methods for large scale data sets have seen increasing scientific interest recently due to their central role for a large variety of machine learning tasks. The main aim of such approaches is to obtain a low-rank approximation of a typically large data matrix by factorizing it into two smaller matrices. One of the most widely used matrix factorization methods is the principal component analysis (PCA), which uses the singular value decomposition (SVD) of the given data matrix.

In this work, we review the particular case of non-negative matrix factorization (NMF), which is favorable for a range of applications where the data under investigation naturally satisfies a non-negativity constraint. These include dimension reduction, data compression, basis learning, and feature extraction as well as higher level tasks such as classification or clustering [11, 26, 27, 30]. PCA-based approaches without any non-negativity constraints would not lead to satisfactory results in this case since possible negative entries of the computed matrices cannot be easily interpreted for naturally non-negative datasets.

Typically, the NMF problem is formulated as a minimization problem. The corresponding cost function includes a suitable discrepancy term, which measures the difference between the data matrix and the calculated factorization, as well as penalty terms to tackle the non-uniqueness of the NMF, to deal with numerical instabilities but also to provide the matrices with desirable properties depending on the application task. The NMF cost functions are commonly non-convex and require tailored minimization techniques to ensure the minimization but also the non-negativity of the matrix iterates. This leads us to the so-called surrogate minimization approaches, which are also known as majorize-minimization algorithms [18, 23, 36]. Such surrogate methods have been investigated intensively for some of the most interesting discrepancy measures and penalty terms [11, 13, 16, 23, 25, 33, 36]. The idea is to replace the original cost function by a so-called surrogate functional, such that its minimization induces a monotonic decrease of the objective function. It should be constructed in such a way that it is easier to minimize and that the deduced update rules should preserve the non-negativity of the iterates, which typically leads to alternating, multiplicative update rules.

It appears that these constructions are obtained case-by-case employing different analytical approaches and different motivations for their derivation. The purpose of this paper, first of all, is to give a unified approach to surrogate constructions for NMF discrepancy functionals. This general construction principle is then applied to a wide class of functionals obtained by different combinations of divergence measures and penalty terms, thus extending the present state of the art for surrogate-based NMF constructions.

Secondly, one needs to develop minimization schemes for these functionals. Here, we develop concepts for obtaining multiplicative minimization schemes, which automatically preserve non-negativity without the need for further projections.

Finally, we exemplify some characteristic properties of the different functionals with MALDI imaging data, which are particularly high-dimensional and challenging hyperspectral datasets.

The paper is organized as follows. Section 2 introduces the basic definition of the considered NMF problems. Section 3 gives an overview about the theory of surrogate
functionals as well as the construction principles. This is then exemplified in Section 4 for the most important cases of discrepancy terms, namely the Frobenius norm and the Kullback–Leibler divergence as well as for a variety of penalty terms. Section 5 discusses alternating minimization schemes for these general functionals with the aim to obtain non-negativity-preserving, multiplicative iterations. Finally, Section 6 contains numerical results for MALDI imaging data.

1.1 Notation

Throughout this work, we will denote matrices in bold capital Latin or Greek letters (e.g., $Y$, $K$, $\Psi$, $A$) while vectors will be written in small bold Latin or Greek letters (e.g. $c$, $d$, $\beta$, $\zeta$). The entries of matrices and vectors will be indicated in a non-bold format to distinguish between the $i$th entry $x_i$ of a vector $x$ and $n$ different vectors $x_j$ for $j = 1, \ldots, n$. In doing so, we write for the entry of a matrix $M$ in the $i$th row and the $j$th column $M_{ij}$ and the $i$th entry of a vector $x$ the symbol $x_i$. The same holds for an entry of a matrix product: the $ij$th entry of the matrix product $MN$ will be indicated as $(MN)_{ij}$.

Furthermore, we will use a dot notation to indicate rows and columns of matrices. For a matrix $M$, we will write $M_{\bullet,j}$ for the $j$th column and $M_{i,\bullet}$ for the $i$th row of the matrix.

What is more, we will use $\| \cdot \|$ for the usual Euclidean norm, $\|M\|_1 := \sum_{ij} |M_{ij}|$ for the 1-norm, and $\|M\|_F$ for the Frobenius norm of a matrix $M$.

Besides that, we will use equivalently the terms function and functional for a mapping into the real numbers.

Finally, the dimensions of the matrices in the considered NMF problem are reused in this work and will be introduced in the following section.

2 Non-negative Matrix Factorization

Before we introduce the basic NMF problem, we give the following definition to clarify the meaning of a non-negative matrix.

**Definition 1** A matrix $M \in \mathbb{R}^{m \times n}$ is called non-negative if $M \in \mathbb{R}^{m \times n}_{\geq 0}$, where $\mathbb{R}_{\geq 0} := \{ x \in \mathbb{R} : x \geq 0 \}$.

The non-negativity of an arbitrary matrix $M$ will be abbreviated for simplicity as $M \succeq 0$ in the later sections of this work.

The basic NMF problem requires to approximately decompose a given non-negative matrix $Y \in \mathbb{R}^{n \times m}$ into two smaller non-negative matrix factors $K \in \mathbb{R}^{n \times p}_{\geq 0}$ and $X \in \mathbb{R}^{p \times m}_{\geq 0}$, such that $p \ll \min(n,m)$ and

$$Y \approx KX.$$ 

For an interpretation, let us assume that we are given $m$ data vectors $Y_{\bullet,j} \in \mathbb{R}^n$ for $j = 1, \ldots, m$, which are stored column-wise in the matrix $Y$. Similarly for $k = 1, \ldots, p$, we denote by $K_{\bullet,k}$, respectively $X_{k,\bullet}$, the column vectors of $K$, respectively the row vectors of $X$. 

We then obtain the following approximation for the column vectors $Y_{\bullet,j}$ as well as the row vectors $Y_{i,\bullet}$:

$$Y_{\bullet,j} \approx \sum_{k=1}^{p} K_{\bullet,k} X_{kj},$$

$$Y_{i,\bullet} \approx \sum_{k=1}^{p} K_{ik} X_{\bullet,k},$$

$$Y \approx KX = \sum_{k=1}^{p} K_{\bullet,k} X_{\bullet,k}.$$ 

Note that the product $K_{\bullet,k} X_{\bullet,k}$ on the right-hand side of the third equation yields rank-one matrices for every $k$.

By these representations, we can regard the rows $X_{k,\bullet}$ as a low-dimensional set of basis vectors, which are tailored for approximating the high-dimensional data vectors, i.e., NMF solves the task of basis learning with non-negativity constraints.

Following the interpretation given above, we can also regard NMF as a basis for compression. $K$ and $X$ are determined by storing $(n + m) \cdot p$ coefficients, as opposed to $n \cdot m$ coefficients for $Y$. The columns of $K$ can be regarded as characteristic components of the given data set $\{Y_{\bullet,j}\}_{j}$. If these data vectors are input for a classification task, one can use the $p$ correlation values with the column vectors of $K$ as features for constructing the classification scheme.

The standard variational approach for constructing an NMF is to define a suitable discrepancy measure $D(\cdot, \cdot)$ between $Y$ and $KX$ and to minimize the resulting functional. Despite their seemingly simple structure, NMF problems are ill-posed, non-linear, and non-convex, i.e., they require stabilization techniques as well as tailored approaches for minimization. In this paper, we consider discrepancy measures based on divergences [17].

**Definition 2** (Divergence) Let $\Omega$ be an arbitrary set. A divergence $D$ is a map $D : \Omega \times \Omega \rightarrow \mathbb{R}$, which fulfills the following properties:

(i) $D(x, y) \geq 0 \ \forall (x, y) \in \Omega \times \Omega$,

(ii) $D(x, y) = 0 \iff x = y$.

**Definition 3** ($\beta$-divergence) The $\beta$-divergence $d_\beta : \mathbb{R}_{>0} \times \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geq 0}$ for $\beta \in \mathbb{R}$ is defined as

$$d_\beta(x, y) := \begin{cases} \frac{x^\beta}{\beta - 1} + \frac{y^\beta}{\beta - 1} - \frac{xy^\beta}{\beta - 1} & \text{for } \beta \in \mathbb{R} \setminus \{0, 1\}, \\ x \log \left( \frac{x}{y} \right) - x + y & \text{for } \beta = 1, \\ \frac{x}{y} - \log \left( \frac{x}{y} \right) - 1 & \text{for } \beta = 0. \end{cases}$$

Furthermore, we define accordingly $D_\beta : \mathbb{R}_{>0}^{n \times m} \times \mathbb{R}_{>0}^{n \times m} \rightarrow \mathbb{R}$ for arbitrary $m, n \in \mathbb{N}$ as

$$D_\beta(M, N) = \sum_{i=1}^{n} \sum_{j=1}^{m} d_\beta(M_{ij}, N_{ij}).$$
The corresponding matrix divergences are defined componentwise, i.e., $\beta = 2$ yields the Frobenius norm and $\beta = 1$ the Kullback–Leibler divergence.

These discrepancy measures are typically amended by so-called penalty terms for stabilization and for enforcing additional properties such as sparsity or orthogonality. This yields the following general minimization task.

**Definition 4** (NMF minimization problem) For a data matrix $Y \in \mathbb{R}^{n \times m}_{\geq 0}$, we consider the following generalized NMF minimization task

$$\min_{K \geq 0, X \geq 0} D_\beta(Y, KX) + \sum_{\ell=1}^L \alpha_\ell \varphi_\ell(K, X). \quad (1)$$

The functional

$$F(K, X) := D_\beta(Y, KX) + \sum_{\ell=1}^L \alpha_\ell \varphi_\ell(K, X) \quad (2)$$

is called the cost functional. Furthermore, we call

(i) $D_\beta(Y, KX)$ the discrepancy term,

(ii) $\alpha_\ell$ the regularization parameters or weights,

(iii) and $\varphi_\ell(K, X)$ the penalty terms.

The functional in (2) is typically non-convex in $(K, X)$. Hence, algorithms based on alternating minimization with respect to $K$ and $X$ are favorable, i.e.,

$$K^{[d+1]} = \arg \min_{K \geq 0} F(K, X^{[d]}), \quad (3)$$

$$X^{[d+1]} = \arg \min_{X \geq 0} F(K^{[d+1]}, X), \quad (4)$$

where the index $d$ denotes the iteration index of the corresponding matrices.

This yields simpler, often convex restricted problems with respect to either $K$ or $X$. Considering for example the minimization of the NMF functional with Frobenius norm and without any penalty term yields a high-dimensional linear system $K^T Y = K^T K X$, which, however, would need to be solved iteratively.

Instead, the so-called surrogate methods for computing NMF decompositions have been proposed recently and are introduced in the next section. They also consider alternating minimization steps for $K$ and $X$, but they replace the restricted minimization problems in (3) and (4) by simpler minimization tasks, which are obtained by locally replacing $F$ by surrogate functionals for $K$ and $X$ separately.

### 3 Surrogate Functionals

In this section, we discuss general surrogate approaches for minimizing general non-convex functionals, which are then exemplified for specific NMF functionals in later sections.

Let us consider a general functional $F : \Omega \rightarrow \mathbb{R}$ where $\Omega \subset \mathbb{R}^N$ and the minimization problem

$$\min_{x \in \Omega} F(x).$$

We will later add suitable conditions guaranteeing the existence of minimizers or at least the existence of stationary points. Surrogate concepts replace this task by solving a
sequence of comparatively simpler and convex surrogate functionals, which can be mini-
mized efficiently. These methods are also commonly referred to as surrogate minimization
(or maximization) algorithms (SM) or also as MM algorithms, where the first M stands for
majorize and the second M for minimize (see also \[18, 23, 36\]). Such approaches have been
demonstrated to be very useful in many fields of inverse problems, in particular for hyperspectral imaging \[11\], medical imaging applications such as transmission tomography \[13, 14\] and MALDI imaging and tumor typing applications \[26\].

Replacing a non-convex functional by a series of convex problems is the main moti-
vation for such surrogate approaches. However, if constructed appropriately, they can also
be used to replace non-differentiable functionals by a series of differentiable problems and
they can be tailored such that gradient descent methods for minimization yield multiplica-
tive update rules which automatically incorporate non-negativity constraints without further
projections.

From this point on, it is important to note that possible zero denominators during the
derivation of the NMF algorithms as well as in the multiplicative update rules themselves
will not be discussed explicitly throughout this work. Usually, this issue is handled in prac-
tice by adding a small positive constant in the denominator during the iteration scheme.
In fact, the instability of NMF algorithms due to the convergence of some entries in the
matrices to zero has not been sufficiently discussed in the literature and still needs proper
solution techniques. We will not focus on this problem and turn now to the basic definition
and properties of surrogate functionals.

### 3.1 Definitions and Basic Properties

As in \[25\], we use the following definition of a surrogate functional.

**Definition 5** (Surrogate functional) Let $\Omega \subseteq \mathbb{R}^N$ denote an open set and $F : \Omega \to \mathbb{R}$ a
functional defined on $\Omega$. Then, $Q_F : \Omega \times \Omega \to \mathbb{R}$ is called a surrogate functional or a
surrogate for $F$, if it satisfies the following conditions:

(i) $Q_F(x, a) \geq F(x)$ for all $x, a \in \Omega$,
(ii) $Q_F(x, x) = F(x)$ for all $x \in \Omega$.

This is the most basic definition, which does not require any convexity or differentiability
of the functional. However, it already allows to prove that the iteration

$$x^{[d+1]} := \arg \min_{x \in \Omega} Q_F(x, x^{[d]})$$

yields a sequence which monotonically decreases $F$ (see also Fig. 1).

**Lemma 1** (Monotonic decrease by surrogate functionals) Let $\Omega \subseteq \mathbb{R}^N$ denote an open
set, $F : \Omega \to \mathbb{R}$ a given function and $Q_F$ a surrogate functional for $F$. Assume that
arg min$_{x \in \Omega} Q_F(x, a)$ is well defined for all $a \in \Omega$. Define the iterated updates by

$$x^{[d+1]} := \arg \min_{x \in \Omega} Q_F(x, x^{[d]})$$

with $x^{[0]} := \arg \min_{x \in \Omega} Q_F(x, a)$ for an arbitrary $a \in \Omega$. Then, $F(x^{[d]})$ is a monotonically
decreasing sequence, i.e.,

$$F(x^{[d+1]}) \leq F(x^{[d]}).$$
Proof The monotone decrease (6) follows directly from the defining properties of surrogate functionals, see Definition 5: We obtain
\[ F(x^{[d+1]}) \leq Q_F(x^{[d+1]}, x^{[d]}) \leq Q_F(x^{[d]}, x^{[d]}) = F(x^{[d]}), \]
where (⋆) follows from the definition of \( x^{[d+1]} \) in (5).

Remark 1 (Addition of surrogate functionals) Let \( \Omega \subseteq \mathbb{R}^n \) be an open set, \( F, G : \Omega \rightarrow \mathbb{R} \) pointwise defined functionals and \( Q_F, Q_G \) corresponding surrogates. Then, \( Q_F + Q_G \) is a surrogate functional for \( F + G \).

For each functional \( F \), there typically exist a large variety of surrogate functionals and we can aim at optimizing their structure. The following additional property is the key to simple and efficient minimization schemes for surrogate functionals.

Definition 6 (Separability of a surrogate functional) Let \( \Omega \subseteq \mathbb{R}^N \) denote an open set, \( F : \Omega \rightarrow \mathbb{R} \) a functional and \( Q_F \) a surrogate for \( F \). The surrogate \( Q_F \) is called separable, if there exist functions \( g_i : \mathbb{R} \times \Omega \rightarrow \mathbb{R} \), such that
\[ Q_F(x, a) = \sum_{i=1}^{N} g_i(x_i, a) \quad \forall x, a \in \Omega. \]

Lemma 1 above only ensures the monotonic decrease of the cost functional, which is not sufficient to guarantee convergence of the sequence \( \{x^{[d]}\} \) to a minimizer of \( F \) or at least to a stationary point of \( F \). The convergence theory for surrogate functionals is far from complete (see also the works [23] and [36]).

Despite this lack of theoretical foundation, surrogate-based minimization yields strictly decreasing sequences for a large variety of applications. In particular, surrogate-based methods can be constructed such that first-order optimality conditions lead to multiplicative update rules, which—in view of the NMF—is a very desirable property.

We now turn to discussing three different construction principles for surrogate functionals.

![Fig. 1 Visualization of the surrogate principle for non-convex F with convex surrogate functional Q_F according to Lemma 1](image)
3.2 Jensen’s Inequality

The starting point is the well-known Jensen’s inequality for convex functions (see [10]).

**Lemma 2** (Jensen’s inequality) Let $\Omega \subseteq \mathbb{R}^N$ denote a convex set, $F : \Omega \to \mathbb{R}$ a convex function and $\lambda_i \in [0, 1]$ non-negative numbers for $i \in \{1, \ldots, k\}$ with $\sum_{i=1}^k \lambda_i = 1$. Then, for all $x_i \in \Omega$, it holds that

$$F \left( \sum_{i=1}^k \lambda_i x_i \right) \leq \sum_{i=1}^k \lambda_i F(x_i).$$

In this subsection, we consider functionals $F$ which are derived from continuously differentiable and convex functions $f : \mathbb{R}_{>0} \to \mathbb{R}$ via

$$F : \Omega \to \mathbb{R}, \quad v \mapsto f(c^T v)$$

for $\Omega \subseteq \mathbb{R}_{\geq 0}^N$ and some auxiliary variable $c \in \Omega$. This also implies that $F$ is convex, since

$$F(v) \geq F(\tilde{v}) + \nabla F(\tilde{v})^T (v - \tilde{v})$$

$$\iff \quad f(c^T v) \geq f(c^T \tilde{v}) + f'(c^T \tilde{v})(c^T v - c^T \tilde{v}).$$

We now choose $\lambda_i \in [0, 1]$ with $\sum_{i=1}^N \lambda_i = 1$ and $\alpha \in \mathbb{R}^N$ and define

$$\lambda_i := \frac{c_i b_i}{c^T b},$$

$$\alpha_i := \frac{c_i v_i}{\lambda_i} \left( = \frac{v_i c^T b}{b_i} \right)$$

for some $b \in \Omega$. This implies

$$F(v) = f(c^T v) = f \left( \sum_{i=1}^N \lambda_i \alpha_i \right) \leq \sum_{i=1}^N \frac{c_i b_i}{c^T b} f \left( \frac{c^T b}{b_i} v_i \right) =: Q_F(v, b). \quad (7)$$

The functional $Q_F : \Omega \times \Omega \to \mathbb{R}$ defines a separable and convex surrogate for $F$, which can be seen by the inequality above and by observing

$$Q_F(v, v) = \sum_{i=1}^N \frac{c_i v_i}{c^T v} f(c^T v) = f(c^T v) = F(v).$$

3.3 Low Quadratic Bound Principle

This concept is based on a Taylor expansion of $F$ in combination with a majorization of the quadratic term. This so-called low quadratic bound principle (LQBP) has been introduced in [5] and was used in particular for the computation of maximum likelihood estimators.

These methods do not require that $F$ itself is convex and its construction is based on the following lemma.

**Lemma 3** (Low quadratic bound principle) Let $\Omega \subseteq \mathbb{R}^N$ denote an open and convex set and $f : \Omega \to \mathbb{R}$ a twice continuously differentiable functional. Assume that a matrix $\Lambda(x) \in \mathbb{R}^{N \times N}$ exists, such that $\Lambda(x) - \nabla^2 f(x)$ is positive semi-definite for all $x \in \Omega$. We
then obtain a quadratic majorization

\[ f(x) \leq f(a) + \nabla f(a)^T (x - a) + \frac{1}{2} (x - a)^T \Lambda(a)(x - a) \quad \forall x, a \in \Omega \]

\[ =: Q_f(x, a), \]

and \( Q_f \) is a surrogate functional for \( f \).

**Proof** The proof of this classical result is based on the second-order Taylor polynomial of \( f \) and shall be left to the reader. \( \square \)

The related update rule for surrogate minimization can be stated explicitly under natural assumptions on the matrix \( \Lambda \).

**Corollary 1** Assume that the assumptions of Lemma 3 hold. In addition, assume that \( \Lambda \) is a positive definite and symmetric matrix. Then, the corresponding surrogate \( Q_f \) is strictly convex in its first variable and we have from (5)

\[ x^{[d+1]} = \arg \min_{x \in \Omega} Q_f(x, x^{[d]}) \]

\[ = x^{[d]} - \Lambda^{-1}(x^{[d]}) \nabla f(x^{[d]}). \]

**Proof** For an arbitrary \( \alpha \in \{1, \ldots, N\} \), we have that

\[ \frac{\partial Q_f}{\partial x_\alpha}(x, a) = \frac{\partial}{\partial x_\alpha} \left( \sum_{i=1}^{N} \frac{\partial f}{\partial x_i}(a)(x_i - a_i) + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \Lambda(a)_{ij} \cdot (x_i - a_i)(x_j - a_j) \right) \]

\[ = \frac{\partial f}{\partial x_\alpha}(a) + \frac{1}{2} \sum_{i=1}^{N} \Lambda(a)_{\alpha i} (x_i - a_i) + \frac{1}{2} \sum_{i \neq \alpha} \sum_{j=i+1}^{N} \Lambda(a)_{ij} (x_i - a_i)(x_j - a_j) + \Lambda(a)_{\alpha \alpha} (x_\alpha - a_\alpha) \]

\[ \overset{(*)}{=} \frac{\partial f}{\partial x_\alpha}(a) + \Lambda(a)_{\alpha \alpha} (x - a), \]

where (\( * \)) utilizes the symmetry of \( \Lambda \). Hence, it holds that \( \nabla_x Q_f(x, a) = \nabla f(a) + \Lambda(a)(x - a) \). The Hessian matrix of \( Q_f \) then satisfies

\[ \nabla_x^2 Q_f(x, a) = \Lambda(a). \]

This implies the positive definiteness of the functional; hence, it has a unique minimizer, which is given by

\[ \nabla_x Q_f(x^*_a, a) = 0 = \nabla f(a) + \Lambda(a)(x^*_a - a) \]

\[ \iff x^*_a = a - \Lambda^{-1}(a) \nabla f(a). \]

This is the update rule above. \( \square \)
The computation of the inverse of $\Lambda_1$ is particularly simple if $\Lambda_1$ is a diagonal matrix. Furthermore, the diagonal structure ensures the separability of the surrogate functional mentioned in Definition 6. Therefore, we consider matrices of the form

$$\Lambda(a)_{ii} := \frac{\nabla^2 f(a)_{ii} + \kappa_i}{a_i},$$

where $\kappa_i \geq 0$ has to be chosen individually depending on the considered cost function. We will see that an appropriate choice of $\kappa_i$ will lead finally to the desired multiplicative update rules of the NMF algorithm.

The matrix $\Lambda(a)$ in (9) fulfills the conditions in Corollary 1 as it can be seen by the following lemma. Therefore, if $\Lambda$ is constructed as in (9), the update rule in (8) can be applied immediately.

**Lemma 4** Let $M \in \mathbb{R}_{\geq 0}^{N \times N}$ denote a symmetric matrix. With $a \in \mathbb{R}^N > 0$ and $\kappa_i \geq 0$, we define the diagonal matrix $\Lambda$ by

$$\Lambda_{ii} := \frac{(Ma)_{ii} + \kappa_i}{a_i}$$

for $i = 1, \ldots, N$. Then, $\Lambda$ and $\Lambda - M$ are positive semi-definite.

**Proof** Let $\zeta \in \mathbb{R}^N$ denote an arbitrary vector and let $\delta$ denote the Kronecker symbol. Then,

$$\zeta^T(\Lambda - M)\zeta = \sum_{i,j=1}^{N} \zeta_i \delta_{ij} \left( \frac{(Ma)_{ii} + \kappa_i}{a_i} \right) \zeta_j - \sum_{i=1}^{N} \zeta_i M_{ij} \zeta_j$$

$$= \sum_{i=1}^{N} \zeta_i^2 \left( \frac{(Ma)_{ii}}{a_i} \right) + \sum_{i,j=1}^{N} \zeta_i \zeta_j M_{ij}$$

$$\geq \sum_{i,j=1}^{N} \zeta_i^2 \frac{a_j}{a_i} M_{ij} - \sum_{i,j=1}^{N} \zeta_i M_{ij} \zeta_j$$

$$= \sum_{i=1}^{N} \zeta_i^2 M_{ii} + \sum_{i,j=1}^{N} \left( \zeta_i^2 \frac{a_j}{a_i} + \zeta_j^2 \frac{a_i}{a_j} \right) M_{ij} - \sum_{i,j=1}^{N} \zeta_i M_{ij} \zeta_j$$

$$= \sum_{i,j=1}^{N} \left[ \frac{1}{2} \zeta_i^2 \frac{a_j}{a_i} M_{ij} + \frac{1}{2} \zeta_j^2 \frac{a_i}{a_j} M_{ij} - \zeta_i M_{ij} \zeta_j \right]$$

$$= \frac{1}{2} \sum_{i,j=1}^{N} \zeta_i^2 \frac{a_j}{a_i} M_{ij} + \zeta_j^2 \frac{a_i}{a_j} M_{ij} - 2 \sqrt{\frac{a_j}{a_i}} \sqrt{\frac{a_i}{a_j}} \zeta_i M_{ij} \zeta_j$$

$$= \frac{1}{2} \sum_{i,j=1}^{N} \left( \sqrt{\frac{a_j}{a_i}} \zeta_i - \sqrt{\frac{a_i}{a_j}} \zeta_j \right)^2 M_{ij} \geq 0.$$

The positive semi-definiteness of $\Lambda$ follows from its diagonal structure. \hfill $\square$

### 3.4 Further Construction Principles

So far we have discussed two major construction principles based on either Jensen’s inequality or upper bounds for the quadratic term in Taylor expansions. Lange [23] lists further
construction principles, which however will not be used for NMF constructions in the subsequent sections of this paper. For completeness, we briefly list their main properties.

A relaxation of the approach based on Jensen’s inequality is achieved by choosing \( \alpha_i \geq 0, i \in \{1, \ldots, N\} \) such that \( \sum_{i=1}^{N} \alpha_i = 1 \) and \( \alpha_i > 0 \) if \( c_i \neq 0 \), which yields

\[
F(v) = f(c^T v) \leq \sum_{i=1}^{N} \alpha_i f \left( \frac{c_i}{\alpha_i} \left( v_i - b_i \right) + c^T b \right) =: Q_F(v, b).
\]

A typical choice is

\[
\alpha_i := \frac{|c_i|^p}{\sum_{j=1}^{N} |c_j|^p}
\]

which leads to surrogate functionals for \( p \geq 0 \). This type of surrogate was originally introduced in the context of medical imaging (see [12]), for positron emission tomography.

Another approach is based on combining arithmetic with geometric means and can be used for constructing surrogates for posynomial functions. For \( \alpha, v, a \in \mathbb{R}^N_{>0} \), we obtain

\[
F(v) = \prod_{i=1}^{N} v_i^{\alpha_i} \leq \left( \prod_{i=1}^{N} a_i^{\alpha_i} \right) \sum_{i=1}^{N} \frac{\alpha_i}{\sum_{k=1}^{N} \alpha_i} \left( \sum_{k=1}^{N} a_i \right) =: Q_F(v, a)
\]

4 Surrogates for NMF Functionals

In this section, we apply the general construction principles of Section 3 to the NMF problem as stated in (1). The resulting functional \( F(K, X) \) depends on both factors of the matrix decomposition and minimization is attempted by alternating minimization with respect to \( K \) and \( X \) as described in (3) and (4).

However, we replace the functional \( F \) in each iteration by suitable surrogate functionals, which allow an explicit minimization. Hence, we avoid the minimization of \( F \) itself, which even for the most simple quadratic formulation requires solving a high-dimensional linear system.

We start by considering the discrepancy terms for \( \beta = 2 \) (Frobenius norm) and \( \beta = 1 \) (Kullback–Leibler divergence) and determine surrogate functionals with respect to \( X \) and \( K \). We then add several penalty terms and develop surrogate functionals accordingly. With regard to the construction of surrogates for the case of \( \beta = 0 \) (Itakura–Saito divergence), we refer to the works [15, 16, 32].

4.1 Frobenius Discrepancy and Low Quadratic Bound Principle

We start by constructing a surrogate for the minimization with respect to \( X \) for the Frobenius discrepancy

\[
F(X) := \frac{1}{2} \| Y - KX \|^2_F.
\]

Let \( Y_{\cdot, j} \), resp. \( X_{\cdot, j} \), denote the column vectors of \( Y \), resp. \( X \). The separability of \( F \) yields

\[
F(X) = \frac{1}{2} \sum_{j=1}^{m} \| Y_{\cdot, j} - KX_{\cdot, j} \|^2 =: \sum_{j=1}^{m} f_{Y_{\cdot, j}}(X_{\cdot, j}).
\]

Hence, the minimization separates for the different \( f_{Y_{\cdot, j}} \) terms. The Hessian of these terms is given by

\[
\nabla^2 f_{Y_{\cdot, j}}(a) = K^T K
\]
and the LQBP construction principle of the previous section with $\kappa_k = 0$ yields

$$\Lambda_{f_{Y^*},j}(a)_{kk} = \frac{(K^T K a)_k}{a_k},$$

leading to the surrogate functionals

$$Q_{f_{Y^*},j}(x, a) = f_{Y^*},j(a) + \nabla f_{Y^*},j(a)^T (x - a) + \frac{1}{2} (x - a)^T \Lambda_{f_{Y^*},j}(a)(x - a).$$

An appropriate choice of $\kappa_k$ ensures the multiplicativity of the final NMF algorithm. In the case of the Frobenius discrepancy term, we will see that suitable $\kappa_k$ can be chosen dependent on $\ell_1$ regularization terms in the cost function, which are not included up to now (see Section 4.4 and Appendix 1.1 for more details on this issue). Due to the absent $\ell_1$ terms, we set $\kappa_k = 0$ to get the desired multiplicative update rules. Summing up the contributions of the columns of $X$ yields the final surrogate

$$Q_F : \mathbb{R}^{p \times m} \times \mathbb{R}^{p \times m} \rightarrow \mathbb{R}, \quad (X, A) \mapsto \sum_{j=1}^m Q_{f_{Y^*},j}(X_{\bullet,j}, A_{\bullet,j}).$$

The equivalent construction for $K$ can be obtained by regarding the rows of $K$ separately, which for

$$g_y : \mathbb{R}^p \rightarrow \mathbb{R}, \quad k \mapsto \frac{1}{2} \|y - kX\|^2$$

yields $\nabla^2 g_y(a) = XX^T$. Putting

$$\Lambda_{g_y}(a)_{kk} = \frac{(aXX^T)_k}{a_k}$$

leads to the surrogate

$$Q_{g_y}(k, a) = g_y(a) + (k - a) \nabla g_y(a) + \frac{1}{2} (k - a) \Lambda_{g_y}(a)(k - a)^T.$$ 

We summarize this surrogate construction in the following theorem.

**Theorem 1** (Surrogate functional for the Frobenius norm with LQBP) We consider the cost functionals $F(X) := 1/2\|Y - KX\|^2$ and $G(K) := 1/2\|Y - KX\|^2$. Then,

$$Q_{F,1}(X, A) = \sum_{j=1}^m Q_{f_{Y^*},j}(X_{\bullet,j}, A_{\bullet,j}),$$

$$Q_{G,1}(K, A) = \sum_{i=1}^n Q_{g_{Y^*},i}(K_{i,\bullet}, A_{i,\bullet})$$

define separable and convex surrogate functionals.

### 4.2 Frobenius Discrepancy and Jensen’s Inequality

Again, we focus on deriving a surrogate functional for $X$; the construction for $K$ will be very similar. Expanding the Frobenius discrepancy yields

$$F(X) := \frac{1}{2} \|Y - KX\|^2_F = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m (Y_{ij} - (KX)_{ij})^2.$$
Putting $v := X \cdot, j \in \mathbb{R}^p_{\geq 0}$ and $c := K_i, \cdot \top \in \mathbb{R}^p_{\geq 0}$ allows us to define
\[
f : \mathbb{R}^p_{\geq 0} \to \mathbb{R} \quad \text{with} \quad f(t) := (Y_{ij} - t)^2,
\]
such that
\[
f(c^T v) = (Y_{ij} - (KX)_{ij})^2.
\]
Hence, we have separated the Frobenius discrepancy suitably for applying Jensen’s inequality. Following the construction principle in Section 3.2, we define
\[
\lambda_k = \frac{K_{ik} A_{kj}}{(KA)_{ij}},
\]
\[
\alpha_k = \frac{K_{ik} X_{kj}}{\lambda_k}
\]
with the auxiliary variable $A \in \mathbb{R}^p \times m_{\geq 0}$ and $b := A \cdot, j \in \mathbb{R}^p_{\geq 0}$, which yields the inequality
\[
(Y_{ij} - (KX)_{ij})^2 \leq \sum_{k=1}^p K_{ik} A_{kj} \left( Y_{ij} - \frac{X_{kj}}{A_{kj}} (KA)_{ij} \right)^2.
\]
Inserting this into the decomposition of the Frobenius discrepancy yields the surrogate $Q_{F,2}(X, A)$ by
\[
F(X) \leq \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m \frac{1}{(KA)_{ij}} \sum_{k=1}^p K_{ik} A_{kj} \left( Y_{ij} - \frac{X_{kj}}{A_{kj}} (KA)_{ij} \right)^2 =: Q_{F,2}(X, A).
\]
The construction of a surrogate for $K$ proceeds in the same way. We summarize the results in the following theorem.

**Theorem 2** (Surrogate functional for the Frobenius norm with Jensen’s inequality) We consider the cost functionals $F(X) := 1/2\|Y - KX\|_F^2$ and $G(K) := 1/2\|Y - KX\|_F^2$. Then,
\[
Q_{F,2}(X, A) := \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m \frac{1}{(KA)_{ij}} \sum_{k=1}^p K_{ik} A_{kj} \left( Y_{ij} - \frac{X_{kj}}{A_{kj}} (KA)_{ij} \right)^2,
\]
\[
Q_{G,2}(K, A) := \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m \frac{1}{(AX)_{ij}} \sum_{k=1}^p A_{ik} X_{kj} \left( Y_{ij} - \frac{K_{ik}}{A_{ik}} (AX)_{ij} \right)^2
\]
define separable and convex surrogate functionals.

These surrogates are equal to the ones proposed in [11]. We will later use first-order necessary conditions of the surrogate functionals for obtaining algorithms for minimization. We already note
\[
\frac{\partial Q_{F,1}}{\partial X_{\alpha \beta}} = \frac{\partial Q_{F,2}}{\partial X_{\alpha \beta}} \quad \text{and} \quad \frac{\partial Q_{G,1}}{\partial K_{\alpha \beta}} = \frac{\partial Q_{G,2}}{\partial K_{\alpha \beta}},
\]
i.e., despite the rather different derivations, the update rules for the surrogates obtained by LQBP and Jensen’s inequality will be identical.
4.3 Surrogates for Kullback–Leibler Divergence

The case $\beta = 1$ in Definition 3 yields the so-called Kullback–Leibler divergence (KLD). For matrices $M, N \in \mathbb{R}_{>0}^{n \times m}$, it is defined as

$$
KL(M, N) := D_1(M, N) = \sum_{i=1}^{n} \sum_{j=1}^{m} M_{ij} \log \left( \frac{M_{ij}}{N_{ij}} \right) - M_{ij} + N_{ij}
$$

and has been investigated intensively in connection with non-negative matrix factorization methods [11, 16, 24, 25]. In our context, we define the cost functional for the NMF decomposition by

$$
F(X, K) := KL(Y, KX).
$$

We will focus in this subsection on Jensen’s inequality for constructing surrogates for the KLD since they will lead to the known classical NMF algorithms (see also [11, 24, 25]). However, it is also possible to use the LQBP principle to construct a suitable surrogate functional for the KLD which leads to different, multiplicative update rules (see Appendix 2).

We start by deriving a surrogate for the minimization with respect to $X$, i.e., we consider

$$
F(X) := KL(Y, KX) = \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} \log(Y_{ij}) - Y_{ij} \log((KX)_{ij}) - Y_{ij} + (KX)_{ij}.
$$

Using the same $\lambda_k$ and $\alpha_k$ as in the section above and applying it to the convex function $f(t) := -\ln(t)$, we obtain

$$
-\ln ((KX)_{ij}) \leq -\sum_{k=1}^{p} K_{ik} A_{kj} \ln \left( \frac{X_{kj}}{A_{kj} (KA)_{ij}} \right).
$$

Multiplication with $Y_{ij} \geq 0$ and the addition of appropriate terms yield

$$
F(X) \leq \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} \ln(Y_{ij}) - Y_{ij} + (KX)_{ij} - \frac{Y_{ij}}{(KA)_{ij}} \sum_{k=1}^{p} K_{ik} A_{kj} \ln \left( \frac{X_{kj}}{A_{kj} (KA)_{ij}} \right)
$$

$$
=: Q_{F,1}(X, A).
$$

The condition $Q_{F,1}(X, X) = F(X)$ follows by simple algebraic manipulations, such that $Q_{F,1}$ is a valid surrogate functional for $F$.

The approach by Jensen’s inequality is very flexible and we obtain different surrogate functionals $Q_{F,2}$ and $Q_{F,3}$ by using, i.e., $f_1(t) = Y_{ij} \ln(Y_{ij}/t) - Y_{ij} + t$ or $f_2(t) = -Y_{ij} \ln(t) + t$ instead of $f$. Inserting the same $\lambda_k$ and $\alpha_k$ as before in (7), we obtain immediately the surrogates

$$
Q_{F,2}(X, A) = \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{1}{(KA)_{ij}} \sum_{k=1}^{p} K_{ik} A_{kj} \left( Y_{ij} \ln \left( \frac{Y_{ij}}{X_{kj} (KA)_{ij}} \right) - Y_{ij} + \frac{X_{kj}}{A_{kj} (KA)_{ij}} \right),
$$

$$
Q_{F,3}(X, A) = \sum_{i=1}^{n} \sum_{j=1}^{m} \left[ Y_{ij} \ln(Y_{ij}) - Y_{ij}
$$

$$
+ \frac{1}{(KA)_{ij}} \sum_{k=1}^{p} K_{ik} A_{kj} \left( -Y_{ij} \ln \left( \frac{X_{kj}}{A_{kj} (KA)_{ij}} \right) + \frac{X_{kj}}{A_{kj} (KA)_{ij}} \right) \right].
$$
It is easy to check that the partial derivatives for all three variants are the same; hence, the update rules obtained in the next section based on first-order optimality conditions will be identical. Applying the same approach for obtaining a surrogate for $K$ yields the following theorem.

**Theorem 3** (Surrogate functional for the KLD with Jensen’s inequality) We consider the cost functionals $F(X) := \text{KL}(Y, KX)$ and $G(K) := \text{KL}(Y, KX)$. Then,

\[
Q_F(X, A) := \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} \ln(Y_{ij}) - Y_{ij} + (KX)_{ij} - \frac{Y_{ij}}{(KA)_{ij}} \sum_{k=1}^{p} K_{ik} A_{kj} \ln\left(\frac{X_{kj}}{A_{kj}}(KA)_{ij}\right),
\]

\[
Q_G(K, A) := \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} \ln(Y_{ij}) - Y_{ij} + (KX)_{ij} - \frac{Y_{ij}}{(AX)_{ij}} \sum_{k=1}^{p} A_{ik} X_{kj} \ln\left(\frac{K_{ik}}{A_{ik}}(AX)_{ij}\right)
\]

define separable and convex surrogate functionals.

### 4.4 Surrogates for $\ell_1$- and $\ell_2$-Norm Penalties

Computing an NMF is an ill-posed problem (see [11]), hence, one needs to add stabilizing penalty terms for obtaining reliable matrix decompositions. The most standard penalties are $\ell_1$- and $\ell_2$-terms for the matrix factors leading to

\[
\min_{K, X \geq 0} D_\beta(Y, KX) + \lambda \|X\|_1 + \frac{\mu}{2} \|K\|^2_F + \frac{\nu}{2} \|X\|^2_F + \omega \|K\|_1
\]

for $\beta \in \{1, 2\}$.

The $\ell_2$-penalty prohibits exploding norms for each matrix factor and the $\ell_1$-term promotes sparsity in the minimizing factors (see [21, 28] for a general exposition). Combinations of $\ell_1$- and $\ell_2$-norms are sometimes called elastic net regularization [20] due to their importance in medical imaging.

These penalty terms are convex and they separate; hence, they can be used as surrogates themselves. For the case of the KLD, this leads to the following surrogate for minimization with respect to $X$:

\[
Q_F(X, A) := Q_{KL}(X, A) + \lambda \|X\|_1 + \frac{\nu}{2} \|X\|^2_F,
\]

where $Q_{KL}$ is the surrogate for the KLD of Theorem 3 for $X$.

The Frobenius case cannot be treated in the same way. If we use the penalty terms as surrogates themselves and obtain the standard minimization algorithm by first-order optimality conditions, then this does not lead to a multiplicative algorithm, which preserves the non-negativity of the iterates. It can be easily seen that the $\ell_1$-penalty term causes this difficulty. For a more extended discussion on this, see Appendix 1.1.

Hence, we have to construct a different surrogate. Similar to the discussion in Section 4.1, we consider here $f_y : \mathbb{R}^p_{\geq 0} \to \mathbb{R}$ with

\[
f_y(x) := \frac{1}{2} \|y - Kx\|^2 + \lambda \|x\|_1 + \frac{\nu}{2} \|x\|^2,
\]

which yields the Hessian $\nabla^2 f_y(a) = KK^T + \nu I$. The choice of $\kappa_k$ is done dependent on the $\ell_1$ regularization term of the cost function $f_y$ as already described in Section 4.1. It can be shown in the derivation of the NMF algorithm that $\kappa_k = \lambda$ for all $k$ leads to multiplicative update rules. A more general cost function is considered in Appendix 1.1, where the concrete effect of $\kappa_k$ is described in more detail.
This yields the following diagonal matrix $\Lambda_{f_y}(a)$:

$$\Lambda_{f_y}(a)_{kk} = \frac{((K^T K + \nu I_{p \times p})a)_k + \lambda}{a_k}. $$

The surrogate for minimization with respect to $X$ is then

$$Q_{f_y}(x, a) = f_y(a) + \nabla f_y(a)^T (x - a) + \frac{1}{2} (x - a)^T \Lambda_{f_y}(a)(x - a).$$

Similar, for minimization with respect to $K$ we obtain the surrogate by using the diagonal matrix

$$\Lambda_{g_y}(a)_{kk} := \frac{(a(XX^T + \mu I_{p \times p}))_k + \omega}{a_k}. $$

### 4.5 Surrogates for Orthogonality Constraints

The observation that a non-negative matrix with pairwise orthogonal rows has at most one non-zero entry per column is the motivation for introducing orthogonality constraints for $K$ or $X$. This will lead to strictly uncorrelated feature vectors, which is desirable in several applications, e.g., for obtaining discriminating biomarkers from mass spectra (see Section 6 on MALDI imaging).

We could add the orthogonality constraint $K^T K = I$ as an additional penalty term $\sigma_K \|K^T K - I\|^2_F$. However, this would introduce fourth order terms. Hence, we introduce additional variables $V$ and $W$ and split the orthogonality condition into two second-order terms leading to

$$\min_{K, X, V, W \geq 0} \left\{ D_\beta(Y, KX) + \frac{\sigma_K,1}{2} \|I - V^T K\|^2_F + \frac{\sigma_K,2}{2} \|V - K\|^2_F 
+ \frac{\sigma_X,1}{2} \|I - XW^T\|^2_F + \frac{\sigma_X,2}{2} \|W - X\|^2_F \right\}. $$

Surrogates for the terms $\|I - V^T K\|^2_F$ and $\|I - XW^T\|^2_F$ can be calculated via Jensen’s inequality (see Section 4.2). The other penalties can be used as surrogates themselves and therefore, we obtain the following theorem.

**Theorem 4** (Surrogate functionals for orthogonality constraints) We consider the cost functionals

$$F(X) := \frac{\sigma_X,1}{2} \|I - XW^T\|^2_F + \frac{\sigma_X,2}{2} \|W - X\|^2_F =: G(W),$$

$$H(K) := \frac{\sigma_K,1}{2} \|I - V^T K\|^2_F + \frac{\sigma_K,2}{2} \|V - K\|^2_F =: J(V).$$
with \( \sigma_{X,1}, \sigma_{X,2}, \sigma_{K,1}, \sigma_{K,2} \geq 0 \). Then,

\[
Q_F(X, A) := \frac{\sigma_{X,1}}{2} \sum_{k=1}^{p} \sum_{\ell=1}^{p} \frac{1}{(AW^T)_{k\ell}} \sum_{j=1}^{m} A_{kj} W_{\ell j} \left( \delta_{k\ell} - \frac{X_{kj}}{A_{kj}} (AW^T)_{k\ell} \right)^2 + \frac{\sigma_{X,2}}{2} \| W - X \|_F^2,
\]

\[
Q_G(W, A) := \frac{\sigma_{X,1}}{2} \sum_{k=1}^{p} \sum_{\ell=1}^{p} \frac{1}{(XA^T)_{k\ell}} \sum_{j=1}^{m} X_{kj} A_{\ell j} \left( \delta_{k\ell} - \frac{W_{\ell j}}{A_{\ell j}} (XA^T)_{k\ell} \right)^2 + \frac{\sigma_{X,2}}{2} \| W - X \|_F^2,
\]

\[
Q_H(K, A) := \frac{\sigma_{K,1}}{2} \sum_{k=1}^{p} \sum_{\ell=1}^{p} \frac{1}{(V^T A)_{k\ell}} \sum_{j=1}^{n} V_{ik} A_{i\ell} \left( \delta_{k\ell} - \frac{K_{i\ell}}{A_{i\ell}} (V^T A)_{k\ell} \right)^2 + \frac{\sigma_{K,2}}{2} \| V - K \|_F^2,
\]

\[
Q_J(V, A) := \frac{\sigma_{K,1}}{2} \sum_{k=1}^{p} \sum_{\ell=1}^{p} \frac{1}{(AT K)_{k\ell}} \sum_{j=1}^{n} A_{ik} K_{i\ell} \left( \delta_{k\ell} - \frac{V_{ik}}{A_{ik}} (AT K)_{k\ell} \right)^2 + \frac{\sigma_{K,2}}{2} \| V - K \|_F^2,
\]

define separable and convex surrogate functionals.

### 4.6 Surrogates for Total Variation Penalties

Total variation (TV) penalty terms are the second important class of regularization terms besides \( \ell_p \)-penalty terms. TV penalties aim at smooth or even piecewise constant minimizers; hence, they are defined in terms of first-order or higher-order derivatives [7].

Originally, they were introduced for denoising applications in image processing [31] but have since been applied to inpainting, deconvolution, and other inverse problems (see, e.g., [8]). The precise mathematical formulation of the total variation in the continuous case is described in the following definition.

**Definition 7** (Total variation (continuous)) Let \( \Omega \subset \mathbb{R}^N \) be open and bounded. The total variation of a function \( u \in L^1_{\text{loc}}(\Omega) \) is defined as

\[
\text{TV}(u) := \sup \left\{ - \int_{\Omega} u \text{ div } \phi \, dx : \phi \in C^\infty_c(\Omega, \mathbb{R}^N) \text{ with } \| \phi(x) \| \leq 1 \ \forall x \in \Omega \right\}.
\]

There exist several analytic relaxations of TV based on \( \ell_1 \)-norms of the gradient, which are more tractable for analytical investigations. For numerical implementations, one rather uses the \( L_1 \)-norm of the gradient \( \| \nabla f \|_{L_1} \) as a more computationally tractable substitute. For discretization, the gradient is typically replaced by a finite difference approximation [9].

For applying TV norms to data, we assume that the row index in the data matrix \( Y \) refers to spatial locations and the column index to so-called channels. In this case, we consider the most frequently used isotropic TV for applying it to measured, discretized hyperspectral data.

**Definition 8** (Total variation (discrete)) For fixed \( \varepsilon_{\text{TV}} > 0 \), the total variation of a matrix \( K \in \mathbb{R}^{n \times p}_{\geq 0} \) is defined as

\[
\text{TV}(K) := \sum_{k=1}^{p} \psi_k \sum_{i=1}^{n} \sqrt{\varepsilon_{\text{TV}}^2 + \sum_{\ell \in N_i} (K_{ik} - K_{i\ell})^2},
\]

where \( \psi_k \in \mathbb{R}_{\geq 0} \) is a weighting of the \( k \)th data channel and \( N_i \subseteq \{1, \ldots, n\} \setminus \{i\} \) denotes the index set referring to spatially neighboring pixels.
We will use the following short hand notation
\[ |\nabla_{ik} K| := \sqrt{\varepsilon^2_{TV} + \sum_{\ell \in N_i} (K_{ik} - K_{\ell k})^2}, \]
which can be seen as a finite difference approximation of the gradient magnitude of the image \( K_{\bullet,k} \) at pixel \( K_{ik} \) for some neighborhood pixels defined by \( N_i \). A typical choice for neighborhood pixels in two dimensions for the pixel \((0,0)\) is \( N(0,0) := \{(1,0), (0,1)\} \) to get an estimate of the gradient components along both axes. Finally, by introducing the positive constant \( \varepsilon_{TV} > 0 \), we get a differentiable approximation of the total variation penalty.

In Section 6, we will discuss the application of NMF methods to hyperspectral MALDI imaging datasets, which has a natural “spatial structure” in its columns.

In this section, we stay with a generic choice of \( N_i \) as well as of the \( \psi_k \) and we construct a surrogate following the approach of the groundbreaking works of [13] and [29].

For \( t \geq 0 \) and \( s > 0 \), we use the inequality (linear majorization)
\[ \sqrt{t} \leq \sqrt{s} + \frac{t - s}{2\sqrt{s}} \]
and apply it in order to compare \( |\nabla_{ik} K| \) with values obtained by an arbitrary non-negative matrix \( A \):
\[ |\nabla_{ik} K| \leq |\nabla_{ik} A| + \frac{|\nabla_{ik} K|^2 - |\nabla_{ik} A|^2}{2|\nabla_{ik} A|} \leq \frac{|\nabla_{ik} K|^2 + |\nabla_{ik} A|^2}{2|\nabla_{ik} A|} = \frac{2\varepsilon^2_{TV} + \sum_{\ell \in N_i} [(K_{ik} - K_{\ell k})^2 + (A_{ik} - A_{\ell k})^2]}{2|\nabla_{ik} A|}. \]
Summation with respect to \( i \), multiplication with \( \psi_k \), and summation with respect to \( k \) lead to
\[ TV(K) \leq \sum_{k=1}^{p} \psi_k \sum_{i=1}^{n} \frac{2\varepsilon^2_{TV} + \sum_{\ell \in N_i} [(K_{ik} - K_{\ell k})^2 + (A_{ik} - A_{\ell k})^2]}{2|\nabla_{ik} A|} =: Q^{Oli}_{TV}(K,A). \]
This yields a candidate for a surrogate \( Q^{Oli}_{TV} \) for the TV penalty term, which is the same as the one used in [29]. However, it is not separable; hence, we aim at a second, separable approximation. For arbitrary \( a, b, c, d \in \mathbb{R} \), we have
\[ \frac{1}{2} \left( (a - b)^2 + (c - d)^2 \right) \leq (a - b)(c - d) + (b - d)^2 + (a - c)^2. \]
This leads to
\[ Q^{TV}_{TV}(K,A) = \sum_{k=1}^{p} \psi_k \sum_{i=1}^{n} \frac{\varepsilon^2_{TV} + \sum_{\ell \in N_i} 1/2 [(K_{ik} - K_{\ell k})^2 + (A_{ik} - A_{\ell k})^2]}{|\nabla_{ik} A|} \leq \sum_{k=1}^{p} \psi_k \sum_{i=1}^{n} \frac{\varepsilon^2_{TV} + \sum_{\ell \in N_i} [(K_{ik} - K_{\ell k})(A_{ik} - A_{\ell k}) + (K_{ik} - A_{ik})^2 + (K_{\ell k} - A_{\ell k})^2]}{|\nabla_{ik} A|} =: Q_{TV}(K,A). \]
Therefore, we have the following theorem.
Theorem 5 (Surrogate functional for TV penalty term) We consider the cost functional \( F(K) := TV(K) \) with the total variation defined in (10). Then,

\[
Q_{TV}(K, A) := \sum_{k=1}^{p} \psi_{k} \sum_{i=1}^{n} \varepsilon_{TV}^{2} + \sum_{\ell \in N_i} \left[ \frac{(K_{ik} - K_{\ell k})(A_{ik} - A_{\ell k}) + (K_{\ell k} - A_{\ell k})^{2} + (K_{ik} - A_{ik})^{2}}{|\nabla_{ik} A|} \right]
\]

defines a separable and convex surrogate functional.

The separability of the surrogate is not obvious. The proof (see Appendix 3) delivers the following notation, which we also need for an description of the algorithms in the next section. First of all, we need the definition of the so-called adjoint neighborhood pixels \( \tilde{N}_{i} \) given by

\[
\ell \in \tilde{N}_{i} \iff i \in N_{\ell}.
\]

One then introduces matrices \( P(A) \in \mathbb{R}_{\geq 0}^{n \times p} \) and \( Z(A) \in \mathbb{R}_{\geq 0}^{n \times p} \) via

\[
P(A)_{ik} := \frac{1}{|\nabla_{ik} A|} \sum_{\ell \in N_{i}} 1 + \sum_{\ell \in \tilde{N}_{i}} \frac{1}{|\nabla_{\ell k} A|},
\]

\[
Z(A)_{ik} := \frac{1}{P(A)_{ik}} \left( \frac{1}{|\nabla_{ik} A|} \sum_{\ell \in N_{i}} \frac{A_{ik} + A_{\ell k}}{2} + \sum_{\ell \in \tilde{N}_{i}} \frac{A_{ik} + A_{\ell k}}{2|\nabla_{\ell k} A|} \right).
\]

Using these notations, it can be shown that the surrogate can be written as

\[
Q_{TV}(K, A) = \sum_{k=1}^{p} \psi_{k} \sum_{i=1}^{n} \left[ P(A)_{ik} (K_{ik} - Z(A)_{ik})^{2} \right] + C(A),
\]

such that we obtain the desired separability. Here, \( C(A) \) denotes some function depending on \( A \). The description of \( Q_{TV} \) with the help of \( P(A)_{ik} \) and \( Z(A)_{ik} \) will also allow us to compute the partial derivatives in a more comfortable way (see also Appendix 1.2).

4.7 Surrogates for Supervised NMF

As a motivation for this section, we consider classification tasks. We view the data matrix \( Y \) as a collection of \( n \) data vectors, which are stored in the rows of \( Y \). Moreover, we do have an expert annotation \( u_{i} \) for \( i = 1, \ldots, n \), which assigns a label to each data vector. For a classification problem with two classes, we have \( u_{i} \in \{0, 1\} \).

As already stated, the rows of the matrix \( X \) of an NMF decomposition can be regarded as a basis for approximating the rows of \( Y \). Hence, one assumes that the correlations between a row \( Y_{i, \bullet} \) of \( Y \) and all row vectors of \( X \), i.e., computing \( Y_{i, \bullet} X^{T} \), contain the relevant information of \( Y_{i, \bullet} \). The vector of correlations yields the so-called feature vector of length \( p \). A classical linear regression model, which uses these feature vectors, then asks to compute weights \( \beta_{k} \) for \( k = 1, \ldots, p \), such that \( Y_{i, \bullet} X^{T} \beta \approx u_{i} \) (for more details on linear discriminant analysis methods, we refer to [4, Chapter 4]).

In matrix notation and using least squares, this is equivalent to computing \( \beta \) as a minimizer of

\[
\| u - Y X^{T} \beta \|^{2}.
\]

We now use \( X \) and \( \beta \) to define

\[
x^{*} := X^{T} \beta.
\]
In tumor typing classifications, where the data matrix $Y$ is obtained by MALDI measurements, the vector $x^*$ can be interpreted as a characteristic mass spectrum of some specific tumor type and can be directly used for classification tasks in the arising field of digital pathology (see also Section 6 and [26]).

The classification of a new data vector $y$ is then simply obtained by computing the scalar product $w = x^T y$ and assigning either the class label 0 or 1 by comparing $w$ with a pre-assigned threshold $s$. This threshold is typically obtained in the training phase of the classification procedure by computing $YX^T \beta$ for some given training data $Y$ and choosing $s$, such that a performance measure of the classifier is optimized.

The approach we have described is based on first computing an NMF, i.e., $K$ and $X$, and then computing the weights $\beta$ of the classifier. Hence, the computation of the NMF is done independently of the class labels $u$, which is also referred to as an unsupervised NMF approach. We might expect that computing the NMF by minimizing a functional which includes the class labels, i.e.,

$$F(K, X, \beta) := D_\beta(Y, KX) + \frac{\rho}{2} \|u - YX^T \beta\|^2,$$

will lead to an improved classifier. In the application field of MALDI imaging, this supervised approach yields an extraction of features from the given training data, which allow a better distinction between spectra obtained from different tissue types such as tumorous and non-tumorous regions (see also [26]).

Surrogates for the first term have been determined in the previous section for the case of the Frobenius norm and the Kullback–Leibler divergence. Hence, we need to determine surrogates of the new penalty term for minimization with respect to $X$ and $\beta$:

$$F(X) := \frac{1}{2} \|u - YX^T \beta\|^2,$$

$$G(\beta) := \frac{1}{2} \|u - YX^T \beta\|^2.$$

Surrogates can be obtained by extending Jensen’s inequality to the matrix valued case. Here, we consider a convex subset $\Omega \subset \mathbb{R}_{>0}^{N \times M}$ and define

$$\tilde{F} : \Omega \rightarrow \mathbb{R}$$

$$V \mapsto f(c^T Vd)$$

with a convex and continuously differentiable function $f$ and auxiliary variables $c \in \mathbb{R}_{>0}^N$ and $d \in \mathbb{R}_{>0}^M$. We now use the following generalized Jensen’s inequality

$$f \left( \sum_{j=1}^N \sum_{k=1}^M \lambda_{jk} \alpha_{jk} \right) \leq \sum_{j=1}^N \sum_{k=1}^M \lambda_{jk} f(\alpha_{jk}). \quad (11)$$

Setting

$$\lambda_{jk} = \frac{Y_{ij} A_{kj} \beta_k}{Y_{i,\cdot} A^\top \beta},$$

$$\alpha_{jk} = \frac{Y_{ij} X_{kj} \beta_k}{\lambda_{jk}}$$
for some $i \in \{1, \ldots, n\}$ yields by inserting $\lambda_{jk}$ and $\alpha_{jk}$ into (11)

$$F(X) \leq \frac{1}{2} \sum_{i=1}^{n} \frac{1}{(YA^T \beta)_i} \sum_{j=1}^{m} \sum_{k=1}^{p} Y_{ij} A_{kj} \beta_k \left(u_i - \frac{X_{kj}}{A_{kj}} (YA^T \beta)_i\right)^2 =: Q_F(X, A).$$

The computation of a surrogate for minimization with respect to $\beta$ proceeds analogously. We summarize the results in the following theorem.

**Theorem 6** (Surrogate functionals for linear regression) Let $F(X) := \frac{1}{2} \|u - YX^T \beta\|^2$ and $G(\beta) := \frac{1}{2} \|u - YX^T \beta\|^2$ denote a cost functional with respect to $X$ and $\beta$. Then,

$$Q_F(X, A) := \frac{1}{2} \sum_{i=1}^{n} \frac{1}{(YA^T \beta)_i} \sum_{j=1}^{m} \sum_{k=1}^{p} Y_{ij} A_{kj} \beta_k \left(u_i - \frac{X_{kj}}{A_{kj}} (YA^T \beta)_i\right)^2,$$

$$Q_G(\beta, a) := \frac{1}{2} \sum_{i=1}^{n} \frac{1}{(YX^T a)_i} \sum_{k=1}^{p} (YX^T)_{iak} \beta_k \left(u_i - \frac{\beta_k}{a_k} (YX^T a)_i\right)^2$$

define separable and convex surrogate functionals.

A big advantage of linear regression models is their simplicity and manageability. However, they are by far not the optimal approach to approximate the binary output data $u$ with a continuous input. Logistic regression models offer a way more natural method for binary classification tasks. Together with the supervised NMF as a feature extraction method, this overall workflow leads in [26] to excellent classification results and outperformed classical approaches.

However, the proposed model is based on a gradient descent approach, such that the non-negativity of the iterates can only be guaranteed by a projection step. Appropriate surrogate functionals for this workflow are still ongoing research and could lead to even better outcomes (see also [35, 36]).

## 5 Surrogate-Based NMF Algorithms

In the previous section, we have defined surrogate functionals for various NMF cost functions. Besides the necessary surrogate properties, we also expect that the minimization of these surrogates is straightforward and can be computed efficiently.

In our case, we demand additionally that the minimization schemes based on solving the first-order optimality conditions leads to a separable algorithm and that it only requires multiplicative updates, which automatically preserve the non-negativity of its iterates. Let us start with denoting the most general functional with Kullback–Leibler divergence, the Frobenius case follows similarly.

For constructing non-negative matrix factorizations, we incorporate $\ell_2$-, sparsity-, orthogonality-, and TV-constraints and also the penalty terms coming from the supervised NMF. Of course, in most applications, one only uses a subset of these constraints for stabilization and for enhancing certain properties. These algorithms can readily be obtained from the general case by putting the respective regularization parameters to zero. The corresponding update rules are classical results and can be found in numerous works [11, 24, 25].
Definition 9 (NMF problem) For $Y \in \mathbb{R}_{\geq 0}^{n \times m}$, $K, V \in \mathbb{R}_{\geq 0}^{n \times p}$, $X, W \in \mathbb{R}_{\geq 0}^{p \times m}$, $\beta \in \mathbb{R}_{\geq 0}$ and a set of regularization parameters $\lambda, \mu, \nu, \omega, \tau, \sigma_{K,1}, \sigma_{K,2}, \sigma_{X,1}, \sigma_{X,2}, \rho \geq 0$, we define the NMF minimization problem by

$$
\min_{K,X,V,W,\beta \geq 0} \left\{ KL(Y, KX) + \frac{\mu}{2} \|K\|_F^2 + \frac{\nu}{2} \|X\|_F^2 + \omega \|K\|_1 + \frac{\tau}{2} TV(K) \\
+ \frac{\sigma_{K,1}}{2} \|I - V^T K\|_F^2 + \frac{\sigma_{K,2}}{2} \|V - K\|_F^2 + \frac{\sigma_{X,1}}{2} \|I - XW^T\|_F^2 \\
+ \frac{\sigma_{X,2}}{2} \|W - X\|_F^2 + \frac{\rho}{2} \|u - YX^T \beta\|_F^2 \right\}.
$$

The choice of the various regularization parameters occurring in Definition 9 is often based on heuristic approaches. We will not focus on that issue in this work and refer instead to [19] and the references therein, where two methods are investigated for the general case of multi-parameter Tikhonov regularization.

The algorithms studied in this section will start with positive initializations for $K$, $X$, $V$, $W$, and $\beta$. These matrices are updated alternatingly, i.e., all matrices except one matrix are kept fixed and only the selected matrix is updated by solving the respective first-order optimality condition.

We will focus in this section on the derivation of the update rules of $K$ (see also Appendix 1.2). The iteration schemes for the other matrices follow analogously.

For that, we only have to consider those terms in the general functional which depend on $K$, i.e., we aim at determining a minimizer for

$$
F(K) := KL(Y, KX) + \frac{\mu}{2} \|K\|_F^2 + \omega \|K\|_1 + \frac{\tau}{2} TV(K) + \frac{\sigma_{K,1}}{2} \|I - V^T K\|_F^2 + \frac{\sigma_{K,2}}{2} \|V - K\|_F^2.
$$

Instead of minimizing this functional, we exchange it with the previously constructed surrogate functionals, which leads to

$$
Q_F(K, A) := Q_{KL}(K, A) + \frac{\mu}{2} \|K\|_F^2 + \omega \|K\|_1 + \frac{\tau}{2} Q_{TV}(K, A) + Q_{Orth}(K, A)
$$

with the surrogates $Q_{KL}$ for the Kullback–Leibler divergence in Theorem 3, $Q_{TV}$ for the TV penalty term in Theorem 5 and $Q_{Orth}$ for the orthogonality penalty terms in Theorem 4.

The computation of the partial derivatives leads to a system of equations

$$
\frac{\partial Q_F}{\partial K_{\xi \xi}}(K, A) = 0
$$

for $\xi \in \{1, \ldots, n\}$ and $\zeta \in \{1, \ldots, p\}$. This leads to a system of quadratic equations

$$
K_{\xi \xi}^2 \left( \mu + \tau \psi_\xi P_{\xi \xi}(A) + \frac{\sigma_{K,1}}{A_{\xi \xi}} (VV^T A)_{\xi \xi} + \sigma_{K,2} \right)
+ K_{\xi \xi} \left( \sum_{j=1}^m X_{\xi j} + \omega - \tau \psi_\xi P(A)_{\xi \xi} Z(A)_{\xi \xi} - (\sigma_{K,1} + \sigma_{K,2}) V_{\xi \xi} \right)
= A_{\xi \xi} \sum_{j=1}^m \frac{Y_{\xi j}}{(AX)_{\xi j}} X_{\xi j}.
$$
Solving for $K_{\xi\zeta}$ and denoting the Hadamard product by $\circ$ as well as the matrix division for each entry separately by a fraction line yield the following update rule for $K$. (Note that the notation for $P(A)$ and $Z(A)$ was introduced in the section on TV regularization above.)

$$K^{[d+1]} = \left[ \begin{array}{c} K^{[d]} \\ \mu 1_{n \times p} + \tau \Psi \circ P(K^{[d]}) + \sigma_{K,1} \frac{VV^TK^{[d]}}{K^{[d]}} + \sigma_{K,2} 1_{n \times p} \end{array} \right] \circ \left( \frac{Y}{K^{[d]}X^T} \right)$$

$$= : \Theta^{[d]}$$

$$+ \frac{1}{4} \left( \frac{1}{n \times p} + \tau \Psi \circ P(K^{[d]}) \circ Z(K^{[d]}) - (\sigma_{K,1} + \sigma_{K,2})V \right)$$

$$\mu 1_{n \times p} + \tau \Psi \circ P(K^{[d]}) + \sigma_{K,1} \frac{VV^TK^{[d]}}{K^{[d]}} + \sigma_{K,2} 1_{n \times p}$$

$$- \frac{1}{2} \left( \frac{1}{n \times p} + \tau \Psi \circ P(K^{[d]}) \circ Z(K^{[d]}) - (\sigma_{K,1} + \sigma_{K,2})V \right)$$

$$\mu 1_{n \times p} + \tau \Psi \circ P(K^{[d]}) + \sigma_{K,1} \frac{VV^TK^{[d]}}{K^{[d]}} + \sigma_{K,2} 1_{n \times p}$$

In the above update rule, $1_{n \times p}$ denotes an $n \times p$ matrix with ones in every entry and $\Psi \in \mathbb{R}^{n \times p}_{\geq 0}$ is defined as

$$\Psi := \begin{pmatrix} \psi_1 & \psi_2 & \cdots & \psi_p \\ \psi_1 & \psi_2 & \cdots & \psi_p \\ \vdots & \vdots & \ddots & \vdots \\ \psi_1 & \psi_2 & \cdots & \psi_p \end{pmatrix}.$$

The exponents are applied on the matrix entries componentwise. Details on the derivation can be found in Appendix 1.2.

The partial derivatives with respect to $X$ are computed similarly. Defining

$$\Lambda^{[d]} := \left( \frac{X^{[d]}}{\nu 1_{p \times m} + \sigma_{X,1} \frac{X^{[d]}W^TW}{X^{[d]}} + \sigma_{X,2} 1_{p \times m} + \rho \beta \beta^T X^{[d]}Y^TY} \right) \circ \left( K^{\top} \frac{Y}{KX^{[d]}} \right)$$

$$\Gamma^{[d]} := \frac{K^{\top} 1_{n \times m} + \lambda 1_{p \times m} - (\sigma_{X,1} + \sigma_{X,2})W - \rho \beta u^TY}{\nu 1_{p \times m} + \sigma_{X,1} \frac{X^{[d]}W^TW}{X^{[d]}} + \sigma_{X,2} 1_{p \times m} + \rho \beta \beta^T X^{[d]}Y^TY}$$

leads to the update

$$X^{[d+1]} = \sqrt{\Lambda^{[d]} + \frac{1}{4} \Gamma^{[d]} \circ \Gamma^{[d]} - \frac{1}{2} \Gamma^{[d]}}.$$

The updates for $V$, $W$ are straight forward and we obtain the following theorem.
Theorem 7 (Alternating algorithm for the NMF problem in Definition 9) The initializations $K^{[0]}, V^{[0]} \in \mathbb{R}_{>0}^{n \times p}, X^{[0]}, W^{[0]} \in \mathbb{R}_{>0}^{p \times m}, \beta^{[0]} \in \mathbb{R}_{>0}^{p}$ and the iterative updates

$$
V^{[d+1]} = \frac{(\sigma_{K,1} + \sigma_{K,2})K^{[d]} - \sigma_{K,2}1_{n \times p}}{\sigma_{K,1}K^{[d]}V^{[d]} + \sigma_{K,2}1_{n \times p}}, 
$$

(13)

$$
K^{[d+1]} = \sqrt{\Theta^{[d]} + \frac{1}{4}\Phi^{[d]} \circ \Phi^{[d]} - \frac{1}{2}\Phi^{[d]}}, 
$$

(14)

$$
W^{[d+1]} = \frac{(\sigma_{X,1} + \sigma_{X,2})X^{[d]} - \sigma_{X,2}1_{p \times m}}{\sigma_{X,1}W^{[d]}X^{[d]} + \sigma_{X,2}1_{p \times m}}, 
$$

(15)

$$
X^{[d+1]} = \sqrt{\Lambda^{[d]} + \frac{1}{4}\Gamma^{[d]} \circ \Gamma^{[d]} - \frac{1}{2}\Gamma^{[d]}}, 
$$

(16)

$$
\beta^{[d+1]} = \frac{X^{[d+1]}Y^\top u - X^{[d+1]}Y^\top YX^{[d+1]} \circ \beta^{[d]}}{\beta^{[d]}}, 
$$

(17)

lead to a monotonic decrease of the cost functional

$$
F(K, X, V, W, \beta) := KL(Y, KX) + \lambda \|X\|_1 + \mu \frac{1}{2} \|K\|_F^2 + \frac{\nu}{2} \|X\|_F^2 + \omega \|K\|_1 
$$

$$
+ \frac{\tau}{2} TV(K) + \frac{\sigma_{K,1}}{2} \|I - V^\top K\|_F^2 + \frac{\sigma_{K,2}}{2} \|V - K\|_F^2 
$$

$$
+ \frac{\sigma_{X,1}}{2} \|I - WX^\top\|_F^2 + \frac{\sigma_{X,2}}{2} \|W - X\|_F^2 + \frac{\rho}{2} \|u - YX^\top \beta\|^2.
$$

Fig. 2 Structure of MALDI imaging data: A mass spectrum is obtained at different positions of a tissue slice. The full dataset is a data cube, which can be visualized with different perspectives. Fixing a position of the tissue slice gives the mass spectrum at this position. Fixing a particular molecular weight reveals the distribution of molecules across the tissue slice with this weight.
It is easy to see that the classical, regularized NMF algorithms described in [11, 24, 25] can be regained by putting the corresponding regularization parameters to zero. In the case of $\ell_1$ and $\ell_2$-regularized NMF, this leads to the cost function

$$F(K, X) = KL(Y, KX) + \lambda \|X\|_1 + \frac{\mu}{2} \|K\|_F^2 + \frac{\nu}{2} \|X\|_F^2 + \omega \|K\|_1.$$ 

The classical update rule for $X$ is obtained by setting

$$\tilde{\Lambda}^{[d]} := X^{[d]} \circ \left( K^{[d+1]^T} \frac{Y}{K^{[d+1]}X^{[d]}} \right),$$

$$\tilde{\Gamma}^{[d]} := K^{[d+1]^T} 1_{n \times m} + \lambda 1_{p \times m},$$

which—in connection with the update rule for $X$ of the previous theorem—leads to

$$X^{[d+1]} = \sqrt{\frac{1}{\nu} \tilde{\Lambda}^{[d]} + \frac{1}{4\nu^2} \tilde{\Gamma}^{[d]} \circ \tilde{\Gamma}^{[d]} - \frac{1}{2\nu} \tilde{\Gamma}^{[d]}} + \frac{2\tilde{\Lambda}^{[d]}}{\tilde{\Gamma}^{[d]} + \sqrt{4\nu \tilde{\Lambda}^{[d]} + \tilde{\Gamma}^{[d]} \circ \tilde{\Gamma}^{[d]}}},$$

which is the update rule described in [11].

---

**Fig. 3** NMF of the rat brain dataset for $p = 6$. Orthogonality constraints on the channels with $\sigma_{K,1} = 1$ and $\sigma_{K,2} = 1$.
By the same approach and with the surrogate functionals derived in Section 4, we obtain the update rules for the Frobenius discrepancy term, i.e., we consider the functional

$$F(K, X, V, W, \beta) := \frac{1}{2} \| Y - KX \|_F^2 + \lambda \| X \|_1 + \frac{\mu}{2} \| K \|_F^2 + \frac{\nu}{2} \| X \|_2^2 + \omega \| K \|_1$$

$$+ \frac{\tau}{2} \text{TV}(K) + \frac{\sigma_{K,1}}{2} \| I - V^T K \|_F^2 + \frac{\sigma_{K,2}}{2} \| V - K \|_F^2$$

$$+ \frac{\sigma_X,1}{2} \| I - XW^T \|_F^2 + \frac{\sigma_X,2}{2} \| W - X \|_2^2 + \frac{\rho}{2} \| u - YX^T \beta \|^2.$$  

A monotonic decrease of this functional is obtained by the following iteration in combination with the update rules for $V, W, \beta$ as in Theorem 7 (see also Appendix 1.1 for more details on the derivation of these algorithms.)

$$K^{[d+1]} = \frac{YX^{[d]}^T + \tau \Psi \circ P(K^{[d]}) \circ Z(K^{[d]}) + (\sigma_{K,1} + \sigma_{K,2}) V^{[d+1]} \tau \Psi \circ P(K^{[d]}) + \sigma_{K,2} 1_{n \times p}}{K^{[d]}X^{[d]}X^{[d]} + \mu K^{[d]} + \omega I_{n \times p} + \sigma_{K,1} V^{[d+1]}V^{[d+1]}YK^{[d]}}$$

$$X^{[d+1]} = \frac{K^{[d+1]}Y + (\sigma_{X,1} + \sigma_{X,2}) W^{[d+1]} + \rho \beta^{[d]} u^TY}{\sigma_{X,2} 1_{p \times m} + K^{[d+1]}X^{[d]} + \lambda I_{p \times m} + \rho \beta^{[d]} \beta^{[d]} 1_{p \times m} 1_{p \times m}^T YX^T + \sigma_{X,1} X^{[d]} W^{[d+1]}W^{[d+1]}X^{[d]}}.$$  

Fig. 4 NMF of the rat brain dataset for $p = 6$. Orthogonality constraints on the channels with $\sigma_{K,1} = 200$ and $\sigma_{K,2} = 200$
6 MALDI Imaging

As a test case, we analyze MALDI imaging data (matrix-assisted laser desorption/ionization) of a rat brain. MALDI imaging is a comparatively novel modality, which unravels the molecular landscape of tissue slices and allows a subsequent proteomic or metabolic analysis [1, 6, 22]. Clustering this data reveals for example different metabolic regions of the tissue, which can be used for supporting pathological diagnosis of tumors.

The data used in this paper was obtained by a MALDI imaging experiment (see Fig. 2 for a schematic experimental setup).

In our numerical experiments, we used a classical rat brain dataset which has been used in several data processing papers before [2, 3, 22]. It constitutes a standard test set for hyperspectral data analysis.

The tissue slice was scanned at 20,185 positions. At each position, a full mass spectrum with 2974 \textit{m/z} (mass over charge) values was collected, i.e., instead of three color channels, as it is usual in image processing, this data has 2974 channels, each channel containing the spatial distribution of molecules having the same \textit{m/z} value.

![NMF of the rat brain dataset for \( p = 6 \). Orthogonality constraints on the channels with \( \sigma_{K,1} = 1 \) and \( \sigma_{K,2} = 1 \) and TV -penalty term with \( \tau = 0.4 \) and \( \varepsilon_{TV} = 10^{-7} \)](image)
The following numerical examples were obtained with the multiplicative algorithms described in the previous section. We just illustrate the effect of the different penalty terms for some selected functionals. One can display either the columns of $K$ as the pseudo channels of the NMF decomposition or the rows of $X$ as pseudo spectra characterizing the different metabolic processes present in the tissue slice (see the Figs. 3, 4, 5, and 6).

Both ways of visualization do have their respective value. Looking at the pseudo spectra in connection with orthogonality constraints leads to a clustering of the spectra and to a subdivision of the tissue slice in regions with potentially different metabolic activities (see [22]). Considering instead the different pseudo spectra, which were constructed in order to have a base which allows a low-dimensional approximation of the dataset, is the basis for subsequent proteomic analysis, e.g., one may target pseudo spectra where the related pseudo channels are concentrated in regions, which were annotated by pathological experts. Mass values which are dominant in those spectra may stem from proteins/peptides relating to biomarkers as indicators for certain diseases. Hence, classification schemes based on NMF decompositions have been widely investigated [26, 30, 34].

![Fig. 6 NMF of the rat brain dataset for $p = 6$. Orthogonality constraints on the channels with $\sigma_{K,1} = 10$ and $\sigma_{K,2} = 10$ and sparsity penalty term on $X$ with $\lambda = 0.06$. The sparsity penalty term has in connection with the orthogonality constraint a comparatively strong influence on the NMF computation: The sparsity in the spectra increases significantly and thus their biological interpretability, whereas the anatomic structure in the pseudo channels diminishes](image-url)
7 Conclusion

In this paper, we investigated methods based on surrogate minimization approaches for the solution of NMF problems. The interest in NMF methods is related to its importance for several machine learning tasks. Application for large datasets requires that the resulting algorithms are very efficient and that iteration schemes only need simple matrix-vector multiplications.

The state of the art for constructing appropriate surrogates is based on case-by-case studies for the different considered NMF models. In this paper, we embedded the different approaches in a general framework, which allowed us to analyze several extensions to the NMF cost functional, including \( \ell_1 \)- and \( \ell_2 \)-regularization, orthogonality constraints, and total variation penalties as well as extensions, which led to supervised NMF concepts.

Secondly, we analyzed surrogates in the context of the related iteration schemes, which are based on first-order optimality conditions. The requirement of separability as well as the need of having multiplicative updates, which preserve non-negativity without additional projections, was analyzed. This resulted in a general description of algorithms for alternating minimization of constrained NMF functionals. The potential of these methods is confirmed by numerical tests using hyperspectral data from a MALDI imaging experiment.

Several further directions of research would be of interest. First of all, besides the most widely used penalty terms discussed in this paper, further penalty terms, e.g., higher-order TV terms, could be considered. Secondly, construction principles for more general discrepancy terms could be analyzed (see also [16]).

Potentially more importantly, this paper contains only very first results for combining NMF constructions directly with subsequent classification tasks. The question of an appropriate surrogate functional for the supervised NMF model with logistic regression used in [26] remains unanswered and also the comparison with algorithmic alternatives such as ADMM methods needs to be explored.

Acknowledgements The authors want to thank Christine De Mol for her excellent presentations at several conferences and our joint discussions, which were the starting point for this paper. The presented results are based on the Master Thesis of the first author.

Funding Information This project was supported by the Deutsche Forschungsgemeinschaft (DFG) within the framework of GRK 2224/1 “\( \pi^3 \): Parameter Identification - Analysis, Algorithms, Applications”.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix 1: Details on the Derivation of the Algorithms in Section 5

In this section, we give a more detailed derivation of the algorithms presented in Section 5. We start with the less complex case of the Frobenius norm as discrepancy term and then turn to the Kullback–Leibler divergence. To cover both aspects, we derive the update rules.
of $X$ for the Frobenius discrepancy term and of $K$ in the case of the KLD. We will also take a closer look at the effect of $\kappa$ in (9) with respect to the LQBP construction principle.

1.1 Frobenius Norm

We consider the general cost function described in Section 5 for the case of the Frobenius norm. To compute the update rules for $X$, it is enough to examine the function

$$F(X) := \frac{1}{2} \|Y - KX\|_F^2 + \lambda \|X\|_1 + \nu \frac{\|X\|_2^2}{2},$$

where all terms independent from $X$ are omitted. Based on Remark 1 and following the discussion of Section 4.4, the construction of a surrogate for $F$ can be done separately for $F_1$ and the remaining penalty terms.

The construction of a surrogate for $F_1$ with the LQBP principle as it has been done similarly in Section 4.4 is essential. If we would use instead a surrogate for the discrepancy term $1/2 \|Y - KX\|_F^2$ from Section 4.1 or 4.2 and take the $\ell_1$-penalty term $\lambda \|X\|_1$ as surrogate itself, it is easy to see that this would not lead to multiplicative update rules. It is the $\ell_1$-penalty term which causes the difficulty. Computing the first-order optimality condition for the corresponding surrogate $\tilde{Q}_F(X, A) := \lambda \|X\|_1 + \tilde{Q}_F(X, A)$ with respect to $X$ would lead to

$$0 = \frac{\partial \tilde{Q}_F}{\partial X_{\xi \xi}}(X, A) = \lambda + \frac{\partial \tilde{Q}_F}{\partial X_{\xi \xi}}(X, A),$$

where the second term on the right-hand side does not depend on $\lambda$. Hence, we get a sign in front of $\lambda$ by solving the equation for $X_{\xi \xi}$ and we will not obtain multiplicative updates for $X$.

A correct surrogate is obtained by using the LQBP principle to $F_1$ and leads to

$$Q_F(X, A) := Q_{F_1}(X, A) + \nu \frac{\|X\|_2^2}{2} + Q_{\text{Orth}}(X, A) + Q_{\text{LR}}(X, A)$$

with

$$Q_{F_1}(X, A) = \sum_{j=1}^m f_{Y_{\ast, j}}(A_{\ast, j}) + \nabla f_{Y_{\ast, j}}(A_{\ast, j})^T (X_{\ast, j} - A_{\ast, j})$$

$$+ \frac{1}{2} \left( X_{\ast, j} - A_{\ast, j} \right)^T A_{f_{Y_{\ast, j}}(A_{\ast, j})} (X_{\ast, j} - A_{\ast, j}),$$

where $f_{Y_{\ast, j}} : \mathbb{R}^p_{\geq 0} \rightarrow \mathbb{R}$ is defined as

$$f_{Y_{\ast, j}}(x) := \frac{1}{2} \|Y_{\ast, j} - Kx\|^2 + \lambda \|x\|_1$$

and with the diagonal matrix

$$\Lambda_{f_{Y_{\ast, j}}(A_{\ast, j})} = \frac{\left( \nabla^2 f_{Y_{\ast, j}}(A_{\ast, j}) A_{\ast, j} \right)_k + \kappa_k}{A_{k j}} = \frac{(K^T K A_{\ast, j})_k + \kappa_k}{A_{k j}}.$$
The functionals $Q_{\text{Orth}}$ resp. $Q_{\text{LR}}$ are the surrogates obtained from Theorem 4 resp. Theorem 6. It will turn out that an appropriate choice of $\kappa_k$ will ensure a multiplicative NMF algorithm.

The computation of the first-order optimality condition for $Q_F$ leads to

$$0 = \frac{\partial Q_F}{\partial X_{\xi \zeta}}(X, A) = (K^T K A)_{\xi \zeta} - (K^T Y)_{\xi \zeta} + \lambda + \frac{(K^T K A)_{\xi \zeta} + \kappa_{\xi}}{A_{\xi, \zeta}}(X_{\xi \zeta} - A_{\xi \zeta})$$

$$+ \sigma_{X, 1} \sum_{k=1}^{p} W_{k \zeta} \left( \frac{X_{\xi \zeta}}{A_{\xi \zeta}} (A W^T)_{\xi k} - \delta_{\xi k} \right) + \sigma_{X, 2} (X_{\xi \zeta} - W_{\xi \zeta})$$

$$+ \rho \beta_{\xi} \sum_{i=1}^{n} Y_{i \zeta} \left( \frac{X_{\xi \zeta}}{A_{\xi \zeta}} (YA^T \beta)_{i} - u_i \right) + \nu X_{\xi \zeta}.$$

One can see immediately that the choice of $\kappa_{\xi} := \lambda$ for all $\xi \in \{1, \ldots, p\}$ is appropriate to get rid of the problematic term $\lambda$. Hence, we obtain

$$0 = -(K^T Y)_{\xi \zeta} + \frac{X_{\xi \zeta}}{A_{\xi \zeta}} ((K^T K A)_{\xi \zeta} + \lambda)$$

$$+ \sigma_{X, 1} \sum_{k=1}^{p} W_{k \zeta} \left( \frac{X_{\xi \zeta}}{A_{\xi \zeta}} (A W^T)_{\xi k} - \delta_{\xi k} \right) + \sigma_{X, 2} (X_{\xi \zeta} - W_{\xi \zeta})$$

$$+ \rho \beta_{\xi} \sum_{i=1}^{n} Y_{i \zeta} \left( \frac{X_{\xi \zeta}}{A_{\xi \zeta}} (YA^T \beta)_{i} - u_i \right) + \nu X_{\xi \zeta}.$$

Reordering the terms leads to

$$(K^T Y)_{\xi \zeta} + (\sigma_{X, 1} + \sigma_{X, 2}) W_{\xi \zeta} + \rho \beta_{\xi} (Y^T u)_{\xi}$$

$$= \frac{X_{\xi \zeta}}{A_{\xi \zeta}} ((K^T K A)_{\xi \zeta} + \nu A_{\xi \zeta} + \lambda + \rho \beta_{\xi} (Y^T Y A^T \beta)_{\xi} + \sigma_{X, 1} (AW^T W)_{\xi \zeta} + \sigma_{X, 2} A_{\xi \zeta}).$$

Solving for $X_{\xi \zeta}$ and extending the equation to the whole matrix $X$ yields finally

$$X = A \circ \frac{K^T Y + (\sigma_{X, 1} + \sigma_{X, 2}) W + \rho \beta u^T Y}{K^T K A + (\sigma_{X, 2} + \nu) A + \lambda 1_{p \times m} + \rho \beta \beta^T A Y^T Y + \sigma_{X, 1} A W^T W}.$$
Multiplying on both sides with $K_{\xi\zeta}$ and sorting the terms already give the system of quadratic equations mentioned in Section 5, namely

$$K_{\xi\zeta}^2 \left( \mu + \tau \psi_{\xi} P(A)_{\xi\zeta} + \frac{\sigma_{K,1}}{A_{\xi\zeta}} (VV^T A)_{\xi\zeta} + \sigma_{K,2} \right)$$

$$+ K_{\xi\zeta} \left( \sum_{j=1}^{m} X_{\xi j} + \omega - \tau \psi_{\xi} P(A)_{\xi\zeta} Z(A)_{\xi\zeta} - (\sigma_{K,1} + \sigma_{K,2}) V_{\xi\zeta} \right)$$

$$= A_{\xi\zeta} \sum_{j=1}^{m} \frac{Y_{\xi j}}{(AX)_{\xi j}} X_{\xi j}.$$ 

Taking into account that

$$m \sum_{j=1}^{n} X_{\xi j} (AX)_{\xi j} = \left( Y AX \right)_{\zeta},$$

we obtain the explicit solution of $K_{\xi\zeta}$ by completing the square and get

$$K_{\xi\zeta} = \left[ \frac{A_{\xi\zeta}}{\mu + \tau \psi_{\xi} P(A)_{\xi\zeta} + \frac{\sigma_{K,1}}{A_{\xi\zeta}} (VV^T A)_{\xi\zeta} + \sigma_{K,2}} \left( Y AX \right)_{\xi\zeta} \right]^{1/2}$$

$$+ \frac{1}{4} \left( \frac{(1_{n \times m} X)_{\xi\zeta} + \omega - \tau \psi_{\xi} P(A)_{\xi\zeta} Z(A)_{\xi\zeta} - V_{\xi\zeta} (\sigma_{K,1} + \sigma_{K,2})}{\mu + \tau \psi_{\xi} P(A)_{\xi\zeta} + \frac{\sigma_{K,1}}{A_{\xi\zeta}} (VV^T A)_{\xi\zeta} + \sigma_{K,2}} \right)^2$$

$$- \frac{1}{2} \left( \frac{(1_{n \times m} X^T)_{\xi\zeta} + \omega - \tau \psi_{\xi} P(A)_{\xi\zeta} Z(A)_{\xi\zeta} - V_{\xi\zeta} (\sigma_{K,1} + \sigma_{K,2})}{\mu + \tau \psi_{\xi} P(A)_{\xi\zeta} + \frac{\sigma_{K,1}}{A_{\xi\zeta}} (VV^T A)_{\xi\zeta} + \sigma_{K,2}} \right).$$

This equation holds for arbitrary $\xi \in \{1, \ldots, n\}$ and $\zeta \in \{1, \ldots, p\}$. We therefore can extend this relation to the whole matrix $K$ and obtain

$$K = \left[ \frac{A}{\mu 1_{n \times p} + \tau \Psi \circ P(A) + \sigma_{K,1} \frac{VV^T A}{A} + \sigma_{K,2} 1_{n \times p}} \right] \circ \left( Y AX \right)^{1/2}$$

$$+ \frac{1}{4} \left( \frac{1_{n \times m} X^T + \omega 1_{n \times p} - \tau \Psi \circ P(A) \circ Z(A) - (\sigma_{K,1} + \sigma_{K,2}) V}{\mu 1_{n \times p} + \tau \Psi \circ P(A) + \sigma_{K,1} \frac{VV^T A}{A} + \sigma_{K,2} 1_{n \times p}} \right)^2$$

$$- \frac{1}{2} \left( \frac{1_{n \times m} X^T + \omega 1_{n \times p} - \tau \Psi \circ P(A) \circ Z(A) - (\sigma_{K,1} + \sigma_{K,2}) V}{\mu 1_{n \times p} + \tau \Psi \circ P(A) + \sigma_{K,1} \frac{VV^T A}{A} + \sigma_{K,2} 1_{n \times p}} \right),$$

which is exactly the described update rule in Section 5.
Appendix 2: Kullback–Leibler Divergence Discrepancy and LQBP

In this section, we will use the LQBP construction principle to derive a multiplicative algorithm for the cost function

\[ F(X) := KL(Y, KX) = \sum_{j=1}^{m} KL(Y_{\bullet,j}, KX_{\bullet,j}) = \sum_{j=1}^{m} fY_{\bullet,j}(X_{\bullet,j}). \]

Similar to the approach in Appendix 1.2, we define according to the LQBP principle the surrogate

\[ Q_F(X, A) = \sum_{j=1}^{m} fY_{\bullet,j}(A_{\bullet,j}) + \nabla fY_{\bullet,j}(A_{\bullet,j})^T(X_{\bullet,j} - A_{\bullet,j}) \]

\[ + \frac{1}{2} (X_{\bullet,j} - A_{\bullet,j})^T A fY_{\bullet,j}(A_{\bullet,j})(X_{\bullet,j} - A_{\bullet,j}) \]

with the diagonal matrix

\[ \Lambda fY_{\bullet,j}(A_{\bullet,j})_{kk} = \frac{(\nabla^2 fY_{\bullet,j}(A_{\bullet,j}) A_{\bullet,j})_k + \kappa_k}{A_{kj}}. \]

It follows for the partial derivatives of \( f \)

\[ \frac{\partial fY_{\bullet,\zeta}}{\partial X_{\beta,\zeta}}(X_{\bullet,\zeta}) = -\sum_{i=1}^{n} \frac{Y_{i,\zeta} K_{i,\beta}}{(KX)_{i,\zeta}} + K_{i,\beta}, \]

\[ \frac{\partial^2 fY_{\bullet,\zeta}}{\partial X_{\alpha,\zeta} \partial X_{\beta,\zeta}}(X_{\bullet,\zeta}) = \sum_{i=1}^{n} \frac{Y_{i,\zeta} K_{i,\alpha} K_{i,\beta}}{(KX)_{i,\zeta}^2}. \]

The first-order optimality condition of the surrogate functional leads then to

\[ 0 = \frac{\partial Q_F}{\partial X_{\xi,\zeta}}(X, A) = -\sum_{i=1}^{n} \frac{Y_{i,\zeta} K_{i,\xi}}{(KA)_{i,\xi}} + \sum_{i=1}^{n} K_{i,\xi} + \sum_{i=1}^{n} \frac{Y_{i,\zeta} K_{i,\xi}}{(KA)_{i,\xi}} + \kappa_{\xi}(X_{\xi,\zeta} - A_{\xi,\zeta}). \]

Setting \( \kappa_{\xi} := \sum_{i=1}^{n} K_{i,\xi} \) and solving for \( X_{\xi,\zeta} \) leads finally to the multiplicative update rule

\[ X^{[d+1]} = \frac{2X^{[d]}}{K^{[d+1]T}Y + K^{[d+1]T}I_{n \times m}} o K^{[d+1]T} Y K^{[d+1]} X^{[d]}, \]

which differs from the classical update rule for the KLD described in [11, 24, 25].

Appendix 3: Surrogate of the TV Penalty—Separability

In this section, we will prove the separability of the surrogate functional

\[ Q_{TV}(K, A) = \sum_{k=1}^{p} \psi_k \sum_{i=1}^{n} \frac{1}{|\nabla i_k A|} \left( \epsilon_{TV}^2 + \sum_{i \in N_i} K_{i,k}^2 + K_{\ell,k}^2 - K_{i,k}(A_{i,k} + A_{\ell,k}) \right. \]

\[ - K_{\ell,k}(A_{i,k} + A_{\ell,k}) + A_{i,k}^2 + A_{\ell,k}^2 \] (18)

described in Theorem 5. Furthermore, we choose an arbitrary \( s \in \{1, \ldots, n\} \) and \( t \in \{1, \ldots, p\} \). The aim is now to find all terms in (18) with \( K_{st}^2 \) and \( K_{st} \).
To find all quadratic terms $K_{st}^2$ in (18), we see that we have to fix the index $k$, such that $k = t$. The remaining indices in (18), which have to be analyzed, are $i$ and $\ell$.

For the case $i = s$, we find that the preceding coefficient is

$$\frac{\psi_t}{|\nabla_{st} A|} \sum_{r \in N_s} 1.$$ 

The case $\ell = s$ can only occur for those indices $i$, which satisfy $s \in N_i$. The definition of the adjoint neighborhood pixels gives

$$\forall i : s \in N_i \iff \forall i : i \in \tilde{N}_s.$$ 

Therefore, the corresponding preceding coefficient is here

$$\frac{\psi_t}{|\nabla_{rt} A|} \sum_{r \in \tilde{N}_s} 1,$$

Altogether, we obtain for the quadratic terms $K_{st}^2$ the coefficient

$$\tilde{P}_{st}(A) := \psi_t \left( \frac{1}{|\nabla_{st} A|} \sum_{r \in N_s} 1 + \sum_{r \in \tilde{N}_s} \frac{1}{|\nabla_{rt} A|} \right),$$

such that $\tilde{P}_{st}(A) \cdot K_{st}^2$ takes all quadratic terms of the matrix entries of $K$ in the surrogate functional into account.

The same can be done with the linear terms $K_{st}$, which leads to the coefficient

$$\tilde{Z}_{st}(A) := -\psi_t \left( \frac{1}{|\nabla_{st} A|} \sum_{r \in N_s} [A_{st} + A_{rt}] + \sum_{r \in \tilde{N}_s} \frac{A_{st} + A_{rt}}{|\nabla_{rt} A|} \right).$$

Therefore, the surrogate $Q_{TV}$ can be written as

$$Q_{TV}(K, A) = \sum_{t=1}^{p} \sum_{s=1}^{n} \left[ \tilde{P}_{st}(A) \cdot K_{st}^2 + \tilde{Z}_{st}(A) \cdot K_{st} \right] + \tilde{C}(A)$$

for some function $\tilde{C}$, which only depends on $A$. This shows the separability of the surrogate.

References

REGULARIZED ORTHOGONAL NONNEGATIVE MATRIX FACTORIZATION AND K-MEANS CLUSTERING

A PREPRINT

Pascal Fernsel
Center for Industrial Mathematics
University of Bremen
Bremen, Germany
p.fernsel@uni-bremen.de

Peter Maaß
Center for Industrial Mathematics
University of Bremen
Bremen, Germany
pmaass@uni-bremen.de

December 15, 2021

ABSTRACT

In this work, we focus on connections between K-means clustering approaches and Orthogonal Nonnegative Matrix Factorization (ONMF) methods. We present a novel framework to extract the distance measure and the centroids of the K-means method based on first order conditions of the considered ONMF objective function, which exploits the classical alternating minimization schemes of Nonnegative Matrix Factorization (NMF) algorithms. While this technique is characterized by a simple derivation procedure, it can also be applied to non-standard regularized ONMF models. Using this framework, we consider in this work ONMF models with $\ell_1$ and standard $\ell_2$ discrepancy terms with an additional elastic net regularization on both factorization matrices and derive the corresponding distance measures and centroids of the generalized K-means clustering model. Furthermore, we give an intuitive view of the obtained results, examine special cases and compare them to the findings described in the literature.

Keywords: Orthogonal nonnegative matrix factorization · K-means Clustering · Distance function · Centroid · Elastic net regularization

AMS subject classifications: 15A23 · 65F22 · 62H30

1 Introduction

Cluster analysis has been studied extensively for more than five decades by the machine learning community and has numerous applications in computer science, social science, biology, medicine and many other fields. Clustering in its classical form is an unsupervised learning technique and is a main task of exploratory data analysis. The most well-known clustering algorithms are based on either partitional or hierarchical clustering techniques. Partitional clustering methods are iterative approaches and try to optimize an objective function, while hierarchical clustering algorithms typically develop a binary tree-based data structure to obtain the desired clustering.

In this work, we focus on the K-means clustering, which is the most widely used partitional clustering algorithm, and analyze its relationships to different Nonnegative Matrix Factorization (NMF) models, which are a specific matrix factorization method with additional nonnegativity constraints. The clustering capability of NMF approaches are well-known throughout the literature and the study of the connections between both approaches leads to several different NMF models for clustering, which typically have advantages compared with usual K-means approaches. However, the corresponding works only consider non-regularized NMF models and mainly establish these connections by directly comparing the objective functions and the constraints of the optimization problems of both the NMF problem as well as the K-means model.

This work rather focuses on obtaining relationships between regularized Orthogonal NMF (ONMF) and generalized K-means models by deriving the corresponding distance measures and centroids of the K-means approach via the typ-
Regularized ONMF and $K$-means Clustering

A PREPRINT

The main task of a clustering method is to partition a given set of objects into groups, such that objects within a group are more similar to each other than objects from different groups. In a mathematical framework, this can be formulated as follows: Partition a given index set \( \{1, \ldots, M\} \) of a corresponding dataset \( \{x_m \in \tilde{X} \mid m = 1, \ldots, M\} \) into \( K \) disjoint sets \( \mathcal{T}_k \subset \{1, \ldots, M\} \), such that \( \cup_{k=1,\ldots,K} \mathcal{T}_k = \{1, \ldots, M\} \), where \( \tilde{X} \) is a non-empty set. Note that in this formulation, the property of similarity is still needed to be defined properly.

\( K \)-means clustering ranks upon the best known partitional clustering algorithms and is also the focus of this work. Following [2], the formal definition of \( K \)-means clustering can be formulated as follows: Let \((\tilde{X}, \text{dist})\) be a metric space with \( \text{dist}(\cdot, \cdot) \) being a distance function over \( \tilde{X} \). Furthermore, we define \( \text{dist}(\mathcal{X}, y) := \min_{x \in \mathcal{X}} \text{dist}(x, y) \) for \( \mathcal{X} \subset \tilde{X} \) and \( y \in \tilde{X} \).

**Definition 2.1** (K-median and \( K \)-means clustering). A clustering of a given set \( \mathcal{X} \subset \tilde{X} \) is a partition based on the set of centroids \( \mathcal{C} := \{c_1, \ldots, c_K\} \subset \tilde{X} \), so that each point in \( \mathcal{X} \) is associated to its nearest centroid \( c_k \). We assume that each point \( x \in \mathcal{X} \) is associated to a weight \( w(x) \in \mathbb{R} \). The cost of \( K \)-median clustering and \( K \)-means clustering of \( \mathcal{X} \) by \( \mathcal{C} \) is defined as \( \psi(\mathcal{C}, \mathcal{X}) := \sum_{x \in \mathcal{X}} w(x) \text{dist}(\mathcal{C}, x) \) and \( \omega(\mathcal{C}, \mathcal{X}) := \sum_{x \in \mathcal{X}} w(x) \text{dist}(\mathcal{C}, x)^2 \) respectively. The corresponding metric \( K \)-median (resp. \( K \)-means) problem is to find a set of centroids \( \mathcal{C} \subset \tilde{X} \), such that the cost \( \psi(\mathcal{C}, \mathcal{X}) \) and \( \omega(\mathcal{C}, \mathcal{X}) \) is minimized.

In this setting, the so-called clusters of the clustering are the disjoint sets \( \mathcal{T}_k \) consisting of the points \( x \in \mathcal{X} \) which are associated to the corresponding centroid \( c_k \). Different from [2], we do not constrain in Definition 2.1 the set of centroids to be in \( \mathcal{X} \). In the remainder of this work, we will assume that the set \( \mathcal{X} \) is defined by \( \mathcal{X} := \{x_m \in \mathbb{R}^n \mid m = 1, \ldots, M\} \).

Furthermore, we note that according to Definition 2.1, the distance function dist is a metric of the metric space \( \mathcal{X} \). However, \( K \)-means clustering algorithms oftentimes are also used with similarity measures such as the cosine similarity, which do not satisfy the properties of a metric. Moreover, it will turn out that the obtained generalized
distance measures in this work will also not satisfy the usual properties of a metric (see Section 3 for more details). However, we will refer to it as a distance measure \( \text{dist}(\cdot, \cdot) \) in the remainder of this work.

Thus, \( K \)-means clustering consists of two basic ingredients: the distance function \( \text{dist}(\cdot, \cdot) \), which gives the needed similarity property of the clustering method, as well as the centroids in \( C \). The standard choice of the distance function for the classical \( K \)-means problem is the Euclidean distance \( \text{dist}(x, y) := \|x - y\|_2 \) leading to the typical minimization of the within-cluster variances given by

\[
\min_{I_1, \ldots, I_K} \sum_{k=1}^K \sum_{m \in I_k} \|x_m - c_k\|_2^2. \tag{1}
\]

It can be shown, that this problem is NP-hard [17]. Hence, heuristic approaches are commonly used to find approximate solutions. The most widely known method is the \( K \)-means algorithm and is based on an alternating minimization.

In a first step, after a suitable initialization of the centroids \( C \), the data points \( x_m \) are associated to the nearest centroid \( c_k \). Afterwards, in the case of the Euclidean distance, the centroids \( c_k \) are recomputed based on the mean of the associated points \( x_m \) for \( m \in I_k \). This process is repeated until the cluster assignments do not change anymore.

In order to see the connection to matrix factorization problems, it is needed to formulate the objective function in (1) in a vectorized form. To do so, we write the data points \( x_m \) row-wise into a data matrix \( X \in \mathbb{R}^{M \times N} \) so that \( X := [x_1, \ldots, x_M]^T \). Moreover, we define the so-called cluster membership matrix \( B \in \{0, 1\}^{M \times K} \), which is given by

\[
B_{mk} := \begin{cases} 0 & \text{if } m \notin I_k, \\ 1 & \text{if } m \in I_k. \end{cases}
\]

Note that since the sets \( I_k \) are disjoint to each other, each row of \( B \) has exactly one non-zero element and gives the needed interpretation of a hard clustering. Hence, the columns of \( B \) are orthogonal to each other, i.e. it holds that \( \langle B_{\bullet, k}, B_{\bullet, \ell} \rangle_2 = 0 \) for \( k \neq \ell \). Furthermore, note that \( |I_k| = \|B_{\bullet, k}\|_1 = \|B_{\bullet, k}\|_2 \). In addition, we introduce the diagonal matrix \( D := \text{diag}(1/|I_1|, \ldots, 1/|I_K|) \in \mathbb{R}^{K \times K} \). By using these matrices, we see that \( DB^T X \in \mathbb{R}^{K \times N} \) yields a matrix which has the centroids \( c_k \), arranged in its rows. In this way, it is possible to rewrite the objective function in (1) as \( \|X - BDB^T X\|_F^2 \) leading to the minimization problem

\[
\min_{B \in \{0, 1\}^{M \times K}, (DB^{1/2})^T BD^{1/2} = I_K} \|X - BDB^T X\|_F^2, \tag{2}
\]

where \( D^{1/2} \) is defined component-wise by \( D_{k,k}^{1/2} := \sqrt{D_{k,k}} \) and with \( I_{K \times K} \) being the identity matrix of size \( K \times K \).

A trivial solution to this problem could be to choose \( K = M \) and \( B = I_{M \times M} \), which would correspond to assign each observation to its own cluster. However, this is obviously not the aim of a clustering method so that usually \( K \leq \min\{M, N\} \) is chosen.

The formulation in (2) makes the relationship between the clustering problem and a matrix factorization problem more clear. By omitting the constraints in the minimization problem (2) and additionally assuming that a nonnegative data matrix \( X \geq 0 \) is given, we automatically have that \( DB^T X \geq 0 \). This gives rise to the so-called Nonnegative Matrix Factorization (NMF) problem \( X \approx UV \) of the data matrix, so that the factorization matrices \( U \in \mathbb{R}^{M \times K} \) and \( V \in \mathbb{R}^{K \times N} \) of the NMF can be compared to the cluster membership matrix \( B \) and the so-called centroid matrix \( DB^T X \) respectively. In the following, we give a definition of the general NMF problem.

**Definition 2.2** (NMF). For a given data matrix \( X \in \mathbb{R}^{M \times N}_{\geq 0} \) and a factorization rank \( K \leq \min\{M, N\} \), the aim is to find two matrices \( U \in \mathbb{R}^{M \times K}_{\geq 0} \) and \( V \in \mathbb{R}^{K \times N}_{\geq 0} \), such that

\[
X \approx UV = \sum_{k=1}^K U_{\bullet, k} V_{k, \bullet}. \tag{3}
\]

NMF was originally introduced by Paatero and Tapper [18] in 1994 as positive matrix factorization. Different from the widely-known Principal Component Analysis (PCA), NMF constraints the factorization matrices to be non-negative. This allows a parts-based representation of the whole dataset, since each row \( X_{m, \bullet} \) and column \( X_{\bullet, n} \) can be represented as a superposition of the few basis vectors \( V_{k, \bullet} \) and \( U_{\bullet, k} \) so that \( X_{m, \bullet} \approx \sum_{k} V_{k, \bullet} U_{\bullet, k} \) and \( X_{\bullet, n} \approx \sum_{k} U_{m, k} V_{k, \bullet} \). This property makes the NMF the ideal tool for nonnegative data, since the interpretability of the factorization matrices is ensured due to the additional nonnegativity constraint. NMF has been extensively used as a feature extraction and data representation tool [8, 14] as well as for clustering [7], compression [24] or even for
solving inverse problems, where the NMF can be used as a joint reconstruction and feature extraction method [1]. Possible application fields include document clustering [11, 19], medical imaging [8, 14, 7, 1], hyperspectral unmixing [10, 5, 6] and music analysis [9] to name just a few.

The typical approach to find an approximate solution for the NMF is based on a variational formulation of the problem. Thus, the NMF is reformulated as a minimization problem with a suitable discrepancy term \( D(\cdot, \cdot) \), which is typically chosen according to the noise distribution of the data. Furthermore, NMF problems are usually ill-posed due to the non-uniqueness of the solution [12, 20]. Hence, suitable regularization terms \( R_j(\cdot) \) are typically added to the NMF cost function to tackle the ill-posedness of the problem and to enforce additional properties of the factorization matrices.

Hence, the general minimization problem of the NMF can be written as

\[
\min_{U, V \geq 0} D(X, UV) + \sum_{j=1}^{J} \alpha_j R_j(U, V) =: \min_{U, V \geq 0} F(U, V),
\]

where \( \alpha_j \geq 0 \) are the regularization parameters, which control the influence of the penalty terms \( R_j(\cdot) \). Typical choices for discrepancy terms are the Frobenius norm in the case of Gaussian noise, the Kullback-Leibler divergence for Poisson noise, the \( \ell_1 \) norm or other divergences. Regarding the penalty terms, common choices are the \( \ell_2 \) and \( \ell_1 \) regularization, which is also used in this work (see Section 3). Further possibilities are more problem specific and include total variation regularization and terms which enforce orthogonality of the factorization matrices or even allow a supervised classification framework [8, 14, 1, 7].

For usual choices of \( D \) and \( R_j \), the corresponding objective function \( F \) is convex in each of the variables \( U \) and \( V \) but non-convex in \((U, V)\). This motivates to consider alternating minimization schemes similar to the K-means algorithm discussed above leading to the update rules

\[
U^{[i+1]} = \arg \min_{U \geq 0} F(U, V^{[i]}),
\]

\[
V^{[i+1]} = \arg \min_{V \geq 0} F(U^{[i+1]}, V).
\]

For a review on the algorithm development of multiplicative updates for a variety of discrepancy and regularization terms, we refer the reader to [8].

Regarding the clustering capability of NMF and comparing \( U \) to the cluster membership matrix \( B \) of the K-means approach, the classical NMF problem with only the nonnegativity constraint on the matrices does not provide the needed hard clustering interpretability on \( U \), since the matrix can contain multiple non-zero entries each of its rows. One typical approach to ensure this property on \( U \) is to additionally require the matrix to be column-wise orthogonal by adding the hard constraint

\[
(U_{\bullet,k}, U_{\bullet,l})_2 = 0 \quad \text{for} \ k \neq l.
\]

This leads to the problem of Orthogonal NMF (ONMF). Occasionally, further constraints like the normalization of the columns \( U_{\bullet,k} \) leading to

\[
U^\top U = I_{K \times K}
\]

can be enforced, which will be also discussed in this work. These constraints indeed yields the desired interpretability of \( U \) as a cluster membership matrix, since the nonnegativity constraint together with (7) ensure that every row of \( U_{m,\bullet} \) only contains at most one non-zero entry \( U_{m,k} > 0 \) indicating the association of the data point \( X_m,\bullet \) to the cluster \( I_k \). Thus, the clusters can also be written as

\[
I_k = \{m \in \{1, \ldots, M\} \mid U_{mk} > 0\}.
\]

In this setting, the matrix \( V \) can then be interpreted as the centroid matrix, which contains the centoids in its rows. Throughout the literature, many relationships between different kinds of K-means and ONMF models could be shown. One of the pioneering works is the one by Ding et al. in [3], which describes equivalences between Kernel K-means and symmetric NMF \( X \approx U U^\top \), bipartite graph K-means clustering and the bi-orthogonal NMF problem given by the minimization problem

\[
\min_{U, V \geq 0} \|X - UV\|_F^2, \quad \text{s.t.} \ U^\top U = I_{K \times K}, \ V V^\top = I_{K \times K},
\]

as well as the classical K-means clustering and ONMF with the stronger constraint on \( U \) by considering \( U_{mk} = 1/\sqrt{|I_k|} \) for \( m \in I_k \) and \( U_{mk} = 0 \) otherwise. Numerous other works followed with relationships between Nonnegative Matrix Tri-Factorizations and simultaneous row and column clustering approaches with applications to document clustering problems as well as connections between relaxed K-means clustering models and semi-NMF, convex NMF
and Kernel NMF [15, 4]. Furthermore, [21] shows the equivalence between a weighted variant of spherical $K$-means and the ONMF model with (8) as a hard constraint on $U$. Further results include the clustering interpretability of sparse NMF [11] and the relation of projective NMF to $K$-means clustering [25].

Besides of the theoretical interest to study these relationships, they also have some practical relevance since NMF models can have several advantages over classical $K$-means clustering methods. For instance, NMF models can do both hard as well as soft clustering and are able to perform a clustering of the rows and columns simultaneously. For more information on the clustering capabilities of NMF approaches, the relationships to $K$-means clustering and the development of algorithms, we refer the interested reader to both survey articles [16, 22].

However, these works do not consider any regularization terms in their NMF models, which is the focus of this work. Furthermore, the derivation of these relationships in the works throughout the literature are mostly based on the comparison of the objective functions and the constraints of the considered $K$-means and NMF model. This is in contrast to the approach used in this work, where the focus lies on obtaining connections between regularized ONMF and generalized $K$-means models by directly deriving the distance measures and centroids of the $K$-means approach based on the considered ONMF model. The used derivation framework exploits the typical alternating minimization scheme of NMF algorithms in (5) and (6) and uses first-order conditions of the ONMF objective function. This technique offers a significantly simpler method to derive connections between $K$-means and NMF compared to the ones used throughout the literature. Furthermore, it can be generalized to regularized ONMF models and is able to directly extract the distance measure and centroids of the $K$-means approach. In Section 3, we consider a regularized ONMF model with an elastic net regularization on both matrices $U$ and $V$. Hence, from a viewpoint of the regularization theory in inverse problems, the used framework also allows to see the effect of the $\ell_1$ and $\ell_2$ penalty terms on the obtained distance measures and the centroids of the $K$-means method.

Another motivational aspect concerns the distance measure for $K$-means clustering. It is well-known that an appropriate choice of the distance measure depending on the considered application is vital for the performance of the clustering algorithm. However, it is also known that $K$-means clustering algorithms typically suffer from the so-called uniformization effect, i.e. the algorithm tends to produce clusters with relatively balanced sizes. The work [23] could show that for a so-called $K$-means distance, which is a generalization of the Bregman divergence, the usual $K$-means algorithm suffer from the uniformization effect. However, the presented framework in this paper also allows to derive generalized $K$-means models with distance measures, which do not belong to the family of Bregman divergences and could lead to $K$-means approaches, which do not suffer from this negative effect.

### 3 Regularized ONMF and $K$-means Clustering

In this section, we introduce the considered ONMF problem and study its relation to generalized $K$-means models by deriving the corresponding distance measures and centroids. Regarding the ONMF model, we consider the $\ell_1$ norm and the $\ell_2$ norm for the discrepancy term and an elastic net regularization on $U$ and $V$ in each case. Hence, the objective functions of the considered ONMF models are

\[
F_1(U, V) := \|X - UV\|_1 + \lambda_U \|U\|_F^2 + \lambda_V \|V\|_F^2,
\]

\[
F_2(U, V) := \|X - UV\|_2^2 + \lambda_U \|U\|_F^2 + \lambda_V \|V\|_F^2,
\]

where $X := [x_1, \ldots, x_M] \in \mathbb{R}_{\geq 0}^{M \times N}$ is a given data matrix with data points $x_m \in \mathbb{R}^N$, so that $X_{m,*} = (x_m)^\top$. Furthermore, $U := [u_1, \ldots, u_K] \in \mathbb{R}_{\geq 0}^{M \times K}$ and $V := [v_1, \ldots, v_K] \in \mathbb{R}_{\geq 0}^{K \times N}$ are the factorization matrices of the ONMF problem with the short notations $u_k := U_{*,k}$ and $v_k := (V_{k,*})^\top$. Moreover, $\lambda_U, \lambda_V, \mu_U, \mu_V \geq 0$ are the regularization parameters and control the influence of the corresponding penalty terms in the objective function. In the most general case, we consider the ONMF minimization problem

\[
\min_{U, V \geq 0} F_i(U, V)
\]

s.t. (C1) holds

\[
(u_k, u_l)_2 = 0 \quad \text{for} \quad k \neq l
\]

for $i \in \{1, 2\}$ with the additional constraint (C1) on $U$ to ensure the needed clustering interpretability as a cluster membership matrix as described in Section 2. Occasionally, we will constrain the matrix $U$ further than in (C1) to discuss special cases (see Section 3.1 and Section 3.4).
3.1 Alternating Minimization and Separability

In this section, we describe the basic strategy to derive the connections between the ONMF and K-means models and introduce some basic tools used in the following Sections.

As described in Section 2, the general framework used in this work is based on the alternating minimization scheme showed in (5) and (6). As the matrix $V$ can be compared to a centroid matrix in the setting of the considered ONMF model (see Section 2), the minimization of the ONMF problem in (12) with respect to $V$ for fixed $U$ leads to the characterization of the centroid of the K-means approach. Furthermore, it will turn out that the distance measure of the K-means method can be identified via the minimization of (12) with respect to $U$ for fixed $V$. However, in order be able to compute the solutions of both minimization problems, the objective functions $F_i$ need to satisfy specific separability properties.

In the case of the minimization with respect to $V$, it is easy to see that both $F_i$ can be written as

$$F_i(U, V) = \sum_{k=1}^{K} D_i(X_k, \tilde{u}_k v_k^T) + R(u_k, v_k),$$

where $X_k \in \mathbb{R}_{\geq 0}^{I_k \times N}$ denotes the submatrix containing the rows of $X$ with the indices in $I_k$ given by the relationship in (9). Furthermore, $\tilde{u}_k \in \mathbb{R}_{\geq 0}^{I_k}$ denotes the corresponding reduced column vector of $u_k$ of non-zero entries. Hence, the minimization of the $F_i$ with respect to $V$ separates with the rows $v_k$. Therefore, the centroids $v_k$ of the K-means approach can be computed by solving the minimization problems

$$\min_{v_k \geq 0} D_i(X_k, \tilde{u}_k v_k^T) + \mu V ||v_k||_2^2 + \lambda V ||v_k||_1,$$

(13)

since the terms independent from $V$ can be omitted. We can further reduce the minimization problem to a set of scalar minimization problems, since

$$D_i(X_k, \tilde{u}_k v_k^T) + R(u_k, v_k) = \sum_{n=1}^{N} \sum_{m \in I_k} D_i(X_{mn}, U_{mk} V_{kn}) + \mu V V_{kn}^2 + \lambda V |V_{kn}|.$$

This leads to the scalar minimization problems

$$\min_{V_{kn} \geq 0} \sum_{m \in I_k} D_i(X_{mn}, U_{mk} V_{kn}) + \mu V V_{kn}^2 + \lambda V |V_{kn}| =: \min_{V_{kn} \geq 0} \varphi_{kn}(V_{kn})$$

(14)

with the objective functions

$$\varphi_{kn}(t) = \sum_{m \in I_k} D_i(X_{mn}, U_{mk} t) + \lambda V |t| + \mu V t^2,$$

(15)

which are considered in the following Sections 3.2 and 3.3.

Regarding the minimization task with respect to $U$ for fixed $V$, the problem also separates for the individual entries of $U$. Due to the constraint in (C1) in combination with the nonnegativity of $U$, each row of $U$ has at most one positive entry. Thus, the $m$-th row of $UV$ is a multiple of one of the rows of $V$ and is compared in the discrepancy term with $X_{m, \bullet}$. Hence, the objective functions $F_i$ can be written as

$$F_i(U, V) = \sum_{m=1}^{M} \left[ D_i(X_{m, \bullet}, U_{m, \pi(m)} V_{\pi(m), \bullet}) + \lambda U |U_{m, \pi(m)}| + \mu U U^2_{m, \pi(m)} \right] + \lambda V \| V \|_1 + \mu V \| V \|_2^2,$$

where $\pi(m) \in \{1, \ldots, K\}$ denotes the index of the non-zero entry in the $m$-th row of $U$ assuming that in each row of $U$, there exists exactly one non-zero entry $U_{m, \pi(m)} > 0$. This motivates to consider the minimization problems

$$\min_{U_{m, \pi(m)} > 0} D_i(X_{m, \bullet}, U_{m, \pi(m)} V_{\pi(m), \bullet}) + \mu U U^2_{m, \pi(m)} + \lambda U |U_{m, \pi(m)}| =: \min_{U_{m, \pi(m)} > 0} \psi_{m, \pi(m)}(U_{m, \pi(m)})$$

(16)

with the objective functions

$$\psi_{m, \pi(m)}(t) = D_i(X_{m, \bullet}, t V_{\pi(m), \bullet}) + \mu t^2 + \lambda U |t|$$

(17)

omitting the terms independent from $U_{m, \pi(m)}$. This minimization problem characterizes the distance measure of the K-means approach and yields

$$\text{dist}(x_m, v_{\pi(m)}) = \left( \min_{t \geq 0} \psi_{m, \pi(m)}(t) \right)^{1/i},$$

(18)
where the index $i$ specifies the considered objective function $F_i$. The index $\pi(m)$, which indicates the non-zero entry $U_{i,m,\pi(m)}$, can be computed by comparing the $m$-th data vector $x_m$ with all centroids of $v_k$ leading to

$$\pi(m) = \arg \min_{k \in \{1,...,K\}} \text{dist}(x_m, v_k).$$

The value of the non-zero entry in the $m$-th row of the matrix $U$ is finally given by

$$U_{m,\pi(m)} = \arg \min_{t \geq 0} \psi_{m,\pi(m)}(t).$$

We note that both scalar minimization problems in (14) and (16) are strictly convex for $\mu_V > 0$ and $\mu_U > 0$ respectively due to the quadratic penalty term of the elastic net regularization.

The above described relationships between the ONMF model and the distance function as well as the centroid of the $K$-means approach can be seen more intuitively by considering the additional constraint $i = 2$, $U_{mk} \in \{0,1\}$ and $\lambda_U = \lambda_V = \mu_U = \mu_V = 0$ as a special case of the ONMF model. For the minimization with respect to $V$, this yields the minimization problem $\min_{v_k > 0} \|X_k - \tilde{u}_k v_k^T\|_F^2$. Using the zero gradient condition, this leads to

$$v_k = \sum_{m \in I_k} x_m / |I_k|,$$

i.e. the mean of the data points of the index set given in $I_k$. For the minimization problem with respect to $U$, we consider (18) with the additional constraint $t \in \{0,1\}$ leading to the usual Euclidean distance, i.e. $\text{dist}(x_m, v_k) = \|x_m - v_k\|_2$. Hence, this ONMF model leads to the classical $K$-means approach with the Euclidean distance function and mean centroids.

For the subsequent analysis, we introduce the notion of a generalized median function. To do so, we consider the vectors $v, w \in \mathbb{R}^N$ and $\lambda, \mu \geq 0$ along with the continuous and convex function

$$f(t) = \sum_{n=1}^{N} |v_n - w_n t| + \mu t^2 + \lambda |t|.$$  \hspace{1cm} (19)

It is easy to see that minimizers of $f$ exist and that the set of minimizers $\arg \min_t f(t)$ either contains a unique element or is a closed interval. For both cases, we define the weighted, regularized median

$$\text{med}_{w}^{f_{1}(\lambda), f_{2}(\mu)}(v) \in \arg \min_{t \in \mathbb{R}} f(t) \subset \mathbb{R}$$  \hspace{1cm} (20)

as the midpoint of the set of minimizers given by $\arg \min_t f(t)$. Note that for $\lambda = \mu = 0$ and $w_n = 1$ for all $n \in \{1,\ldots,N\}$, the weighted regularized median is the classical median, so that $\text{med}_{w}^{f_{1}(\lambda), f_{2}(\mu)}(v) = \text{med}(v)$.

### 3.2 Classical Discrepancy Term

In this section, we derive the distance measures and the centroids based on the ONMF problem in (12) for $i = 2$. For the minimization with respect to $V$, we consider the scalar minimization problem

$$\min_{t \geq 0} \sum_{m \in I_k} (X_{mn} - U_{mk} t)^2 + \mu V t^2 + \lambda V |t| =: \min_{t \geq 0} \phi_{kn}(t)$$

based on the objective function in (15). To find a minimizer of the above problem, we examine the zero gradient condition

$$0 \in \partial \phi_{kn}(t) = \sum_{m \in I_k} 2(U_{mk} t - X_{mn} U_{mk}) + 2 \mu V t + \lambda V \partial |t|(t),$$

where $\partial |t|$ is the subdifferential of the absolute value function. This can be equivalently written as

$$\sum_{m \in I_k} \frac{X_{mn} U_{mk}}{\|u_k\|_2^2 + \mu V} \in t + \frac{\lambda V}{2(\|u_k\|_2^2 + \mu V)} \partial |t|(t) =: (\text{id} + \gamma \partial |t|)(t)$$

with $\gamma := \lambda V / (\|u_k\|_2^2 + 2 \mu V)$. By using the soft thresholding function $\tau_{\gamma} := (\text{id} + \gamma \partial |t|)^{-1}$, we finally obtain

$$V_{kn} = \arg \min_{t \geq 0} \phi_{kn}(t) = \tau_{\gamma} \left( \sum_{m \in I_k} \frac{X_{mn} U_{mk}}{\|u_k\|_2^2 + \mu V} \right) = \begin{cases} \frac{\sum_{m \in I_k} X_{mn} U_{mk}}{\|u_k\|_2^2 + \mu V} - \gamma, & \gamma \leq \frac{\sum_{m \in I_k} X_{mn} U_{mk}}{\|u_k\|_2^2 + \mu V}, \\ 0, & \text{else}, \end{cases}$$

(21)

where the case

$$\frac{\sum_{m \in I_k} X_{mn} U_{mk}}{\|u_k\|_2^2 + \mu V} \leq -\gamma$$
can be omitted due to the nonnegativity constraint.

For the minimization with respect to $U$, we consider the minimization task
\begin{equation}
\min_{t \geq 0} \sum_{n=1}^{N} (X_{m,n} - V_{\pi(m),n}t)^2 + \mu U t^2 + \lambda U \|t\| =: \min_{t \geq 0} \psi_{m,\pi(m)}(t)
\end{equation}
based on the objective function in (17). Similar as in the case before, the solution can be obtained via the zero gradient condition
\[
0 \in \sum_{n=1}^{N} 2(V_{\pi(m),n}t - X_{m,n})V_{\pi(m),n} + 2\mu U t + \lambda U \partial \|t\| = 0,
\]
which can be equivalently written as
\[
\frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \in t + \frac{\lambda U}{2(\|\psi_{\pi(m)}\|^2_2 + \mu U)} \partial \|t\| =: (\text{id} + \gamma \partial \|t\|)(t)
\]
with $\gamma := \lambda U/(2\|\psi_{\pi(m)}\|^2_2 + 2\mu U)$. As before, by using the soft thresholding function $\tau_\gamma$, we finally obtain
\begin{equation}
U_{m,\pi(m)} = \arg \min_{t \geq 0} \psi_{m,\pi(m)}(t) = \tau_\gamma \left( \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \right) = \begin{cases} \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} - \gamma, & \gamma \leq \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \\ 0 & \text{else}. \end{cases}
\end{equation}

Based on (18) and by inserting (23) into $\psi_{m,\pi(m)}(t)$, we get the distance measure
\begin{equation}
\text{dist}(x_m, \psi_{\pi(m)})^2 = \|x_m - \tau_\gamma \left( \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \right) \psi_{\pi(m)}\|_2^2 + \mu U \left( \tau_\gamma \left( \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \right) \right)^2 + \lambda U \tau_\gamma \left( \frac{\langle x_m, \psi_{\pi(m)} \rangle^2}{\|\psi_{\pi(m)}\|^2_2 + \mu U} \right)
\end{equation}
between a data point $x_m$ and a centroid $\psi_{\pi(m)}$.

### 3.3 $\ell_1$ Discrepancy Term

Different from the classical Frobenius norm as a discrepancy term, the $\ell_1$ discrepancy term leads to a more robust NMF formulation with respect to noise and outliers [13].

Regarding the minimization with respect to $V$, we consider the scalar minimization problem
\begin{equation}
\min_{t \geq 0} \sum_{m \in I_k} |X_{mn} - U_{mk}t| + \mu V t^2 + \lambda V \|t\| =: \min_{t \geq 0} \varphi_{kn}(t)
\end{equation}
with the objective function given in (15). The latter has the same structure as in (19) and can be solved by taking the weighted and regularized median defined in (20). Hence, we have that
\begin{equation}
V_{kn} = \text{med}_{\psi_{\pi(m)}}(x_{\pi(m),k}) \in \arg \min_{t \geq 0} \varphi_{kn}(t),
\end{equation}
where $U_{*,k}, X_{*,n} \in \mathbb{R}_{\geq 0}^{[I_k]}$ are the corresponding reduced vectors with the entries $U_{m,k} \in \{U_{mk} \mid m \in I_k\}$ and $X_{mn} \in [X_{mn} \mid m \in I_k]$ for $n = 1, \ldots, [I_k]$.

Similarly, for the minimization with respect to $U$, we consider the minimization task
\begin{equation}
\min_{t \geq 0} \sum_{m \in I_k} |X_{mn} - V_{\pi(m),n}t| + \mu U t^2 + \lambda U \|t\| =: \min_{t \geq 0} \psi_{m,\pi(m)}(t)
\end{equation}
based on the objective function in (17) to obtain the non-zero entry $U_{m,\pi(m)}$ in the $m$-th row of $U$. This also has the same structure as in (19) and thus yields
\begin{equation}
U_{m,\pi(m)} = \text{med}_{\psi_{\pi(m)}}(x_{\pi(m),m}) \in \arg \min_{t \geq 0} \psi_{m,\pi(m)}(t),
\end{equation}
which can be seen as the weighted median of the entries in $x_m$ with elastic-net regularization. Finally, based on (18) and by inserting (26) into the function $\psi_{m,\pi(m)}$, we obtain the distance measure
\begin{equation}
\text{dist}(x_m, \psi_{\pi(m)}) = \|x_m - \text{med}_{\psi_{\pi(m)}}(x_{\pi(m)}) \psi_{\pi(m)}\|_1 + \mu U \left( \text{med}_{\psi_{\pi(m)}}(x_{\pi(m)}) \psi_{\pi(m)} \right)^2 + \lambda U \left( \text{med}_{\psi_{\pi(m)}}(x_{\pi(m)}) \psi_{\pi(m)} \right)
\end{equation}
between a data point $x_m$ and a centroid $\psi_{\pi(m)}$. A more intuitive understanding and the consideration of special cases of the generalized distance measures in (27) and (24) can be found in the following Section 3.4.
3.4 Special Cases

In this section, we give a more intuitive understanding of the obtained distance measures and centroids in the previous sections and discuss some special cases.

First, we note that the obtained distance measures in (27) and (24) do not satisfy the usual properties of a metric in a metric space as in Definition 2.1. However, the distance measure \( \text{dist}(x_m, v_{\pi(m)}) \) for both the \( \ell_1 \) and \( \ell_2 \) projection of the data point \( x_m \) onto \( v_{\pi(m)} \) respectively.

In the case of the \( \ell_2 \) discrepancy term, we obtain due to (23) two different distance measures in the case of sparsity regularization with \( \lambda_U > 0 \). In the case of \( \gamma > \langle x_m, v_{\pi(m)} \rangle / \|v_{\pi(m)}\|_2^2 + \mu_U \), which means \( \lambda_U/2 > \langle x_m, v_{\pi(m)} \rangle/2 \), this leads to approximately orthogonal \( x_m \) and \( v_{\pi(m)} \) if \( \lambda_U \) is small and to \( \tau_n(\langle x_m, v_{\pi(m)} \rangle/2/\|v_{\pi(m)}\|_2^2 + \mu_U) = 0 \), so that we obtain the distance measure \( \text{dist}(x_m, v_{\pi(m)})^2 = \|x_m\|_2^2 \). For the other case \( \lambda_U/2 \leq \langle x_m, v_{\pi(m)} \rangle/2 \), we obtain the more complex distance measure

\[
\text{dist}(x_m, v_{\pi(m)})^2 = \|x_m - \langle x_m, v_{\pi(m)} \rangle/\|v_{\pi(m)}\|_2^2 - \gamma \rangle v_{\pi(m)} \|_2^2 + \mu_U \left( \langle x_m, v_{\pi(m)} \rangle^2/\|v_{\pi(m)}\|_2^2 + \mu_U - \gamma \right) + \lambda_U \left( \langle x_m, v_{\pi(m)} \rangle^2/\|v_{\pi(m)}\|_2^2 + \mu_U - \gamma \right) \geq 0
\]

(28)

Note that in both cases, the distance measure does not have the usual properties of a metric in a metric space as \( \text{dist}(x, x) \neq 0 \) for some vector \( x \in \mathbb{R}^N \). Besides the classical Tikhonov regularization with the corresponding regularization parameters \( \mu_U \) and \( \mu_V \), it is interesting to examine the connection between the obtained distance measure and the sparsity regularization with \( \lambda_U > 0 \). In the case of a high sparsity regularization along with the case \( \lambda_U/2 > \langle x_m, v_{\pi(m)} \rangle/2 \), we obtain a distance measure \( \text{dist}(x_m, v_{\pi(m)}) \) which is independent from the centroid \( v_{\pi(m)} \) so that in these cases, the corresponding data point can be assigned arbitrarily to any centroid. This can be also inferred based on the sparsity regularization on \( U \). Due to the nonnegativity of \( U \) together with the orthogonality constraint in (C1), the matrix is already sparse and only contains at most one non-zero entry \( U_{mk} \), in each row indicating the association of the data point \( x_m \) to the cluster \( I_k \). However, if \( \lambda_U \) is set sufficiently large, the additional sparsity regularization can lead to rows \( U_{mk} \) without any non-zero entry and yields the same interpretation, namely that the corresponding data point can be assigned to an arbitrary cluster.

Before examining unregularized versions of the ONMF models, it is interesting to consider the distance measure in (28) with \( \mu_U = 0 \). It can be easily shown that this leads to

\[
\text{dist}(x_m, v_{\pi(m)})^2 = \|x_m\|_2^2 \sin^2 \angle(x_m, v_{\pi(m)}) + \lambda_U \|v_{\pi(m)}\|_2^2 \left( \langle x_m, v_{\pi(m)} \rangle^2 - \lambda_U/4 \right),
\]

which explicitly shows the angle dependence of the distance measure. Here, we use the usual notation

\[
\cos \angle(x, y) = \frac{x \cdot y}{\|x\|_2 \|y\|_2}, \quad \sin \angle(x, y) = \sqrt{1 - \cos^2 \angle(x, y)},
\]

where \( \angle(x, y) \) denotes the angle between two vectors \( x, y \in \mathbb{R}^N \).

Furthermore, we note that for both the \( \ell_1 \) and \( \ell_2 \) discrepancy terms, the centroids given in (25) and (21) cannot be computed directly from the data points \( \{x_m | m = 1, \ldots, M\} \) and are dependent from the cluster membership matrix.

Moreover, it is interesting to examine shortly some unregularized special cases and compare them with some results throughout the literature. Pompili et al. in [21] consider the \( \ell_2 \) discrepancy term with \( \lambda_U = \lambda_V = \mu_U = \mu_V = 0 \) and further restricts \( U \) to satisfy \( U^T U = I_{K \times K} \) by including the additional normalization constraint \( (u_k, v_k) = 1 \) for all \( k \in \{1, \ldots, K\} \). It can be shown, that the normalization constraint can be equivalently imposed on the rows of \( V \) [21], which then leads to the distance measure \( \text{dist}(x_m, v_{\pi(m)}) = \|x_m - \langle x_m, v_{\pi(m)} \rangle/\|v_{\pi(m)}\|_2 \|v_{\pi(m)}\|_2 \) based on (24). This coincides with the distance measure given in the proof of [21], where the equivalence to a weighted variant of the spherical \( K \)-means method is shown. For the centroids, we obtain \( v_k = \frac{1}{\|u_k\|_2^2} \sum_{m \in I_k} U_{mk} x_m \), based on (21).

If we further restrict \( U \) to be

\[
U_{mk} = \begin{cases} 0, & m \notin I_k, \\ \frac{1}{\sqrt{|I_k|}}, & m \in I_k, \end{cases}
\]

9
which also leads to \( \mathbf{U}^\top \mathbf{U} = \mathbf{I}_{K \times K} \), we obtain the distance measure
\[
\text{dist}(\mathbf{x}_m, \mathbf{v}_{\pi(m)})^2 = \|\mathbf{x}_m - \frac{1}{\sqrt{|I_{\pi(m)}|}} \mathbf{v}_{\pi(m)}\|^2.
\] (29)

by considering the minimization task (22) and inserting the non-zero solution \( 1/\sqrt{|I_{\pi(m)}|} \). Furthermore, based on (21), the centroids are given by
\[
\mathbf{v}_k = \frac{1}{\sqrt{|I_k|}} \sum_{m \in I_k} \mathbf{x}_m.
\] (30)

Both the distance measure and the centroid given in (29) and (30) coincides with the ones stated in [21]. If we instead constrain \( \mathbf{U} \) to be
\[
U_{mk} = \begin{cases} 
0, & m \notin I_k, \\
1, & m \in I_k, 
\end{cases}
\] (31)

we obtain based on the above procedure the usual Euclidean distance and the mean of the data points given by
\[
\text{dist}(\mathbf{x}_m, \mathbf{v}_{\pi(m)}) = \|\mathbf{x}_m - \mathbf{v}_{\pi(m)}\|_2,
\]
\[
\mathbf{v}_k = \frac{1}{|I_k|} \sum_{m \in I_k} \mathbf{x}_m,
\]

used in the classical \( K \)-means approach. Finally, for the \( \ell_1 \) discrepancy term and the constraints in (31), we obtain
\[
\text{dist}(\mathbf{x}_m, \mathbf{v}_{\pi(m)}) = \|\mathbf{x}_m - \mathbf{v}_{\pi(m)}\|_1, \\
V_{kn} = \text{med}(\tilde{\mathbf{X}}_{\bullet,n}),
\]

based on (27) and (25), which yields the usual \( K \)-median approach with the \( \ell_1 \) distance and the median of the data points. Note that due to the application of the median, the centroids are chosen among the available data points \( \{\mathbf{x}_m \mid m = 1, \ldots, M\} \). Furthermore, the application of the \( \ell_1 \) distance function leads to a more stable clustering approach with respect to outliers.

4 Conclusion

In this work, we presented a novel derivation framework to obtain connections between generalized \( K \)-means clustering approaches and regularized ONMF models by directly extracting the distance measures and centroids of the \( K \)-means method based on the considered ONMF problem. We applied this technique to non-standard ONMF models with elastic net regularization on both factorization matrices and derived the corresponding distance measures and centroids of the generalized \( K \)-means model. Furthermore, we gave an intuitive view on the obtained results and shortly described the effect of some regularization terms in the ONMF models on the distance function and the clustering outcome. Finally, we analyzed several special cases and found that the obtained results based on the proposed framework coincide with the results in the existing literature.

Several further research directions could be of interest. A good starting point could be a further theoretical as well as numerical evaluation of the obtained generalized distance measures and centroids in Section 3. Moreover, a possible extension of this work could be to study whether the proposed derivation framework can be applied to more general discrepancy and regularization terms. Especially the analysis of gradient based regularization terms, which enforce spatial coherence in the clusterings, and their effects on the obtained distance measures and centroids of the generalized \( K \)-means approach could be of particular interest, since this can lead to an improved clustering performance for specific applications [7].

A further theoretical direction constitutes the extension of the whole framework to infinite dimension spaces leading to continuous factorization problems. In combination with gradient based regularization terms, the analysis of first order conditions could lead to connections to \( K \)-means methods in a continuous setting as well as partial differential equations, whose solutions could lead to the desired centroids and distance measures.

Acknowledgments

This project was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) within the framework of RTG “\( \pi^3 \): Parameter Identification – Analysis, Algorithms, Applications” – Project number 281474342/GRK2224/1.
Regularized ONMF and K-means Clustering

References


JOINT RECONSTRUCTION AND LOW-RANK
DECOMPOSITION FOR DYNAMIC INVERSE PROBLEMS

SIMON ARRIDGE
Department of Computer Science
University College London
London WC1E 6BT, United Kingdom

PASCAL FERNSEL*
Center for Industrial Mathematics
University of Bremen
28359 Bremen, Germany

ANDREAS HAUPTMANN
Department of Mathematical Sciences
University of Oulu
90014 Oulu, Finland
&
Department of Computer Science
University College London
London WC1E 6BT, United Kingdom

(Communicated by Xiaoqun Zhang)

Abstract. A primary interest in dynamic inverse problems is to identify the underlying temporal behaviour of the system from outside measurements. In this work, we consider the case, where the target can be represented by a decomposition of spatial and temporal basis functions and hence can be efficiently represented by a low-rank decomposition. We then propose a joint reconstruction and low-rank decomposition method based on the Nonnegative Matrix Factorisation to obtain the unknown from highly undersampled dynamic measurement data. The proposed framework allows for flexible incorporation of separate regularisers for spatial and temporal features. For the special case of a stationary operator, we can effectively use the decomposition to reduce the computational complexity and obtain a substantial speed-up. The proposed methods are evaluated for three simulated phantoms and we compare the obtained results to a separate low-rank reconstruction and subsequent decomposition approach based on the widely used principal component analysis.

1. Introduction. Several inverse problems are concerned with the reconstruction of solutions in multiple physical dimensions such as space, time and frequency. Generally, such problems require very large datasets in order to satisfy conditions for accurate reconstruction, whereas in practice only subsets of such complete data can be measured. Furthermore, the information content of the solutions from such reduced data may be much less than suggested by the complete set. In these cases, regularisation in the reconstruction process is required to compensate for the

2020 Mathematics Subject Classification. Primary: 15A23, 65K10, 65F22; Secondary: 15A69.
Key words and phrases. Nonnegative matrix factorisation, dynamic inverse problems, low-rank decomposition, variational methods, dynamic computed tomography.

*Corresponding author: Pascal Fernsel.
reduced information content, for instance by correlating features between auxiliary physical dimensions.

For instance, dynamic inverse problems have gained considerable interest in recent years. This development is partly driven by the increase in computational resources and the possibility to handle large data sizes more efficiently, but also novel and more efficient imaging devices enabling wide areas of applications in medicine and industrial imaging. In medical imaging, dynamic information is essential for accurate diagnosis of heart diseases or for applications in angiography to determine blood flow by injecting a contrast agent to the patient’s blood stream. But also in nondestructive testing and chemical engineering, tomographic imaging has become increasingly popular to monitor dynamic processes. The underlying problem in these imaging scenarios is often that a fine temporal sampling, i.e., a large number of channels in the discrete setting, is only possible under considerable restrictions to sampling density at each time instance. This limitation often renders time-discrete (static) reconstructions insufficient. Additionally, an underlying problem in many dynamic applications is given by the specific temporal dynamics of the process, which are often non-periodic and hence prevents temporal binning approaches. Thus, it is essential to include the dynamic nature of the imaging task in the reconstruction process.

With increasing computational resources, it has become more feasible to address the reconstruction task as a fully dynamic problem in a spatio-temporal setting. In these approaches, it is essential to include the dynamic information in some form into the reconstruction task [27]. This could be done for instance by including a regularisation on the temporal behaviour as penalty in a variational setting [44, 45]. Such approaches have been used in a wide variety of applications, such as magnetic resonance imaging [18, 39, 46], X-ray tomography [5, 40] and applications to process monitoring with electrical resistance tomography [10]. More advanced approaches aim to include a physical motion model and estimate the motion of the target from the measurements itself. This can be done for instance by incorporating an image registration step into the reconstruction algorithm and reformulate the reconstruction problem as a joint motion-estimation and reconstruction task [6, 7, 16, 38]. Another possibility is the incorporation of an explicit motion model by metamorphosis as considered in [11, 25].

In this work, we consider another possibility to incorporate regularisation, and in particular temporal regularity, to the reconstruction task by assuming a low-dimensional representation of the unknown. This leads naturally to a low-rank description of the underlying inverse problem and is especially suitable to reduce data size in cases where we have much fewer basis functions to represent the unknown than the temporal sampling. In a continuous setting, this yields the analysis of low-rank approximations in tensor product of Hilbert spaces, for which we refer the reader to [30, 50]. We rather focus on low-rank approximation methods in a discretised framework, which leads to the use of specific matrix factorisation approaches and their optimisation techniques.

In particular, in this work we propose a joint reconstruction and decomposition in a variational framework using non-negative matrix factorisation, which naturally represents the physical assumption of nonnegativity of the dynamic target and allows for a variety of regularising terms on spatial and temporal basis functions. Following this framework, we propose two algorithms, that either jointly recover the reconstruction and the low-rank decomposition, or alternatively recovers only
the low-rank representation of the unknown without the need to construct the full spatio-temporal target in the reconstruction process. Here, the second approach effectively incorporates the dimension reduction and can lead under coherent angular sampling in time to a significant reduction in computational complexity. This can be particularly useful, if one is only interested in the dynamics of the system and not the full reconstruction. For incoherent temporal sampling the method effectively aggregates the angular information between time-steps similar to a Bayesian filtering approach [26]. Finally, the main theoretical result of our paper shows that a flexible variational framework can be formulated and we prove that the derived algorithms lead to a monotonic decrease of the respective cost functions.

Reconstruction algorithms for dynamic inverse problems based on low-rank approximations have been proposed earlier. For instance, by a combination of low-rank and sparsity constraints [21, 52, 49, 48] based on the idea to decompose the given data matrix into a sum of two matrices. Here, one matrix has low-rank and models the stationary background over time and the other matrix is sparse to represent the temporal variation in space, which is expected to be sparse. Further low-rank approximation models are based on a trained principal component analysis used for cine cone-beam CT [22] or patch based low-rank regularisation terms [28]. Similar to our considered approaches is the work of [8], where a low-rank reconstruction method is used for the application to cine cone-beam CT, based on a matrix factorisation model and the assumption that only a few principle components are sufficient to reconstruct a given body motion of the patient.

This paper is organised as follows. In Section 2 we discuss our setting for dynamic inverse problems and continue to discuss low-rank decomposition approaches. Specifically, Principal Component Analysis (PCA) and Nonnegative Matrix Factorisation (NMF), which is the focus in this study. As a baseline, we first present a low-rank reconstruction method followed by one of the considered decomposition methods. We then continue to present the proposed framework of joint reconstruction and decomposition with the NMF. In particular, we prove that the proposed framework leads to a monotonic decrease of the cost functions. We then proceed in Section 3 to evaluate the algorithms under considerations with the use case of dynamic X-ray tomography and three simulated phantoms with different characteristics. We conclude the study in Section 4 with some thoughts on the extension of the proposed framework.

2. Reconstruction and low-rank decomposition methods.

2.1. A setting for dynamic inverse problems. In this work, we consider a general multi-dimensional inverse problem, where the unknown \( x(s, t) \) is defined on a spatial domain \( \Omega_1 \subset \mathbb{R}^d_1 \) with dependence on a secondary variable \( t \in \mathbb{R}_{\geq 0} \) defined in a bounded interval \( T := [0, T] \). This setting admits some quite general applications where the secondary variable could have other physical interpretations, notably wavelength for hyper-spectral problems; however, to fix our ideas, we henceforth consider \( t \) to explicitly represent time, and our application to be that of dynamic inverse problems. Consequently, the underlying equation of the resulting inverse problem can be described in the following form

\[
A(x(s, t); t) = y(\sigma, t) \quad \text{for } t \in T,
\]

where \( A \) is a time-dependent linear bounded operator between suitable Hilbert spaces for the functions \( x : \Omega_1 \times T \to \mathbb{R}_{\geq 0} \) and the measurement data \( y : \Omega_2 \times T \to \mathbb{R}_{\geq 0} \).
$\mathbb{R}_{\geq 0}$ with domain $\Omega_2 \subset \mathbb{R}^{d_2}$. We will primarily consider the non-stationary case here, where the forward operator $A$ is dependent on $t$.

In the special case of a stationary operator $A(\cdot; t) \equiv A$ for all $t \in T$, where for each $t$ the operator follows the same sampling process, we can achieve possible computational improvements. The resulting implications will be discussed later in Section 2.5.

Furthermore, the underlying assumption in this work is that the unknown $x$ can be approximated by a set of spatial $b^k : \Omega_1 \to \mathbb{R}_{\geq 0}$ and channel basis functions $c^k(t) : T \to \mathbb{R}_{\geq 0}$ for $1 \leq k \leq K$. Then, the unknown can be approximated by the decomposition

$$x(s, t) \approx \sum_{k=1}^{K} b^k(s) c^k(t).$$

This formulation naturally gives rise to the reconstruction and low-rank decomposition framework to extract the relevant features given by $b^k$ and $c^k$. An illustration for a possible phantom represented by (2) is shown in Figure 1.

We intentionally keep the formulation general here to allow for applications different to dynamic inverse problems, such as multi-spectral imaging. Nevertheless, the derived reconstruction and feature extraction framework in this paper will be used in Section 3 for the specific application to dynamic computed tomography.

![Figure 1](image-url)

**Figure 1.** Illustration of a phantom that can be represented by the decomposition in (2). The phantom consists of $K = 3$ components: the background and two dynamic components with periodically changing intensity (left and right plot). As such, this phantom can be efficiently represented by a low-rank decomposition considered in this study.

Furthermore, a suitable discretisation of the continuous formulation (1) is needed to introduce the feature extraction methods in the forthcoming sections. Let us first discretise the secondary variable, such that $t \in \mathbb{N}$ with $1 \leq t \leq T$. For the spatial domain, we assume a vectorised representation such that the resulting unknown can be represented as a matrix $X \in \mathbb{R}^{N \times T}$, which leads to the matrix equation

$$A_t X_{\bullet, t} = Y_{\bullet, t} \quad \text{for} \quad 1 \leq t \leq T,$$

where $A_t \in \mathbb{R}^{M \times N}$ is the discretised forward operator, $X_{\bullet, t}$ the $t$-th column of $X$ and $Y_{\bullet, t}$ the $t$-th column of the data matrix $Y \in \mathbb{R}^{M \times T}$. Analogously, we will write $M_{n, \bullet}$ for the $n$-th row of an arbitrary matrix $M$.

Suitable restrictions to the matrices in Equation (3) will be made in the following sections to ensure the applicability of the considered frameworks and, if possible, to properly represent the decomposition (2).
2.2. Feature extraction methods. In this section, we introduce two feature extraction methods, namely the Principal Component Analysis (PCA) and the Nonnegative Matrix Factorisation (NMF). These approaches are used to compute the latent components of the reconstruction $X$. The NMF will be used in Section 2.4 to introduce a joint reconstruction and low-rank decomposition framework to tackle the problem stated in (3).

2.2.1. Principal component analysis. Large and high dimensional datasets demand modern data analysis approaches to reduce the dimensionality and increase the interpretability of the data while keeping the loss of information as low as possible. Many different techniques have been developed for this purpose, but PCA is one of the most widely used and goes back to [43].

For a given matrix $X \in \mathbb{R}^{N \times T}$ with $N$ different observations of an experiment and $T$ features, the PCA is a linear orthogonal transformation given by the weights $\hat{C}_{k,\bullet} = (\hat{C}_{k1}, \ldots, \hat{C}_{kT})$ with $\hat{C} \in \mathbb{R}^{\hat{K} \times T}$, which transforms each observation $X_{n,\bullet}$ to principal component scores given by $B_{n,\hat{k}} = \sum_{t} X_{nt} \hat{C}_{kt}$ with $B_{\bullet,\hat{k}} \in \mathbb{R}^{N \times \hat{K}}$ and $\hat{K} = \min(N - 1, T)$, such that

- the sample variance $\text{Var}(B_{\bullet,\hat{k}})$ is maximised for all $\hat{k}$,
- each row $\hat{C}_{k,\bullet}$ is constrained to be a unit vector
- and the sample covariance $\text{cov}(B_{\bullet,\hat{k}}, B_{\bullet,\hat{k}}') = 0$ for $\hat{k} \neq \hat{k}$. 

Together with the usual assumption that the number of observations is higher than the underlying dimension, this leads to $\hat{K} = T$ and the full transformation $B = X \hat{C}^\top$, where $\hat{C}$ is an orthogonal matrix. The $t$-th column vector $(C_{t,\bullet})^\top$ defines the $t$-th principal direction and is an eigenvector of the covariance matrix $S = X^\top X/(N - 1)$. The corresponding $t$-th largest eigenvalue of $S$ denotes the variance of the $t$-th principal component.

The above transformation is equivalent to the factorisation of the matrix $X$ given by

$$X = BC,$$

which allows to decompose each observation into the principal components, such that $X_{n,\bullet} = \sum_{t=1}^{T} B_{n,t} C_{t,\bullet}$. Hence, it follows that $X = \sum_{t=1}^{T} B_{\bullet,t} C_{t,\bullet}$.

Furthermore, it is possible to obtain an approximation of the matrix $X$ by truncating the sum at the first $K < T$ principle components for all $n$, which yields a rank $K$ matrix $X^{(K)}$ given by

$$X^{(K)} = \sum_{k=1}^{K} B_{\bullet,k} C_{k,\bullet}.$$ 

Based on the Eckart-Young-Mirsky theorem [24], $X^{(K)}$ is the best rank $K$ approximation of $X$ in the sense that it minimises the discrepancy $\|X - X^{(K)}\|$ for both the Frobenius and spectral norm.

One typical approach to compute the PCA is based on the Singular Value Decomposition (SVD) of the data matrix $X = U \Sigma V^\top$ and will be used in this work. Setting $B := U \Sigma$ and $C := V^\top$ gives already the desired factorisation in (4) based on the PCA.

2.2.2. Nonnegative matrix factorisation. Nonnegative Matrix Factorisation (NMF), originally introduced as positive matrix factorisation by Paatero and Tapper in
1994 [42], is an established tool to obtain low-rank approximations of nonnegative data matrices. It has been widely used in the machine learning and data mining community for compression, basis learning, clustering and feature extraction for high-dimensional classification problems with applications in music analysis [20], document clustering [15] and medical imaging problems such as tumor typing in Matrix-Assisted Laser Desorption/Ionisation (MALDI) imaging in the field of bioinformatics [36].

Different from the PCA approach above, the NMF enforces nonnegativity constraints on the factor matrices without any orthogonality restrictions. This makes the NMF the method of choice for application fields, where the underlying physical model enforces the solution to be nonnegative assuming that each datapoint can be described as a superposition of some unknown characteristic features of the dataset. The NMF makes it possible to extract these features while constraining the matrix factors to have nonnegative entries, which simplifies their interpretation. These data assumptions are true for many application fields including the ones mentioned above but also especially for our considered problem of dynamic computed tomography, where the measurements consist naturally of the nonnegative absorption of photons. Mathematically, the basic NMF problem can be formulated as follows: For a given nonnegative matrix $X \in \mathbb{R}_{\geq 0}^{N \times T}$, find nonnegative matrices $B \in \mathbb{R}_{\geq 0}^{N \times K}$ and $C \in \mathbb{R}_{\geq 0}^{K \times T}$ with $K \ll \min\{N,T\}$ such that

$$X \approx BC.$$  

The factorisation allows to approximate the rows $X_{n,:}$ and columns $X_{:,t}$ as a superposition of the $K$ columns $B_{:,k}$ of $B$ and rows $C_{k,:}$ of $C$ respectively, such that $X_{n,:} \approx \sum_{k=1}^{K} B_{nk}C_{k,:}$ and $X_{:,t} \approx \sum_{k=1}^{K} C_{kt}B_{:,k}$. Similarly, it holds that

$$X \approx BC = \sum_{k=1}^{K} B_{:,k}C_{k,:},$$

where the $K$ terms of the sum are rank-one matrices. Hence, the sets $\{B_{:,k}\}_k$ and $\{C_{k,:}\}_k$ can be interpreted as a low-dimensional basis to approximate $X$, i.e. the NMF performs the task of basis learning with additional nonnegativity constraints.

The usual approach to compute the factorisation is to define a suitable discrepancy term $D_{\text{NMF}}$, which has to be chosen according to the noise assumption of the underlying problem, and to reformulate the NMF as a minimisation problem. Typical discrepancies include the default case of the Frobenius Norm on which we will focus on, the Kullback-Leibler divergence or other generalised divergences [12].

Furthermore, NMF problems are usually ill-posed due to the non-uniqueness of the solution [29] and require the application of suitable regularisation techniques. One common method is to include penalty terms in the minimisation problem to tackle the ill-posedness of the problem but also to enforce desirable properties of the factorisation matrices. Typical examples range from $\ell_1, \ell_2$ and total variation regularisation terms [33] to more problem specific terms, which enforce additional orthogonality of the matrices or even allow supervised classification workflows if the NMF is used as a prior feature extraction method [19, 36].

Hence, the general regularised NMF problem can be written as

$$\min_{B,C \geq 0} D_{\text{NMF}}(X, BC) + \sum_{\ell=1}^{L} \gamma_{\ell} P_{\ell}(B,C) =: \min_{B,C \geq 0} F(B,C),$$

with $D_{\text{NMF}}(X, BC) = \|X - BC\|_F^2$. The penalty terms $P_{\ell}$ can be chosen according to the specific application and the underlying noise model.
where $\mathcal{P}_\ell$ denote the penalty terms, $\gamma \geq 0$ the corresponding regularisation parameters and $\mathcal{F}$ the cost function of the NMF.

Another crucial step in the whole workflow of the feature extraction via both the NMF and the PCA is the determination of an optimal number $K$ of components. Typical methods are based on approximative or heuristic techniques and include residual analysis, core consistency diagnostics based on a principal component analysis or the investigation of the rank-one matrices $B_{\bullet,k}C_{\bullet,k}$ (see [12] and the references therein). Following the ideas of [8], this work focuses on the latter approach, which will be further specified in Section 3.1 and Appendix C.1.

The considered optimisation approach in this work is based on the so-called Majorise-Minimisation (MM) principle and gives rise to multiplicative update rules of the matrices in (5), which automatically preserve the nonnegativity of the iterates provided that they are initialised nonnegative. For more details on this optimisation technique, we refer the reader to Appendix A. The idea of the feature extraction procedure based on the NMF can be well illustrated by considering the example from Figure 1 that satisfies the decomposition assumption from (2). Here, the highlighted spatial regions change their intensities according to the given dynamics. The NMF allows a natural interpretation of the factorisation matrices $B$ and $C$ as the spatial and temporal basis functions for this case, as illustrated in Figure 2. The column $X_{\bullet,t}$ of $X$ denotes the reconstruction of the $t$-th time step of the inverse problem in (3). The NMF allows to decompose the spatial and temporal features of $X$: The matrix $B$ contains the spatial features in its columns with the corresponding temporal features in the rows of $C$.

![Figure 2. Structure of the NMF in the context of the dynamic Shepp-Logan phantom as shown in Figure 1. Here, the nonnegative spatial and temporal basis functions can be naturally represented by the matrices $B$ and $C$.](image)

### 2.3. Separated reconstruction and low-rank decomposition.

Let us first discuss a separated reconstruction and feature extraction approach to solve the inverse problem in (3), which means that first a reconstruction is computed and afterwards the feature extraction is performed subsequently with one of the previously discussed methods. We consider this method as baseline for our comparison.
The considered reconstruction method for this separated framework involves a basic gradient descent approach together with a regularisation step and a subsequent total variation denoising, which will henceforth referred to as \texttt{gradTV}. The details on the algorithm are provided in Algorithm 1. In particular, we aim to compute solutions to the least squares problem and incorporate the low-rank assumptions as additional penalty of the nuclear norm of $X_{*,t}$, that is

$$\min_{X_{*,t} \geq 0} \|Y_{*,t} - A_t X_{*,t}\|_2^2 + \alpha \|X_{*,t}\|_*$$

for all $t$; see e.g. [37, 21, 49, 52]. This can then be efficiently solved by a proximal gradient descent scheme with a soft-thresholding on the singular values enforcing the low-rank structure. Ideally, one would like to include the total variation regularisation as penalty term, but as this tends to be computationally expensive for the fine temporal sampling, we include this as a subsequent denoiser.

In practice, after a suitable initialisation of the reconstruction matrix, the gradient descent step is computed with an, a priori defined, fixed stepsize $\rho_{\text{grad}}$. For the proximal step, the truncated SVD of $X$ is computed and a soft thresholding of the singular values is performed with a fixed threshold $\rho_{\text{thr}}$. Afterwards, we enforce the nonnegativity with a projection step on the reconstruction $X$. When the stopping criterion is satisfied, a TV denoising algorithm\footnote{https://www.mathworks.com/matlabcentral/fileexchange/36278-split-bregman-method-for-total-variation-denoising} based on [23, 48] with the corresponding parameter $\rho_{\text{TV}}$ is applied.

\textbf{Algorithm 1 gradTV}

1: \textbf{Initialise:} $X$

2: \textbf{Input:} $\rho_{\text{grad}}, \rho_{\text{thr}}, \rho_{\text{TV}} > 0$

3: \textbf{repeat}

4: \hspace{1em} $X_{*,t} \leftarrow X_{*,t} - \rho_{\text{grad}} (A_t^* A_t X_{*,t} - A_t^* Y_{*,t})$ \hspace{1em} for all $t$

5: \hspace{1em} $(U, \Sigma, V) \leftarrow \text{SVD}(X)$

6: \hspace{1em} $\Sigma \leftarrow \text{SoftThresh}_{\rho_{\text{thr}}} (\Sigma)$

7: \hspace{1em} $X \leftarrow U \Sigma V^\top$

8: \hspace{1em} $X \leftarrow \max(X, 0)$

9: \textbf{until} \text{StoppingCriterion} satisfied

10: $X \leftarrow \text{TVDenoiser}_{\rho_{\text{TV}}}(X)$

11: \textbf{return} $X$

After the reconstruction procedure given by Algorithm 1, we perform the feature extraction of the reconstruction $X$ via both the PCA and the NMF and call the approach \texttt{gradTV$_{PCA}$} and \texttt{gradTV$_{NMF}$} respectively.

For \texttt{gradTV$_{PCA}$}, we simply compute the PCA of $X$ based on its SVD. Concerning the method \texttt{gradTV$_{NMF}$}, we consider the standard NMF model

$$\min_{B,C \geq 0} \|X - BC\|_F^2 + \frac{\tilde{\mu}_C}{2} \|C\|_F^2$$

with the parameter $\tilde{\mu}_C$. The $\ell_2$ regularisation penalty term on $C$ is motivated by our application in Section 3. The corresponding multiplicative algorithms to solve (6) are well-known [13, 19] and a special case of the derived update rules in the next section.
2.4. **Joint reconstruction and low-rank decomposition.** Instead of the previously discussed separated reconstruction, we now aim to include the feature extraction into the reconstruction procedure. This gives rise to consider a joint reconstruction and low-rank decomposition approach based on the NMF, rather than one based on a low-rank plus sparsity approach based on PCA [9, 47, 53]. The basic idea of the method is to incorporate the reconstruction procedure of the inverse problem in (3) into the NMF workflow. To do this, we have to additionally assume that 

\[ A_t \in \mathbb{R}^{M \times N} \geq 0, \quad Y \in \mathbb{R}^{M \times T} \geq 0 \quad \text{and} \quad X \in \mathbb{R}^{N \times T} \geq 0 \]

to ensure the desired nonnegativity of the factorisation matrices \( B \) and \( C \), which corresponds to the assumptions of the decomposition in (2). The main motivation is that this joint approach allows the reconstruction process to exploit the underlying latent NMF features of the dataset, which can therefore enhance the quality of the reconstructions by enabling regularisation of temporal and spatial features separately.

This can be achieved by including a discrepancy term \( D_{IP}(Y_{*,t}, A_t X_{*,t}) \) of the inverse problem into the NMF cost function in (5). This leads together with some possible penalty terms for the reconstruction \( X \) to the model

\[
\min_{B,C,X \geq 0} \{ T \sum_{t=1}^{T} \| A_t X_{*,t} - Y_{*,t} \|^2 + \frac{\alpha}{2} \| BC - X \|^2_F + \lambda_B \| B \|_1 + \frac{\mu_B}{2} \| B \|^2_F + \lambda_C \| C \|_1 + \frac{\mu_C}{2} \| C \|^2_F + \lambda_X \| X \|_1 + \frac{\mu_X}{2} \| X \|^2_F + \tau \frac{1}{2} \text{TV}(B) \},
\]

with \( \alpha \geq 0 \) for the joint reconstruction and low-rank decomposition problem, which we will call **BC-X**. Furthermore, we can enforce \( X := BC \) as a hard constraint such that the reconstruction matrix will have at most rank \( K \). In this case, the discrepancy \( D_{NMF} \) vanishes and we end up with the model **BC**:

\[
\min_{B,C \geq 0} \{ D_{IP}(Y_{*,t}, A_t (BC)_{*,t}) + \sum_{t=1}^{L} \gamma_t \mathcal{P}_t(B, C, X) \}.
\]

2.4.1. **Considered NMF models.** For both models (7) and (8), we use the standard Frobenius norm for both the discrepancy terms \( D_{NMF} \) and \( D_{IP} \). Furthermore, the optimisation method discussed in Section 2.4.2 allows to include a variety of penalty terms into the cost function. This makes it possible to construct suitable regularised NMF models and to enforce additional properties to the matrices depending on the specific application. For the theoretical part of this work and to show that the optimisation approach can generalise to various regularisation terms, we will consider standard \( \ell_1 \) and \( \ell_2 \) terms on each matrix and an isotropic total variation penalty on the matrix \( B \). The latter is motivated by our considered application in Section 3, which denoises the spatial features and thus also the reconstruction matrix. Hence, we will focus on the following NMF models in the remainder of this work:
\[
\min_{B,C \geq 0} \left\{ \sum_{t=1}^{T} \frac{1}{2} \| A_t(BC)_{:,t} - Y_{:,t} \|^2_2 + \lambda_C \| C \|_1 + \frac{\mu_C}{2} \| C \|_2^2 + \lambda_B \| B \|_1 + \frac{\mu_B}{2} \| B \|_2^2 + \frac{\tau}{2} TV(B) \right\}.
\]

The regularisation parameters \( \alpha, \lambda_C, \mu_C, \lambda_B, \mu_B, \lambda_X, \mu_X, \tau \geq 0 \) control the influence of the different penalty terms. Furthermore, \( \| \cdot \|_F \) denotes the Frobenius norm, \( \| M \|_1 := \sum_{ij} | M_{ij} | \) the 1-norm for matrices \( M \) and \( TV(\cdot) \) is the following smoothed isotropic total variation \([14, 19, 33]\).

**Definition 2.1.** The total variation of a matrix \( B \in \mathbb{R}^{N \times K} \) is defined as

\[
TV(B) := \sum_{k=1}^{K} \sum_{n=1}^{N} \sqrt{\varepsilon_{TV}^2 + \sum_{\ell \in \mathcal{N}_n} (B_{nk} - B_{\ell k})^2},
\]

where \( \varepsilon_{TV} > 0 \) is a small positive constant and \( \mathcal{N}_n \) are index sets referring to spatially neighboring pixels.

For the rest of this work, we will use the following shorthand notation:

\[
| \nabla_{nk} B | := \sqrt{\varepsilon_{TV}^2 + \sum_{\ell \in \mathcal{N}_n} (B_{nk} - B_{\ell k})^2}.
\]

A typical example for the neighbourhood of the pixel \((0,0)\) in two dimensions is \( \mathcal{N}_{(0,0)} = \{(1,0), (0,1)\} \) to get an estimate of the gradient components in both directions of the axes. The parameter \( \varepsilon_{TV} \) ensures the differentiability of the TV penalty term. This is needed due to the considered MM principle for the optimisation approach and to avoid singularities in the arising matrices of the derived algorithms, since TV regularisation typically has the tendency to set first-order differences to zero (see \([41, 14]\) and Section 2.4.2). This modification leads to a differentiable cost function for both NMF models in BC and BC-X and allows further possible convergence results of the obtained algorithms as it was done in \([14]\) with a different cost function. A small choice of \( \varepsilon_{TV} > 0 \) guarantees that \( TV(\cdot) \) is close to the “true” discrete TV penalty.

Note that the \( \ell_1 \) penalty term does not lead to any difficulties regarding the differentiability of the cost function and the used optimisation approach, since both factorisation matrices \( B \) and \( C \) are constrained to be nonnegative. This fact becomes crucial for the optimisation procedure especially concerning the application of the MM principle and the construction of suitable surrogate functions based on the quadratic upper bound principle (see Appendix A.1).

The MM principle gives the needed flexibility to deal with such generalised cost functions in BC-X and BC together with the non-standard discrepancy term and all considered regularisation terms. In Appendix B, we provide a detailed derivation of the algorithms and show that the update rules lead to a monotonic decrease of the cost functions in BC-X and BC. For the subsequent numerical evaluation in Section 3, the penalty terms of both NMF models will be restricted to the ones which will be relevant for the considered application.

At this point, we would like to provide a short comparison of the NMF model in BC and the considered method in the work \([8]\), which introduces a comparable combined matrix factorisation model for the application to cine cone-beam CT (cine-CBCT). The authors consider a similar discrepancy term as in BC, where each
column in the matrix product $BC$ corresponds to a cine-CBCT image at a specific point in time with a corresponding projection angle. Furthermore, their model considers an upper limit for the discrepancy term as a hard constraint and an $\ell_2$ as well as a sparsity penalty term on $TB$ with $T$ being a specific wavelet transform, which allows the reconstruction of the lung motion during a CT measurement based on the few basis vectors given by the columns in $B$ and the coefficients in $C$. Different from their approach, we rather focus on PCA approaches and in particular multiple NMF models together with their nonnegativity constraints to improve the physical and anatomic interpretation of the features in $B$. Furthermore, we include a TV regularisation procedure to denoise the spatial basis functions as well as the reconstructions in $X$ and show for all joint reconstruction approaches based on the Algorithms given in Theorem 2.2 and 2.3 the monotone decrease of the cost function, which constitutes the major theoretical part of this work.

2.4.2. Algorithms. In this section, we present in Theorem 2.2 and 2.3 the multiplicative algorithms for the NMF problems in $BC-X$ and $BC$. As mentioned in Section 2.2.2, the multiplicative structure of the iteration scheme ensures automatically the nonnegativity of the matrices $B$ and $C$ as long as they are initialised nonnegative. The derivation of such algorithms in this work are based on the MM principle. The main idea of this approach is to replace the considered NMF cost function $F$ with a suitable auxiliary function $Q_F$, whose minimisation is much easier to handle and leads to a monotone decrease of $F$. Furthermore, specific construction techniques of these surrogate functions lead to the desired multiplicative update rules, which fulfill the nonnegativity constraint. We provide a short description of the main principles in Appendix A. A more detailed discussion of different construction methods for various kinds of discrepancy and penalty terms of $F$ can be found in the survey paper [19].

For better readability, we present here only the main results. A detailed construction of the surrogate functions as well as derivation of the algorithms for both cost functions $BC-X$ and $BC$ can be found in Appendix B. Consequently, we will only state the main results in Theorem 2.2 and 2.3 here. Nevertheless, due to the construction of a suitable surrogate function for the TV penalty term (see Appendix B and [19] for more details), we first introduce the following matrices $P(B), Z(B) \in \mathbb{R}_\geq 0^{N \times K}$ as

\begin{align}
\text{(9)} \quad P(B)_{nk} := \frac{1}{|\nabla_{nk} B|} \left( 1 + \sum_{\ell \in N_n} \frac{1}{|\nabla_{\ell k} B|} \right), \\
\text{(10)} \quad Z(B)_{nk} := \frac{1}{P(B)_{nk}} \left( \frac{1}{|\nabla_{nk} B|} \sum_{\ell \in N_n} \frac{B_{nk} + B_{\ell k}}{2} + \sum_{\ell \in N_n} \frac{B_{nk} + B_{\ell k}}{2|\nabla_{\ell k} B|} \right),
\end{align}

where $N_n$ is the set of the so-called adjoint neighbourhood pixels, which is given by the relation

$$
\ell \in N_n \leftrightarrow n \in N_{\ell}.
$$

The differentiable approximation of the TV penalty term with $\varepsilon_{TV} > 0$ ensures that both matrices $P(B)$ and $Z(B)$ are well-defined for all $B \in \mathbb{R}_{\geq 0}^{N \times K}$. Furthermore, we write $1_{M \times N}$ for an $M \times N$ matrix with ones in every entry.

We then obtain the two algorithms for both models under consideration. First for the $BC-X$ model that jointly obtains the reconstruction $X$ and the decomposition:
Theorem 2.2 (Algorithm for BC-X). For $A_t \in \mathbb{R}_{\geq 0}^{M \times N}$, $Y \in \mathbb{R}_{\geq 0}^{M \times T}$ and initialisations $X^{0} \in \mathbb{R}_{\geq 0}^{N \times T}$, $B^{0} \in \mathbb{R}_{\geq 0}^{N \times K}$, $C^{0} \in \mathbb{R}_{\geq 0}^{K \times T}$, the alternating update rules

$$X_{\bullet,t}^{[d+1]} = X_{\bullet,t}^{[d]} \circ \frac{A_t^\top Y_{\bullet,t} + \alpha B^{[d]} C^{[d]}_{\bullet,t}}{A_t^\top A_t X_{\bullet,t}^{[d]} + (\mu_X + \alpha) X_{\bullet,t}^{[d]} + \lambda_X I_{N \times 1}}$$

$$B^{[d+1]} = B^{[d]} \circ \frac{\alpha X_{\bullet,t}^{[d]} C^{[d]}_{\bullet,t} + \mu_B B^{[d]} + \lambda_B I_{N \times K} + \tau B^{[d]} \circ P(B^{[d]})}{\alpha B_{\bullet,t}^{[d]} X_{\bullet,t}^{[d+1]} + \mu_B C^{[d]}_{\bullet,t} + \lambda_B I_{K \times T}}$$

$$C_{\bullet,t}^{[d+1]} = C_{\bullet,t}^{[d]} \circ \frac{B_{\bullet,t}^{[d+1]} A_t^\top Y_{\bullet,t} + \mu_C C_{\bullet,t}^{[d]} + \lambda_C I_{K \times 1}}{B_{\bullet,t}^{[d+1]} A_t (B_{\bullet,t}^{[d+1]} C_{\bullet,t}^{[d]} + \mu_C C_{\bullet,t}^{[d]} + \lambda_C I_{K \times 1}}$$

lead to a monotonic decrease of the cost function in BC-X.

Similarly, for the BC model we obtain the updates rules without constructing the matrix $X$ during the reconstruction process:

Theorem 2.3 (Algorithm for BC). For $A_t \in \mathbb{R}_{\geq 0}^{M \times N}$, $Y \in \mathbb{R}_{\geq 0}^{M \times T}$ and initialisations $B^{0} \in \mathbb{R}_{\geq 0}^{N \times K}$, $C^{0} \in \mathbb{R}_{\geq 0}^{K \times T}$, the alternating update rules

$$B^{[d+1]} = B^{[d]} \circ \frac{\sum_{t=1}^{T} A_t^\top Y_{\bullet,t} \cdot (C_{\bullet,t}^{[d]} + \mu_P B^{[d]} + \lambda_P I_{K \times T}) \circ Z(B^{[d]})}{\sum_{t=1}^{T} A_t^\top A_t (B^{[d]} C_{\bullet,t}^{[d]} + \mu_B B^{[d]} + \lambda_B I_{N \times K} + \tau B^{[d]} \circ P(B^{[d]})}$$

$$C_{\bullet,t}^{[d+1]} = C_{\bullet,t}^{[d]} \circ \frac{B_{\bullet,t}^{[d+1]} A_t^\top Y_{\bullet,t} + \mu_C C_{\bullet,t}^{[d]} + \lambda_C I_{K \times 1}}{B_{\bullet,t}^{[d+1]} A_t (B_{\bullet,t}^{[d+1]} C_{\bullet,t}^{[d]} + \mu_C C_{\bullet,t}^{[d]} + \lambda_C I_{K \times 1}}$$

lead to a monotonic decrease of the cost function in BC.

We remind that the derivation is described in Appendix B, which leads to the update rules in the Theorems above. Due to the multiplicative structure of the algorithms, zero entries in the matrices stay zero during the iteration scheme and can cause divisions by zero. This issue is handled via the strict positive initialisation in both Theorems. Furthermore, very small or high numbers can cause numerical instabilities and lead to undesirable results. As a standard procedure, this problem is handled by suitable projection steps after every iteration step [12].

2.5. Complexity reduction for stationary operator. Let us now consider the case of a stationary operator, i.e. $A(\cdot;t)$ in equation (1) does not change with $t$. Then we simply write $A$ or $A$ for the matrix representation in (3). If further the number of channels $T$ is large, the application of the forward operator represented a major computational burden per channel. In particular, we make use here of the assumption $T \gg K$, i.e. the number of channels is much larger than the basis functions for the decomposition. In this case, we can effectively reduce the computational cost by shifting the application of the forward operator to the spatial basis functions contained in $B$. That means, we make essential use of the decomposition $X \approx BC$ in the reconstruction task and as such avoid to construct the approximation to $X$. Consequently, we will only consider the case of BC here. Since $A$ is independent from $t$, the NMF model BC becomes

$$\text{sBC} \min_{B,C \geq 0} \left\{ \frac{1}{2} \left\| ABC - Y \right\|_F^2 + \lambda_C \left\| C \right\|_1 + \frac{\mu_C}{2} \left\| C \right\|_F^2 + \lambda_B \left\| B \right\|_1 + \frac{\mu_B}{2} \left\| B \right\|_F^2 + \frac{\tau}{2} \text{TV}(B) \right\}.$$
To illustrate this, let us consider the update equation in Theorem 2.3 for $B$, where we can simplify the first term in the denominator as follows:

$$\sum_{t=1}^{T} A^T A (B^{[d]} C^{[d]})_{\bullet \bullet} = A^T A \sum_{t=1}^{T} (B^{[d]} C^{[d]})_{\bullet \bullet} (C^{[d]})_{\bullet \bullet} = A^T A B^{[d]} C^{[d]} C^{[d] \top}.$$ 

The other terms in the update rules can be simplified similarly such that we obtain the following reduced update equations.

**Corollary 1 (Algorithm for sBC).** For $A \in \mathbb{R}^{M \times N}$, $Y \in \mathbb{R}^{M \times T}$ and initialisations $B^{[0]} \in \mathbb{R}^{N \times K}$, $C^{[0]} \in \mathbb{R}^{K \times T}$, the alternating update rules

$$B^{[d+1]} = B^{[d]} \circ \frac{A^T Y C^{[d] \top} \tau} {A^T A B^{[d]} C^{[d]} C^{[d] \top} + \mu_B B^{[d]} + \lambda_B I_{N \times K} + \tau B^{[d]} \circ P(B^{[d]})}$$

$$C^{[d+1]} = C^{[d]} \circ \frac{Y A B^{[d+1]} C^{[d]} + \mu_C C^{[d]} + \lambda_C I_{K \times T}} {B^{[d+1]} A^T Y},$$

lead to a monotonic decrease of the cost function in sBC.

Finally, the order of application is essential here to obtain the complexity reduction. In particular, we implemented the algorithm such that $A$ acts on the basis functions in $B$. That means, we compute first the product $A^T A B$ followed by multiplication with $C$. That means, we can expect a reduction of computational complexity by a factor $T/K$ with the sBC model being especially useful for dimension reduction under fine temporal sampling.

3. **Application to dynamic CT.** In the following, we will apply the presented methods to the use case of dynamic computerised tomography (CT). Here, the quantity of interest is given as the attenuation coefficient $x(s,t)$ at time $t \in [0,T]$ on a bounded domain in two dimensions $s \in \Omega_1 \subset \mathbb{R}^2$. Following the formulation in (1), the time-dependent forward operator is given by the Radon transform

$$y(\theta,\sigma,t) := (R_{\mathcal{I}(t)} x(s,t))(\theta,\sigma) = \int_{s \in \mathcal{I}(t)} x(s,t) ds.$$ 

Here, the measurement $y(\theta,\sigma,t)$ consist of line integrals over the domain $\Omega_1$ for each time point $t \in \mathcal{T}$ and is referred to as the sinogram. This measurement depends on two parameters, the angle $\theta \in S^1$ on the unit circle and a signed distance to the origin $\sigma \in \mathbb{R}$. Consequently, the measurements depend on a set of angles at each time step $\mathcal{I}(t)$ such that $(\theta,\sigma) \in \mathcal{I}(t)$ at time $t$, which we will refer to this as the sampling patterns. In a slight abuse of notation, we will use $|\mathcal{I}(t)|$ for the number of angles, i.e. directions for the line integrals, at each time point.

In the following, we consider two scenarios for the choice of angles in $\mathcal{I}(t)$ and by that defining the nature of the forward operator as discussed in Section 2.1. In the general case of a nonstationary forward operator, that means the sampling patterns are time-dependent, we assume that the angles change but the amount of angles is constant over time $|\mathcal{I}(t)| \equiv c$. Additionally, we will consider the case for stationary operators, which means in our setting that the set of angles does not change over time. Hence, this leads to a stationary measurement operator of the dynamic process in (11) such that we can write $\mathcal{I}(t) \equiv \mathcal{I}(t = 0)$. We note that even though the measurement process is stationary, the obtained measurement $y(\theta,\sigma,t)$ itself is still time dependent.

For the computations, we discretise (11) to obtain a matrix vector representation as in (3). We will write $R_t$ for the discrete Radon transform with respect to the
sampling pattern $\mathcal{I}(t)$ at time point $t$, which gives rise to the discrete reconstruction problem for dynamic CT

\begin{equation}
R_t X_{\bullet,t} = Y_{\bullet,t} \quad \text{for} \quad 1 \leq t \leq T.
\end{equation}

We note that due to the definition of the Radon transform by line integrals, the matrix $R_t \in \mathbb{R}^{M \times N}$ has only nonnegative entries and hence satisfies the assumption for Theorem 2.2 and 2.3. Furthermore, $N$ denotes here the number of pixels in the original image and $M$ is given by the product $M := |\mathcal{I}(t)|n_S$, where $n_S$ is the number of detection points.

3.1. Results and discussion. For a qualitative evaluation of the proposed NMF approaches, we consider in the following sections three simulated datasets. Due to the known ground truth in all cases, we are able to measure the performance of each method via computing the mean of the Peak Signal to Noise Ratio (PSNR) and the mean of the Structural Similarity Index Measure (SSIM) \cite{4} over all time steps for every experiment.

For each dataset, the parameters of all methods are chosen empirically by performing various experiments for a wide range of the considered regularization parameters. The parameter configuration is selected, which leads to the best results in terms of quality and stability. For the NMF models of the joint reconstruction and low-rank decomposition approach, we restrict ourselves to the total variation penalty term on $B$ to provide some denoising effect on the spatial features and the $\ell_2$ penalty on $C$ for the time features, since we expect and enforce smooth changes in time. We consider the standard case for the TV term with the default pixel neighbourhood and choose the smoothing parameter $\varepsilon_{TV} = 10^{-5}$ relatively small.

Concerning the choice of the number of features $K$, we follow the heuristic approach in \cite{8} by analysing the importance of the rank-one matrices $B_{\bullet,k}C_{k,\bullet}$. As a first step, $K$ is chosen large enough and the considered algorithms are executed after the optimal regularisation parameters are found (see Appendix C.2). In such a way, the algorithm is forced to extract additional unnecessary features, which in turn can be identified based on the small norm of the rank-one matrices $B_{\bullet,k}C_{k,\bullet}$. Following \cite{8}, we use in this work the $\infty$-norm $\|B_{\bullet,k}C_{k,\bullet}\|_\infty$, which is defined as the maximum absolute row sum of the matrix corresponding to the pixel of the reconstruction $B_{\bullet,k}C_{k,\bullet}$, which has the largest absolute sum over time. The plots of these values together with a short description regarding the choice of $K$ for all datasets can be found in Appendix C.1.

Furthermore, for both datasets we measure different angles at each time step based on a tiny golden angle sampling \cite{51} using consecutive projections with increasing angle of $\varphi = 32.039\ldots$, such that projection angles are not repeated. Nevertheless, we remind that we keep the total number of observed angles constant for each time step.

For all considered approaches, we use the unfiltered backprojection, given by the adjoint of the Radon transform, applied to the noisy data matrix $Y$ as the initialisation for the reconstruction matrix $X$. In case of the NMF approaches, the matrices $B$ and $C$ are initialised via the SVD of $X$ based on \cite{3}. After the initialisation and at every iteration of the NMF algorithm, a suitable projection step for small values is performed to prevent numerical instabilities and zero entries during the multiplicative algorithm \cite{12}.
The algorithms were implemented with MATLAB® R2019b and run on an Intel® Core™ i7-7700K quad core CPU @4.20 GHz with 32 GB of RAM. The codes are available online in our GitLab [1].

To this end, we present a summary and short explanation of all considered algorithms in this experimental section in Table 1.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC</td>
<td>Joint reconstruction and feature extraction with the NMF model BC without constructing X, see algorithm in Theorem 2.3</td>
</tr>
<tr>
<td>BC-X</td>
<td>Joint reconstruction and feature extraction with NMF model BC-X and explicit construction of X, see algorithm in Theorem 2.2</td>
</tr>
<tr>
<td>sBC</td>
<td>Joint reconstruction and feature extraction method with NMF model sBC for stationary operator, see algorithm in Corollary 1</td>
</tr>
<tr>
<td>gradTV</td>
<td>Low-rank based reconstruction method for X, see Algorithm 1</td>
</tr>
<tr>
<td>gradTV_PCA</td>
<td>Separated reconstruction and feature extraction with Algorithm 1 and subsequent PCA computation</td>
</tr>
<tr>
<td>gradTV_NMF</td>
<td>Separated reconstruction and feature extraction with Algorithm 1 and subsequent NMF computation based on the model in (6)</td>
</tr>
</tbody>
</table>

Table 1. Summary and short explanation of the considered algorithms in the experimental section.

3.1.1. Shepp-Logan phantom. This synthetic dataset consists of a dynamic two-dimensional Shepp-Logan phantom with $T = 100$ and a spatial size of $128 \times 128$ (see Figure 1 for the ground-truth). During the whole time, two of the inner ellipsoids change their intensities sinusoidally with different frequencies, while the rest of the phantom remains constant.

In the following, we perform a variety of experiments for $|I_t| \in \{2, \ldots, 12\}$ with 1% and 3% Gaussian noise respectively. For all cases, we choose $K = 5$ for the number of the NMF features based on the findings in Appendix C.1.

The other parameters of all methods are determined empirically and are displayed in Table 3 in Appendix C.2 for both noise levels. The stopping criterion for all methods is met, if 1200 iteration steps are reached or if the relative change of all matrices $B, C$ and $X$ goes below $5 \cdot 10^{-5}$.

We show first some results for the case with $|I_t| = 6$ and 1% Gaussian noise in Figure 3 for the joint NMF methods and Figure 4 for the separate reconstruction and extraction. The order of shown features is based on the singular values of $B$ for gradTV_PCA and on the $\ell_2$-norm of the spatial features for NMF approaches.

In this case, all considered approaches are able to successfully identify the constant and dynamic parts of the dataset and extract meaningful spatial and temporal features. The extracted spatial features of BC, BC-X and gradTV_NMF show very clearly the dynamic and non-dynamic parts of the Shepp-Logan phantom. However, the spatial features of gradTV_NMF are slightly more blurred and affected by minor artefacts especially in both dynamic features. This underlines the positive effect of the separate TV regularisation on the spatial feature matrix $B$ in the joint methods. In contrast, gradTV_PCA is able to identify the main components of the dataset correctly, but there is a clear corruption of the dynamic features with other parts from the phantom. Furthermore, all spatial features contain negative parts due to the non-existent nonnegativity constraint of the gradTV_PCA approach which makes
Figure 3. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.1 with $|I_t| = 6$ angles per time step and 1% Gaussian noise. Shown are the leading extracted features for the BC model and for BC-X.

Figure 4. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.1 with $|I_t| = 6$ angles per time step and 1% Gaussian noise. Shown are the leading extracted features for the gradTV PCA model and for gradTV NMF.
their interpretation more challenging. Hence, the additional nonnegativity constraint of the NMF methods improves significantly the quality and interpretability of the extracted components in comparison with the PCA based extraction method. The temporal features of all methods are clearly extracted and are consistent with the underlying ground truth of the dataset. However, we note that BC and BC-X have a slight difficulty to resolve the lower intensity part close to 0, which is probably caused by the multiplicative structure of the algorithms. The different magnitudes in the temporal features arise from the soft constraints being applied on the matrix $C$ and is primarily dependent on the choice of the regularisation parameter $\mu_C$ of the $\ell_2$ penalty on $C$.

![Figure 5](image_url)

**Figure 5.** Results for the dynamic Shepp-Logan phantom considered in Section 3.1.1 with $|\mathcal{I}| = 6$ angles per time step and 3\% Gaussian noise. Shown are the leading extracted features for the BC model and for gradTV_PCA. Similar observations can be made for the case $|\mathcal{I}| = 6$ and 3\% Gaussian noise. We present the reconstructed features in Figure 5 for BC and gradTV_PCA only. The higher amount of noise can be observed especially in the spatial features of gradTV_PCA, whereas it only has a slight effect in the BC model. Finally, we present the reconstructed features with BC and BC-X in Figure 6 for $|\mathcal{I}| = 3$, i.e. only three three angles per time step with a noise level of 1\%. The major difference to the previous cases can be seen in the results of the BC model. Here, the method splits up the dynamics of the right ellipse into two different temporal features, such that the true dynamics are not retained. However, the BC-X approach perform remarkably well with respect to the feature extraction despite the rather low number of projection angles. This might indicate, that enforcing the reconstruction $X$ to have small data error helps in the BC-X model to stabilise the reconstruction in highly sparse data settings.
Figure 6. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.1 with \(|I_t| = 3\) angles per time step and 1% Gaussian noise. Shown are the leading extracted features for the BC model and for BC-X.

Let us shortly discuss other considered values of \(|I_t|\), that are not shown here. First of all, the performance of gradTV\_PCA and gradTV\_NMF with respect to the feature extraction behaves very similar for both noise cases. Besides the above mentioned drawbacks, both approaches give remarkably consistent results especially for low number of angles and do not tend as much to split up features like in BC and BC-X. The latter occurs in different degrees for several numbers of angles. For 1% noise, it occurs for \(|I_t| \in \{3, 7, 8, 10\}\) in BC and for \(|I_t| = 10\) in BC-X. In the case of a noise level of 3%, the split-up effect only occurs for \(|I_t| = 10\) in BC. However, for \(|I_t| = 10\), it is possible to partially recover the correct temporal feature by simply adding up both features. Nevertheless, both approaches provide better reconstruction quality of \(X\) than gradTV as we will discuss in the following.

**Quantitative Evaluation.** Let us now discuss the quantitative reconstruction quality for all methods. In Figure 7 and 8, we show the mean PSNR and SSIM of the reconstructions for 1% and 3% noise over all time steps for all considered numbers of projection angles. Note that for the NMF model BC-X, we compute the quality measures for \(X\). The same goes for gradTV, where we only compute the quality measures of \(X\) after the reconstruction procedure independently of the subsequent feature extraction method. In the case of BC, the reconstruction is computed as \(X = BC\).

As expected, the reconstruction quality tends to get better if more angles per time step are considered. More importantly, we see that it is possible to obtain reasonable reconstructions with just a few projections per time step especially in the case of the joint reconstruction and feature extraction method via the NMF.
approach. In particular, we reach a stable reconstruction quality already with 5 or more angles for both joint methods and 1% noise.

![Mean PSNR and SSIM values of the reconstructions of the dynamic Shepp-Logan phantom considered in Section 3.1.1 with 1% Gaussian noise for different numbers of projection angles.](image)

<table>
<thead>
<tr>
<th>Number of angles per time step</th>
<th>Mean PSNR</th>
<th>Mean SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Mean PSNR and SSIM values of the reconstructions of the dynamic Shepp-Logan phantom considered in Section 3.1.1 with 3% Gaussian noise for different numbers of projection angles.](image)

<table>
<thead>
<tr>
<th>Number of angles per time step</th>
<th>Mean PSNR</th>
<th>Mean SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The BC model clearly performs best with respect to the reconstruction quality. For almost every number of angles, the mean PSNR and SSIM values outperform the ones of the BC-X and gradTV method for both noise levels. In the case of 3% noise (see Figure 8) we can see that gradTV performs slightly better than BC-X in most of the cases in terms of their SSIM values. Still, the mean PSNR values of gradTV are significantly lower than the ones in BC-X for all numbers of angles. A
selection of reconstructions for the experiments in Figure 7 and 8 are provided as videos in our GitLab [1].

Note that for BC-X, it is also possible to compute the reconstruction based on the decomposition $B \cdot C$ instead of the joint reconstruction $X$ in the algorithm. Interestingly, our experiments showed that the reconstruction quality of $B \cdot C$ is in almost all cases better than the one of the matrix $X$ itself and also mostly outperforms the gradTV approach. We believe, that this is due to the stronger regularising effect on the components $B$ and $C$, which especially influences the SSIM.

The computation times for the reconstruction and feature extraction with 1% noise for all algorithms until the stopping criterion is fulfilled are shown in Figure 9. As expected, the computation time tends to increase with the number of projection angles and, considering all methods, ranges approximately from 1 to 5 minutes. For $|I_t| \leq 8$, the BC-X method is the fastest while it is outperformed by gradTV_PCA for $|I_t| \geq 9$. gradTV_NMF and BC needs more time in all experiments compared to gradTV_PCA. The significant temporal difference between BC-X and BC is due to its higher computational complexity: Owing to the model formulation of BC with the discrepancy term $\|R_t(BC)_{\bullet, t} - Y_{\bullet, t}\|^2_2$, the update rules in Theorem 2.3 for both matrices $B$ and $C$ contain the discretised Radon transform $R_t$. This is in contrast to the BC-X algorithm, where $R_t$ only appears in the update rule of $X$.

Based on the presented results for the dynamic Shepp-Logan phantom, we can conclude that the joint approaches BC and BC-X outperform both other methods with respect to the reconstruction quality and for most cases of the extracted features. Nevertheless, the models gradTV_PCA and gradTV_NMF give remarkably consistent and stable results of the extracted components throughout all numbers of angles. Furthermore, the nonnegativity constraint of the NMF improves significantly the interpretability and quality of the extracted spatial features.

Stationary Operator. As we have seen, the computational complexity of the BC model with the non-stationary operator is clearly higher than for all other cases.
Thus, let us now consider the possibility to speed up the reconstructions with a stationary operator, which leads us to the complexity reduced formulation presented in Corollary 1 as the sBC model. Here, we present similarly to the case above experiments with the same dynamic Shepp-Logan phantom for $|I_t| \in \{2, \ldots, 30\}$ and 1% Gaussian noise, as we primarily aim to illustrate the reduction of the computational cost. Furthermore, the same hyperparameters and stopping criteria are used as before.

The reconstructed features for the cases $|I_t| = 6$ and $|I_t| = 30$ are shown in Figure 10. In particular, comparing the results in Figure 10a to the corresponding results of BC in Figure 3a, one can immediately see a significant difference between the extracted spatial features. This is due to the fact that the same projection angles are used at every time step and leads to the clearly visible individual projection directions for the stationary model sBC. Consequently, the details in the Shepp-Logan phantom are not well recovered, such that the extracted constant feature is significantly inferior to the one of BC. As one would expect, more projection angles per time step are needed to reconstruct finer details. This effect can be clearly seen for 30 angles in Figure 10b.

However, all temporal basis functions with sBC for $|I_t| = 6$ are remarkably well reconstructed despite the low number of projection angles. This is also true for the other considered values of $|I_t|$. Moreover, we observe that sBC is able to extract the correct three main features for every $|I_t| \in \{2, \ldots, 30\}$.

This behaviour is different from the dynamic case discussed above. The reason for this is probably based on the different projection directions at every time step in the dynamic case, which results in directional dependencies of the occurring reconstruction artefacts in contrast to the stationary case. This can make it difficult for the NMF to distinguish the main features in the non-stationary case and thus leads to a more stable feature extraction in the here presented stationary case.

The quantitative measures are shown in Figure 11 for all experiments. Comparing the computation time of BC with the one of sBC, we obtain a clear speed-up by a factor of 10–20 with the stationary model. However, as expected, comparing Figure 11b and 11c with the quality measures of BC in Figure 7, one can observe that significantly more projection angles per time step are needed in the stationary case to provide a sufficient reconstruction quality. In conclusion, we can say that the sBC model is especially recommended if one is primarily interested in the dynamics of the system under consideration, as we could extract the temporal basis functions stably for all considered angles with $|I_t| \geq 2$.

A selection of the reconstructions are available as video files in our GitLab [1].

3.1.2. Vessel phantom. The second test case is based on a CT scan of a human lung, see Figure 12. Here, the decomposition is given by the constant background and a segmented vessel that exhibits a sudden increase in attenuation followed by an exponential decay. This could for instance represent the injection of a tracer to the blood stream.

In contrast to the previous dataset, we perform only selected experiments for specific choices of noise levels and numbers of projection angles. More precisely, we present results for 1% Gaussian noise together with $|I_t| \in \{7, 12\}$ and 3% Gaussian noise.

---

The phantom is based on the CT scans in the ELCAP Public Lung Image database: http://www.via.cornell.edu/lungdb.html
Results for the dynamic Shepp-Logan phantom considered in Section 3.1.1 with a stationary operator and 1% Gaussian noise. Shown are the leading extracted features with the sBC model for $|I_t| = 6$ angles per time step and $|I_t| = 30$.

Figure 11. Needed time in seconds, mean PSNR and mean SSIM values of the reconstructions of the dynamic Shepp-Logan phantom with 1% Gaussian noise for the stationary case sBC and different numbers of projection angles.

In all cases, we choose $K = 4$ NMF features based on the findings in Appendix C.1. Furthermore, the stopping criterion from the experiments with the dynamic Shepp-Logan phantom is changed for this dataset in such a way, that the maximum number of iterations is raised to 1400 to ensure sufficient convergence. The choice of the regularisation parameters of all methods are displayed in Table 4 in Appendix C.

Figure 13 and 14 show the feature extraction results for the noise level of 1% and $|I_t| = 12$, where all approaches are able to extract both the main constant and
Figure 12. Illustration of the vessel phantom dataset consisting of $T = 100$ phantoms of dimension $264 \times 264$, where the intensity of the blue highlighted area changes according to blue curve on the left.

Figure 13. Results for the vessel phantom with $|Z_t| = 12$ angles per time step and 1% Gaussian noise. Shown are the leading extracted features for the BC model and for BC-X.

dynamic component of the underlying ground truth. The order of the features here is based on a manual sorting.

Similar to the results for the Shepp-Logan phantom in Section 3.1.1, the joint methods BC and BC-X have difficulties to recover the lower intensities in the temporal features, whereas gradTV\_PCA produce slight artefacts in the dynamic spatial feature due to the missing nonnegativity constraint. In addition, gradTV\_NMF is able to recover more details in the vessel compared to the joint approaches. This is due to the relatively high choice of the total variation regularisation parameter $\tau$ in BC and BC-X to ensure a sufficient denoising effect on the matrix $B$. The low peak in the second temporal feature of gradTV\_NMF is caused by the choice of the $\ell_2$ regularisation parameter $\tilde{\mu}_C$.
Further experiments show that the quality of the extracted components of $\text{BC-X}$ decreases steadily for lower angles until the main features cannot be identified anymore for $|I_t| \leq 8$. $\text{BC}$ produces inferior results and cannot extract reasonable components anymore for $|I_t| \leq 10$.

In comparison, both separated approaches $\text{gradTV}_{\text{PCA}}$ and $\text{gradTV}_{\text{NMF}}$ are still able to extract decent features for $|I_t| = 7$. For $|I_t| \leq 6$, the performance of both methods decreases significantly.
Similar results for gradTV,PCA can be obtained for 3% noise and $|I_t| = 12$, which are shown in Figure 15b. Its constant feature is inferior to the one of BC in Figure 15a due to the additional nonnegativity constraint of the NMF model. However, the details of the vessel in the dynamic spatial feature of BC are lost due to the total variation regularisation and the temporal features are affected by several disturbances. Further tests with the noise level of 3% showed that both joint methods are not able to recover the underlying features for $|I_t| \leq 10$, while the separated approaches gives still acceptable results for $|I_t| = 6$.

| Noise | $|I_t|$ | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
|-------|-------|------|------|------|------|------|------|
| 1%    | 7     | 34.130 | 0.9016 | 32.969 | 0.8382 | 31.463 | 0.8414 |
| 1%    | 12    | 35.050 | 0.9068 | 33.919 | 0.8496 | 34.309 | 0.8839 |
| 3%    | 12    | 30.148 | 0.7484 | 28.119 | 0.5708 | 29.375 | 0.6698 |

Table 2. Mean PSNR and SSIM values of the reconstruction results for the vessel phantom for different noise levels and numbers of projection angles. Values in brackets indicate that the dynamic part of the dataset in the corresponding experiment could not be reconstructed sufficiently well.

The reconstruction quality of the experiments are shown in Table 2. Similar to the Shepp-Logan phantom, the joint approach BC produces the best results compared to all other methods in terms of the mean PSNR and SSIM values. Further experiments confirm this observation for $4 \leq |I_t| \leq 11$.

However, these observations have to be treated with caution. BC is not able to recover the dynamics for $|I_t| \leq 10$ and 1% noise. In the case of BC-X, the dynamics can be reconstructed to some degree within the angle range $9 \leq |I_t| \leq 11$, but are not recognizable anymore for $|I_t| \leq 8$. In the case of 3% Gaussian noise, gradTV is still able to give acceptable reconstruction results for $|I_t| = 10$. For less angles, the reconstructed dynamics of gradTV get constantly worse until they are not apparent anymore for $|I_t| \leq 6$.

The computation times of the experiments in Table 2 range approximately from 7 to 15 minutes. The corresponding reconstructions can be found as video files in our GitLab [1].

3.1.3. Further experiments under violation of model assumptions. In this last part we will evaluate, if the proposed models are still able to provide useful reconstruction if the model assumption on the decomposition (2) is not fulfilled. In particular, this means we consider also spatial movements in the following. Similar to the dataset in Section 3.1.1, the ground truth in this section is also based on the Shepp-Logan phantom with $T = 100$, a spatial size of $128 \times 128$ and with the dynamics contained in the same ellipses. However, the right ellipse changes its intensity while the left ellipse in this dataset changes its size periodically violating the basic assumption of the NMF to be reconstructible as a superposition of just a few spatial and temporal features. Selected time steps of the ground truth are shown in Figure 16.
For this dataset, we perform experiments for \(|I_t| \in \{2, \ldots, 12\}\) and 1% Gaussian noise. As before, the choice of \(K\) is based on the \(\infty\)-norm of the rank-one matrices \(\|B_k \cdot C_k\|_{\infty}\). The corresponding details can be found in Appendix C.1. Based on those findings, we choose \(K = 3\) for BC-X and \(K = 4\) for BC and gradTV-NMF. Furthermore, we perform additional experiments for BC with \(K = 3\) and gradTV_NMF for \(K = 5\). Finally, the same stopping criteria are used as in Section 3.1.1 and the chosen regularisation parameters can be found in Table 5 of Appendix C.2.

Figure 17 and 18 show the extracted features for 12 angles per time step for every method with the respective choices of \(K\). The stationary feature as well as the right ellipse, whose dynamic is only given by a change of intensity, can be well extracted by every method. Similar to the previous experiments, BC has difficulties to reconstruct the lower intensities in the temporal features. Furthermore, BC and BC-X are able to extract the spatial movement of the left ellipse in the form of spatial features indicating the maximal and minimal spatial expansion of the ellipse together with their corresponding temporal components. Similar to the split-up effect in the previous results, BC splits the dynamics of the left ellipse to two different spatial features while including a slight amount of the dynamics of the right ellipse in the fourth spatial feature. This approximate decomposition can be attributed to the variational formulation of the reconstruction problem, where the found solution does minimise the cost function averaged over time.

Similar results are obtained for gradTV_PCA and gradTV_NMF in Figure 18. The main difference is that both methods are able to extract additional spatial and temporal features for the dynamics of the left ellipse showing the intermediate states of its movement, which leads finally to better reconstructions of the spatial movement based on the factorisation \(B \cdot C\).
Figure 17. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.3 with $|\mathcal{I}_t| = 12$ angles per time step and 1% Gaussian noise. Shown are the extracted features for BC and BC-X with $K = 4$ and $K = 3$ respectively.

Figure 18. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.3 with $|\mathcal{I}_t| = 12$ angles per time step and 1% Gaussian noise. Shown are the 5 leading extracted features for gradTV PCA and all features for gradTV NMF.
Figure 19. Results for the dynamic Shepp-Logan phantom considered in Section 3.1.3 with $|\mathcal{I}_t| = 12$ angles per time step and 1% Gaussian noise. Shown are the extracted features for BC with $K = 3$ and gradTV$_{NMF}$ with $K = 5$.

Figure 19 shows some additional experiments for BC and gradTV$_{NMF}$ with $K = 3$ and $K = 5$ respectively. Different from the previous results is that BC does not split up features anymore as it was the case for $K = 4$ and that the fifth feature of gradTV$_{NMF}$ leads to further intermediate states regarding the dynamics of the left ellipse. Hence, an accurate choice of $K$ is highly relevant for the obtained results of the joint reconstruction and low-rank decomposition approach.

Finally, Figure 20 shows some further quantitative results in terms of the mean PSNR and SSIM for all considered methods and $|\mathcal{I}_t| \in \{2, 3, \ldots, 12\}$. Similar to the previous findings in Section 3.1.1, the combined methods BC and BC-X outperform gradTV in terms of the mean PSNR especially for low number of angles. A selection of reconstructions is provided as videos in our GitLab [1].

4. Conclusion. In this work, we considered dynamic inverse problems with the assumption that the target of interest has a low-rank structure and can be efficiently represented by spatial and temporal basis functions. This assumption leaded naturally to a reconstruction and low-rank decomposition framework. In particular, we concentrated here on the nonnegative matrix factorisation as decomposition because it exhibits three main advantages:

i.) It naturally incorporates the physical assumption of nonnegativity

ii.) Basis functions are not restricted to being strictly orthogonal and therefore correspond more naturally to actual components
iii.) It allows the flexibility to incorporate separate regularisation on each of the factorisation matrices

In particular, the last point is of importance, as it allows to consider different regularisers for spatial and temporal basis functions, and as such can be tailored to different applications.

We then proposed two approaches to obtain a joint reconstruction and low-rank decomposition based on the NMF, termed BC-X and BC. Both methods performed better than a baseline method regarding the quality of the reconstructions, that computes a reconstruction with low-rank constraint followed by a subsequent decomposition. In particular, the second BC model has shown to have a stronger regularising effect on the reconstructed features as well as the reconstruction, which can be simply obtained as $X = BC$. We believe this is due to the fact, that only the decomposition is recovered during the reconstruction without the need to build the reconstruction $X$ explicitly and hence the resulting features at the end exhibit a higher regularity. More importantly, if one considers a stationary operator in the complexity reduced sBC model, we can obtain a considerable computational speed-up. Even though, due to constant projection angles the spatial basis functions are not as well recovered as in the non-stationary case, but the temporal features can be nicely extracted even for as low as 2 angles. This might be especially of interest in applications, where one is primarily interested in the underlying dynamics of the imaged target.

Despite the reconstruction principle of the NMF based on a superposition of just a few extracted features, we have demonstrated in Section 3.1.3, that all considered methods can still extract meaningful spatial and temporal features if the model assumptions are violated. Especially the separated methods gradTV_PCA and gradTV_NMF were able to reconstruct the spatial movement of the object adequately. We believe that this is partially due to the variational formulation of the NMF models which provides a solution in average over all time steps. Even though the presented methods are not primarily designed to reconstruct spatial movements.
in the target, this suggests that a combination with other approaches allowing for movements could be promising. For instance, optical flow constraints assume brightness consistency in the target [6] or morphological motion models [25] that allow for a flexible and general model for dynamic inverse problems. At the end, this underlines that the appropriate choice of model and hence the employed reconstruction algorithms depends on the application and needs to be chosen carefully.

Moreover, a more thorough numerical evaluation of the proposed methods on a clinical dataset would be of interest. Depending on the data, we note that the number of NMF features $K$ probably needs to be chosen higher than in the case of the simulated datasets discussed in Section 3. Regarding the a priori choice of the regularisation parameters, a further useful extension of the presented framework is the development of methods for making an optimal choice of these parameters.

Finally, further optimisation approaches could be investigated, such as the proximal alternating linearised minimisation scheme described in the recent work [17], which allows for non-smooth penalty terms. However, it is not clear if the optimisation method can be readily applied to the cost functions in BC and BC-X together with their non-standard discrepancy terms.

Acknowledgments. This project was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) within the framework of RTG “π³: Parameter Identification – Analysis, Algorithms, Applications” – Projektumnummer 281474342/GRK2224/1. This work was partially supported by the Academy of Finland Project 336796 (Finnish Centre of Excellence in Inverse Modelling and Imaging, 2018–2025) and Project 338408, EPSRC grant EDCLIRS (EP/N022750/1) as well as CMIC-EPSRC platform grant (EP/M020533/1).

Appendix A. Optimisation techniques for NMF problems. The majority of optimisation techniques for NMF problems are based on alternating minimisation schemes. This is due to the fact that the corresponding cost function in (5) is usually convex in $B$ and $C$ for fixed $C$ and $B$ respectively but non-convex in $(B, C)$ together, which yields algorithms of the form

$$B^{(d+1)} := \arg\min_{B \geq 0} F(B, C^{(d)}) ,$$

$$C^{(d+1)} := \arg\min_{C \geq 0} F(B^{(d+1)}, C) .$$

Typical minimisation approaches are based on alternating least squares methods, multiplicative algorithms as well as projected gradient descent and quasi-newton methods [12]. In this work, we focus on the derivation of multiplicative update rules based on the so-called Majorise-Minimisation (MM) principle [31]. This approach allows the derivation of multiplicative update rules for non-standard NMF cost functions and gives therefore the flexibility to adjust the discrepancy and penalty terms according to the NMF model motivated by the corresponding application [19]. What is more, the update rules consist only of multiplications and summations of matrices, which allow very simple implementations of the algorithms and ensure automatically the nonnegativity of the iterates $B$ and $C$ without the need of any projection steps, provided they are initialised nonnegative.

A.1. Multiplicative algorithms. The works of Lee and Seung [34, 35] brought much attention to NMF methods in general and, in particular, the multiplicative
algorithms, which they derived based on the MM principle for the standard case with the Frobenius norm and the Kullback-Leibler divergence as discrepancy terms.

The main idea of the MM approach is to replace the original cost function $F$ by a majorising so-called \textit{surrogate function} $Q_F$, which is easier to minimise and leads to the desired multiplicative algorithms due to its tailored construction.

**Definition A.1 (Surrogate Function).** Let $\Omega \subset \mathbb{R}^n$ be an open subset and $F : \Omega \to \mathbb{R}$ a function. Then $Q_F : \Omega \times \Omega \to \mathbb{R}$ is called a \textit{surrogate function} or \textit{surrogate} of $F$, if it fulfills the following properties:

i) $Q_F(x, \tilde{x}) \geq F(x)$ for all $x, \tilde{x} \in \Omega$

ii) $Q_F(x, x) = F(x)$ for all $x \in \Omega$

The minimisation step of the MM approach is then defined by the update rule

\[ x^{[d+1]} = \arg \min_{x \in \Omega} Q_F(x, x^{[d]}), \]

assuming that the $\arg \min_{x \in \Omega} Q_F(x, \tilde{x})$ exists for all $\tilde{x} \in \Omega$. Due to the defining properties of a surrogate function in Definition A.1, the monotonic decrease of the cost function $F$ is easily shown:

\[ F(x^{[d+1]}) \leq Q_F(x^{[d+1]}, x^{[d]}) \leq Q_F(x^{[d]}, x^{[d]}) = F(x^{[d]}). \]

This principle is also illustrated in Figure 21. Typical construction techniques lead to surrogate functions, which are strictly convex in the first component to ensure the unique existence of the corresponding minimiser. Furthermore, the surrogates must be constructed in such a way, that the minimisation in Equation (13) yields multiplicative updates to ensure the nonnegativity of the matrix iterates. Finally, another useful property is the separability of $Q_F$ with respect to the first variable. This ensures, that $Q_F(x, \tilde{x})$ can be written as a sum, where each component just depends on one entry of $x$ and allows the derivation of the multiplicative algorithm via the zero gradient condition $\nabla_x Q_F = 0$.

One typical construction method is the so-called \textit{Quadratic Upper Bound Principle} (QUBP) [2, 31], which forms one of the main approaches to construct suitable surrogate functions for NMF problems. Overviews of other construction principles,
which will not be used in this work, can be found in [31, 32]. The QUBP is described in the following Lemma.

**Lemma A.2.** Let $\Omega \subset \mathbb{R}^n$ be an open and convex subset, $\mathcal{F} : \Omega \to \mathbb{R}$ twice continuously differentiable with bounded curvature, i.e. there exists a matrix $\Lambda \in \mathbb{R}^{n \times n}$, such that $\Lambda - \nabla^2 \mathcal{F}(x)$ is positive semi-definite for all $x \in \Omega$. We then have

$$
\mathcal{F}(x) \leq \mathcal{F}(\tilde{x}) + \nabla \mathcal{F}(\tilde{x})^T (x - \tilde{x}) + \frac{1}{2} (x - \tilde{x})^T \Lambda (x - \tilde{x}) \quad \forall x, \tilde{x} \in \Omega
$$

$$
=: Q_{\mathcal{F}}(x, \tilde{x}),
$$

where $Q_{\mathcal{F}}$ is a surrogate function of $\mathcal{F}$.

This is a classical result based on the second-order Taylor polynomial and will not be proven here.

If the matrix $\Lambda$ is additionally symmetric and positive definite, it can be shown [19] that the update rule for $x$ according to (13) via the zero gradient condition $\nabla \mathcal{F}(\tilde{x}) = 0$ gives the unique minimiser

$$
x_\ast = \tilde{x} - \Lambda^{-1} \nabla \mathcal{F}(\tilde{x}).
$$

In this work, we will only apply the QUBP for quadratic cost functions $\mathcal{F}$, whose Hessian is automatically a constant matrix. For these functions, typical choices of $\Lambda$ are diagonal matrices of the form

$$
(16) \quad \Lambda(\tilde{x})_{ii} := \frac{(\nabla^2 f(\tilde{x}) \tilde{x})_i + \kappa_i}{\tilde{x}_i},
$$

which are dependent on the second argument of the corresponding surrogate $Q_{\mathcal{F}}(x, \tilde{x})$. The parameters $\kappa_i \geq 0$, are constants and have to be chosen depending on the considered penalty terms of the NMF cost function.

The diagonal structure of $\Lambda(\tilde{x})$ ensures its simple invertibility, the separability of the corresponding surrogate and the desired multiplicative algorithms based on (13). Hence, the update rule in (15) can be viewed as a gradient descent approach with a suitable stepsize defined by the diagonal matrix $\Lambda$.

**Appendix B. Derivation of the algorithms.** In this section, we derive the multiplicative update rules for the NMF minimisation problems in BC-X and BC.

**B.1. Model BC-X.**

**B.1.1. Algorithm for X.** We start first of all with the NMF model BC-X and the minimisation with respect to $X$. The cost function of the NMF problem in BC-X for the minimisation with respect to $X$ reduces to

$$
(17) \quad \mathcal{F}(X) := \sum_{t=1}^{T} \left[ \frac{1}{2} \|A_t X_{\ast, t} - Y_{\ast, t}\|^2_F + \frac{\mu_X}{2} \|X\|_F + \lambda_X \|X\|_1 + \frac{\alpha}{2} \|X - BC\|_F^2 \right] =: \mathcal{F}_1(X) + \mathcal{F}_2(X)
$$

by neglecting the constant terms. To apply the QUBP and to avoid fourth-order tensors during the computation of the Hessians, we use the separability of $\mathcal{F}_1$ of the NMF with respect to the columns of $X$, i.e. it can be written as sum, where each term depends only on the respective column $X_{\ast, t}$. Hence, we write

$$
\mathcal{F}_1(X) = \sum_{t=1}^{T} \left[ \frac{1}{2} \|A_t X_{\ast, t} - Y_{\ast, t}\|^2_F + \frac{\mu_X}{2} \|X_{\ast, t}\|^2_F + \lambda_X \|X_{\ast, t}\|_1 \right] := \sum_{t=1}^{T} f_1(X_{\ast, t}).
$$
We can assume that $X$ contains only strictly positive entries due to the strict positive initialisations of the multiplicative algorithms. Hence, the functions $f_t$ are twice continuously differentiable despite the occurring $\ell_1$ regularisation term. The computations of the gradient and the Hessian of $f_t$ are straightforward and we obtain

$$\nabla f_t(X_{\bullet,t}) = A_t^T A_t X_{\bullet,t} - A_t^T Y_{\bullet,t} + \mu_X X_{\bullet,t} + \lambda_X I_{N \times 1},$$

$$\nabla^2 f_t(X_{\bullet,t}) = A_t^T A_t + \mu_X I_{N \times N},$$

where $I_{N \times N}$ is the $N \times N$ identity matrix. Choosing $\kappa_n = \lambda_X$ for all $n$ in (16), we define the surrogate $Q_{f_t}$ according to Lemma A.2. It is then easy to see, that

$$Q_{f_t}(X, \tilde{X}) := \sum_{t=1}^{T} Q_{f_t}(X_{\bullet,t}, \tilde{X}_{\bullet,t})$$

defines a separable and convex surrogate function for $f$. For $Q_{f_2}$, we set simply $Q_{f_2}(X, \tilde{X}) := \alpha/2 \|X - BC\|^2_F$, such that we end up with

$$Q_{f}(X, A) := Q_{f_1}(X, A) + Q_{f_2}(X, A)$$

as a suitable surrogate for $F$. Based on the update rule in (13), we consider the zero gradient condition $\nabla_X Q_{f}(X, \tilde{X}) = 0$ and compute

$$\frac{\partial Q_{f}}{\partial X_{nt}}(X, \tilde{X}) = \frac{\partial f_t}{\partial X_{nt}}(X_{\bullet,t}) + \left( \Lambda(X_{\bullet,t})(X_{\bullet,t} - \tilde{X}_{\bullet,t}) \right)_n + \frac{\alpha}{2} \frac{\partial}{\partial X_{nt}} \|X - BC\|^2_F$$

$$= \left( A_t^T A_t \tilde{X}_{\bullet,t} \right)_n - (A_t^T Y_{\bullet,t})_n + \mu_X \tilde{X}_{nt} + \lambda_X$$

$$+ \left( \frac{(A_t^T A_t + \mu_X I_{N \times N}) \tilde{X}_{\bullet,t}}{X_{nt}} \right)_n \lambda_X$$

$$= - (A_t^T Y_{\bullet,t})_n + X_{nt} \left( \frac{(A_t^T A_t \tilde{X}_{\bullet,t})}{\tilde{X}_{nt}} + \mu_X \tilde{X}_{nt} + \lambda_X \right) + \alpha(X_{nt} - (BC)_{nt})$$

$$= 0.$$

Rearranging the equation leads to

$$X_{nt} = \frac{(A_t^T Y_{\bullet,t})_n + \alpha(BC)_{nt}}{(A_t^T A_t \tilde{X}_{\bullet,t})_n + \mu_X \tilde{X}_{nt} + \lambda_X}.$$ 

We therefore have

$$X_{\bullet,t} = \tilde{X}_{\bullet,t} \circ \frac{A_t^T Y_{\bullet,t} + \alpha BC_{\bullet,t}}{A_t^T A_t \tilde{X}_{\bullet,t} + (\mu_X + \alpha) X_{\bullet,t} + \lambda_X I_{N \times 1}},$$

which yields the multiplicative update rule

$$X_{\bullet,t} \leftarrow X_{\bullet,t} \circ \frac{A_t^T Y_{\bullet,t} + \alpha BC_{\bullet,t}}{A_t^T A_t X_{\bullet,t} + (\mu_X + \alpha) X_{\bullet,t} + \lambda_X I_{N \times 1}}$$

based on (13). Note that the correct choice of the matrix $\Lambda$ together with the $\kappa_i$ is crucial to ensure the multiplicative structure of the algorithm.
Algorithm for $B$. The minimisation with respect to $B$ reduces the cost function in $\mathbf{BC-X}$ to
\[
F(B) := \frac{\alpha}{2} \|BC - X\|_F^2 + \frac{\mu_B}{2} \|B\|_2^2 + \lambda_B \|B\|_1 + \frac{\tau}{2} TV(B)
\]
and involves the TV regularisation on $B$ of the NMF model. Analogously to the previous section, we use the separability of $F_1$ and write
\[
F_1(B) = \sum_{n=1}^N \left[ \frac{\alpha}{2} \|X_{n, \bullet} - B_{n, \bullet} C\|_F^2 + \frac{\mu_B}{2} \|B_{n, \bullet}\|_2^2 + \lambda_B \|B_{n, \bullet}\|_1 \right] = \sum_{n=1}^N f_n(B_{n, \bullet}).
\]
By computing the gradients
\[
\nabla f_n(B_{n, \bullet}) = \alpha(B_{n, \bullet} C - X_{n, \bullet}) C^T + \mu_B B_{n, \bullet} + \lambda_B I_{1 \times K}
\]
and choosing $\kappa_k = \lambda_B$ in (16), we define analogously the surrogates $Q_{f_n}$, which leads to the convex surrogate
\[
Q_{F_1}(B, \tilde{B}) := \sum_{n=1}^N Q_{f_n}(B_{n, \bullet}, \tilde{B}_{n, \bullet})
\]
for $F_1$. The derivation of a suitable surrogate for the TV regularisation term $F_2$ is based on an approach different from the QUBP and shall not be discussed in detail. We just state the result and refer the reader for details to [41, 14, 19]. A convex and separable surrogate function for $F_2$ is given by
\[
Q_{F_2}(B, \tilde{B}) = \frac{\tau}{2} \sum_{k=1}^K \sum_{n=1}^N \left[ P(\tilde{B})_{nk} (B_{nk} - Z(\tilde{B})_{nk})^2 \right] + G(\tilde{B}),
\]
with the matrices $P(\tilde{B}), Z(\tilde{B}) \in \mathbb{R}_{>0}^{N \times K}$ defined in (9) and (10) and a function $G$ depending only on the matrix $\tilde{B}$. Hence, we finally end up with $Q_F(B, \tilde{B}) := Q_{F_1}(B, \tilde{B}) + Q_{F_2}(B, \tilde{B})$ as a suitable surrogate for $F$.

Similar to the computations in the previous paragraph, the zero gradient condition yields then
\[
\frac{\partial Q_F}{\partial B_{nk}}(B, \tilde{B}) = -\alpha(XCT)_{nk} + B_{nk} \frac{\alpha(\tilde{B}CCT)_{nk} + \mu_B \tilde{B}_{nk} + \lambda_B \tau P(\tilde{B})_{nk} (B_{nk} - Z(\tilde{B})_{nk})}{B_{nk}} = 0
\]
and therefore
\[
B_{nk} = \tilde{B}_{nk} \frac{\alpha(XCT)_{nk} + \tau P(\tilde{B})_{nk} Z(\tilde{B})_{nk}}{\alpha(\tilde{B}CCT)_{nk} + \mu_B \tilde{B}_{nk} + \lambda_B + \tau P(\tilde{B})_{nk} B_{nk}}.
\]
Hence, we have the update rule
\[
B \leftarrow B \odot \frac{\alpha XCT + \tau P(\tilde{B}) \odot Z(\tilde{B})}{\alpha BCC^T + \mu B + \lambda_B I_{N \times K} + \tau P(\tilde{B}) \odot \tilde{B}}.
\]

Algorithm for $C$. The optimisation with respect to the matrix $C$ can be tackled analogously with the QUBP and will not be described in detail. In this case, the cost function can be reduced to well-known regularised NMF problems [13], which leads to the update rule
\[
C \leftarrow C \odot \frac{\alpha B^T X}{\alpha B^T BC + \mu C + \lambda C I_{K \times T}}.
\]
B.2. Model BC. In this section, we discuss the computation of the optimisation algorithms for the NMF model BC.

B.2.1. Algorithm for B. In this case, the cost function reduces to

\[
\mathcal{F}(B) := \sum_{t=1}^{T} \frac{1}{2} ||A_t(BC)_{n,k} - Y_{n,k}||^2 + \frac{\mu_B}{2} ||B||_F^2 + \lambda_B ||B||_1 + \frac{\tau}{2} \text{TV}(B).
\]

Analogously to the previous cases, we analyze the functions \(\mathcal{F}_1\) and \(\mathcal{F}_2\) separately. The difference is here, that \(\mathcal{F}_1\) is not separable with respect to the rows of \(B\) due to the discrepancy term and therefore, it is necessary to compute the gradient and the Hessian of the whole function \(\mathcal{F}_1\). Hence, the gradient \(\nabla \mathcal{F}_1(B)\) is an \(N \times K\) matrix and the Hessian \(\nabla^2 \mathcal{F}_1(B)\) a fourth-order tensor, which are given by their entries

\[
\nabla \mathcal{F}_1(B)_{n,k} = \sum_{t=1}^{T} C_{kt} (A^t_{BC})_{n,k} - \sum_{t=1}^{T} C_{kt} (A^t Y_{n,k}) + \mu_B B_{n,k} + \lambda_B,
\]

\[
\nabla^2 \mathcal{F}_1(B)_{(n,k),(\hat{n},\hat{k})} = \sum_{t=1}^{T} C_{kt} C_{k\hat{t}} (A^t_{BC})_{n,k\hat{n},\hat{k}} + \mu_B \delta_{(n,k),(\hat{n},\hat{k})},
\]

where \(\delta_{(n,k),(\hat{n},\hat{k})} = 1\) if and only if \((n,k) = (\hat{n},\hat{k})\). The natural expansion of the quadratic upper bound principle given in Lemma A.2 is the ansatz function

\[
Q_{\mathcal{F}_1}(B, \tilde{B}) := \mathcal{F}_1(\tilde{B}) + \langle B - \tilde{B}, \nabla \mathcal{F}_1(\tilde{B}) \rangle_F + \frac{1}{2} \sum_{(n,k)} \sum_{(\hat{n},\hat{k})} (B - \tilde{B})_{n,k} \Lambda(\tilde{B})_{(n,k),(\hat{n},\hat{k})}(B - \tilde{B})_{\hat{n}\hat{k}}
\]

with the fourth order tensor

\[
\Lambda(\tilde{B})_{(n,k),(\hat{n},\hat{k})} := \begin{cases} \sum_{(i,j)} \nabla^2 \mathcal{F}_1(\tilde{B})_{(n,k),(i,j)} \tilde{B}_{ij} + \lambda_B & \text{for } (n,k) = (\hat{n},\hat{k}), \\ 0 & \text{for } (n,k) \neq (\hat{n},\hat{k}), \end{cases}
\]

where \((\cdot,\cdot)_F\) denotes the Frobenius inner product.

Taking the same surrogate \(Q_{\mathcal{F}_2}\) for the TV penalty term as in (19), we end up with the surrogate function

\[
Q_{\mathcal{F}}(B, \tilde{B}) := Q_{\mathcal{F}_1}(B, \tilde{B}) + Q_{\mathcal{F}_2}(B, \tilde{B})
\]

for \(\mathcal{F}\). Its partial derivative with respect to \(B_{n,k}\) is given by

\[
\frac{\partial Q_{\mathcal{F}}}{\partial B_{n,k}}(B) = - \sum_{t=1}^{T} C_{kt} (A^t_{BC})_{n,k} + B_{n,k} \sum_{t=1}^{T} C_{kt} (A^t_{BC})_{n,k} + \mu_B \tilde{B}_{n,k} + \lambda_B + \tau P(\tilde{B})_{n,k} (B_{n,k} - Z(\tilde{B})_{n,k}).
\]

The zero-gradient condition gives then the equation

\[
B_{n,k} = \tilde{B}_{n,k} + \frac{\sum_{t=1}^{T} C_{kt} (A^t_{BC})_{n,k} + \tau P(\tilde{B})_{n,k} Z(\tilde{B})_{n,k}}{\sum_{t=1}^{T} C_{kt} (A^t_{BC})_{n,k} + \mu_B \tilde{B}_{n,k} + \lambda_B + \tau P(\tilde{B})_{n,k}}.
\]
which can be extended to the whole matrix $B$. Therefore, based on (13), we have the update rule

$$B \leftarrow B \circ \left( \frac{\sum_{t=1}^{T} A_t^T Y_{t, \cdot} (C^t)_{\cdot, t} + \tau P(B) \circ Z(B)}{\sum_{t=1}^{T} A_t^T A_t (BC)_{\cdot, t} \cdot (C^t)_{\cdot, t} + \mu_B B + \lambda_B I_{N \times K} + \tau B \circ P(B)} \right).$$

Figure 22. Plots of $\|B_{k, \cdot} C_{k, \cdot}\|_{\infty}$ with $K = 10$ in descending order for the dynamic Shepp-Logan phantom (Figure 22a) and the vessel phantom (Figure 22b) considered in Section 3.1.1 and 3.1.2 with 1% Gaussian noise and the parameters given in Table 3 and 4. In the case of $\text{gradTV}_{\text{PCA}}$, the ten leading features with respect to the singular values are considered.

Figure 23. Plots of $\|B_{k, \cdot} C_{k, \cdot}\|_{\infty}$ with $K = 10$ in descending order for the dynamic Shepp-Logan phantom considered in Section 3.1.3 with 1% Gaussian noise and the parameters given in Table 5. In the case of $\text{gradTV}_{\text{PCA}}$, the ten leading features with respect to the singular values are considered.
B.2.2. Algorithm for C. In this case, the cost function is separable with respect to the columns of $C$, such that

$$\mathcal{F}(C) := \sum_{t=1}^{T} \frac{1}{2} \| A_t BC_{\bullet,t} - Y_{\bullet,t} \|^2 + \frac{\mu_C}{2} \| C_{\bullet,t} \|^2 + \lambda_C \| C_{\bullet,t} \|_1 =: \sum_{t=1}^{T} f_t(C_{\bullet,t}).$$

Hence, we can split the minimisation into the columns of $C$ to use the standard QUBP without considering higher order tensors. We compute

$$\nabla f_t(C_{\bullet,t}) = B^T A^T_t A_t (BC)_{\bullet,t} - B^T A^T_t Y_{\bullet,t} + \mu_C C_{\bullet,t} + \lambda_C I_{K \times 1},$$

$$\nabla^2 f_t(C_{\bullet,t}) = B^T A^T_t A_t B + \mu_C I_{K \times K}.$$ 

By choosing $\kappa_k = \lambda_C$ for all $k$ in (16), we define $Q_{f_t}(C_{\bullet,t}, \tilde{C}_{\bullet,t})$ as a surrogate function for $f_t$ according to Lemma A.2. The update rule in (15) gives then

$$C_{\bullet,t} \mapsto C_{\bullet,t} - \Lambda^{-1}(\tilde{C}_{\bullet,t}) \nabla f_t(\tilde{C}_{\bullet,t}),$$

which leads to

$$C_{\bullet,t} \leftarrow C_{\bullet,t} \odot \frac{B^T A^T_t Y_{\bullet,t}}{B^T A^T_t A_t (BC)_{\bullet,t} + \mu_C C_{\bullet,t} + \lambda_C I_{K \times 1}}.$$ 

Appendix C. Parameter choice.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>BC</th>
<th>BC-X</th>
<th>gradTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>1% noise</td>
<td>3% noise</td>
<td>1% noise</td>
</tr>
<tr>
<td>$\mu_C$</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>$\tau$</td>
<td>10</td>
<td>50</td>
<td>6</td>
</tr>
<tr>
<td>$\rho_{\text{grad}}$</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\rho_{\text{tv}}$</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\rho_{\text{tv}}$</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\rho_{\text{C}}$</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3. Parameter choice of the experiments in Section 3.1.1 for the dynamic Shepp-Logan phantom for 1% and 3% Gaussian noise.

C.1. Number of features. To determine an optimal number of features $K$ for the numerical part of this work, we perform experiments with $K = 10$ for both datasets as described in Section 3.1. Figure 22 shows the plots of the values $\| B_{\bullet,k} C_{\bullet} \|_\infty$ for $k = 1, \ldots, K$ in descending order for BC, BC-X, gradTV,PCA and gradTV, NMF.

As expected, three main features can be extracted in the case of the dynamic Shepp-Logan phantom, which is clearly confirmed by the plot in Figure 22a for all methods. However, for BC-X, several features for $k \geq 4$ still have a slight contribution to the overall reconstruction. Together with further experiments for $K \geq 4$ leading to more stable and better reconstruction qualities, we choose in the case of the dynamic Shepp-Logan phantom for the numerical experiments in Section 3.1.1 $K = 5$ for all methods using the NMF as a feature extraction step.

Similar results are obtained in Figure 22b for the vessel phantom with its two main spatial and temporal features. In this case, we choose $K = 4$ for all considered methods.
Table 4. Parameter choice of the experiments in Section 3.1.2 for the vessel phantom for 1% and 3% Gaussian noise.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>BC</th>
<th>BC-X</th>
<th>gradTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>1% noise</td>
<td>3% noise</td>
<td></td>
<td></td>
</tr>
<tr>
<td>α</td>
<td>–</td>
<td>–</td>
<td>3·10^2</td>
</tr>
<tr>
<td>µC</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>τ</td>
<td>1.3·10^2</td>
<td>4.3·10^2</td>
<td>90</td>
</tr>
<tr>
<td>ρgrad</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ρthr</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>ρTV</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>˜µC</td>
<td>–</td>
<td>–</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Table 5. Parameter choice of the experiments in Section 3.1.3 for the dynamic Shepp-Logan phantom for 1% Gaussian noise.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>BC</th>
<th>BC-X</th>
<th>gradTV</th>
</tr>
</thead>
<tbody>
<tr>
<td>α</td>
<td>–</td>
<td>–</td>
<td>70</td>
</tr>
<tr>
<td>µC</td>
<td>0.1</td>
<td>0.1</td>
<td>–</td>
</tr>
<tr>
<td>τ</td>
<td>10</td>
<td>10</td>
<td>–</td>
</tr>
<tr>
<td>ρgrad</td>
<td>–</td>
<td>–</td>
<td>1·10^{-3}</td>
</tr>
<tr>
<td>ρthr</td>
<td>–</td>
<td>–</td>
<td>2·10^{-4}</td>
</tr>
<tr>
<td>ρTV</td>
<td>–</td>
<td>–</td>
<td>1·10^{-2}</td>
</tr>
<tr>
<td>˜µC</td>
<td>–</td>
<td>–</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Finally, Figure 23 shows the plots of $\|B_k C_k\|_{\infty}$ for the dynamic Shepp-Logan phantom, which is considered in Section 3.1.3 and illustrated in Figure 16. Based on this and further experiments, we choose $K = 4$ for BC and gradTV,NMF as well as $K = 3$ for BC-X.

C.2. Further parameters.

REFERENCES


Spatially Coherent Clustering Based on Orthogonal Nonnegative Matrix Factorization

Pascal Fernsel

Center for Industrial Mathematics, University of Bremen, 28359 Bremen, Germany; p.fernsel@uni-bremen.de; Tel.: +49-(0)-421-218-63814

Abstract: Classical approaches in cluster analysis are typically based on a feature space analysis. However, many applications lead to datasets with additional spatial information and a ground truth with spatially coherent classes, which will not necessarily be reconstructed well by standard clustering methods. Motivated by applications in hyperspectral imaging, we introduce in this work clustering models based on Orthogonal Nonnegative Matrix Factorization (ONMF), which include an additional Total Variation (TV) regularization procedure on the cluster membership matrix to enforce the needed spatial coherence in the clusters. We propose several approaches with different optimization techniques, where the TV regularization is either performed as a subsequent post-processing step or included into the clustering algorithm. Finally, we provide a numerical evaluation of 12 different TV regularized ONMF methods on a hyperspectral dataset obtained from a matrix-assisted laser desorption/ionization imaging measurement, which leads to significantly better clustering results compared to classical clustering models.

Keywords: orthogonal nonnegative matrix factorization; clustering; spatial coherence; hyperspectral data; MALDI imaging

1. Introduction

Cluster analysis has been studied over the past fifty years in the machine learning community and is one of the central topics in unsupervised learning with a wide range of possible research directions and application fields, including image segmentation, document clustering, and bioinformatics [1]. The general clustering problem is to partition a given set of objects \( O \) into different groups, such that the objects within a group are more similar to each other compared to the objects in other groups. One typical approach is based on a feature space analysis. The basic concept is to assign to each object \( \sigma \in O \) a feature vector \( x_\sigma \in X \) containing the characteristics of \( \sigma \), where \( X \) is a suitably defined feature space. Furthermore, a similarity measure and a suitable minimization problem is defined to introduce the notion of similarity between the feature vectors and to formulate the clustering problem.

However, many types of datasets contain additional spatial information, which is typically not used in a classical cluster analysis. Characteristic examples are images or, more generally, hyperspectral datasets, where each measured spectrum is associated to a point in a two- or three-dimensional space. Furthermore, many application fields, such as mass spectrometry imaging or Earth remote sensing, naturally lead to datasets with spatially coherent regions. Hence, a classical cluster analysis, which is entirely based on a feature space analysis, does not lead necessarily to spatially coherent clusters and is, therefore, not sufficient to reconstruct the coherent regions in these kind of data.

Hence, this work focuses on a combined clustering analysis, which takes into account both the feature space and the spatial coherence of the clusters. We introduce numerous clustering methods based on ONMF for general nonnegative datasets with spatial information and include a TV regularization procedure to regularize the cluster membership.
matrix to induce the needed spatial coherence. Furthermore, we discuss different optimization techniques for the ONMF models and derive the corresponding minimization algorithms. Finally, we perform a numerical evaluation on a mass spectrometry imaging dataset acquired from a matrix-assisted laser desorption/ionization imaging measurement of a human colon tissue sample and compare the proposed clustering methods to classical ONMF approaches.

This paper is organized as follows. After a short description of the related work and the used notation in Sections 1.1 and 1.2, we give a brief outline of the basics of ONMF approaches, its relations to K-means clustering, and details on possible solution algorithms in Section 2. In Section 3, we introduce the proposed methods in this work, which are divided into so-called separated methods and combined methods. Section 4 is entirely devoted to the numerical experiments and the evaluation of the discussed methods. Finally, Section 5 concludes the findings and gives an outlook for future possible research directions.

1.1. Related Work

The natural relation between ONMF and clustering models is well studied. One of the first theoretical analysis was provided by Ding et al. in Reference [2]. By comparing the cost functions of different Nonnegative Matrix Factorization (NMF) and K-means models, the authors could show in their work, for example, the strong relationship between K-means clustering and ONMF with an orthogonality constraint on one of the factorization matrices, as well as kernel K-means and symmetric ONMF. Furthermore, the connections to spectral clustering and tri-factorizations were studied. Several works followed with a similar theoretical emphasis on tri-factorizations [3] and multiple other NMF models [4].

The previous mentioned works focus on the theoretical side but also give some first update rules to solve the corresponding ONMF problems. However, more work has been done for the algorithm development. Many classical approaches are based on multiplicative update rules [3,5–7]. More recent works are, for instance, based on nuclear norm optimization [8], further multiplicative update schemes [9,10], hierarchical alternating least squares [11–13], and proximal alternating linearized minimization [14], together with the very recent work of Reference [15], EM, such as algorithms and augmented Lagrangian approaches [16], deep neural networks [17], and other techniques [18,19]. Very recently, Reference [20] developed a block coordinate descent-based projected gradient algorithm specifically for solving ONMF problems. Regarding nonnegative matrix tri-factorizations, the very recent work of Reference [21] introduces a block inertial Bregman proximal algorithm for general nonsmooth and non-convex problems and applies it to the application case of symmetric nonnegative matrix tri-factorization. Furthermore, the authors in Reference [22] develop four algorithms for the specific problem case of symmetric multi-type nonnegative matrix tri-factorization comprising a fixed-point method, block-coordinate descent with projected gradient, and a gradient method with exact line search, as well as an adaptive moment estimation approach. Finally, we would like to refer the interested reader at this point to the two review articles on NMF methods for clustering by Reference [23,24] and a book on NMF by Reference [25].

While analyzing and developing optimization algorithms for NMF clustering methods is a major topic throughout the literature, studying clustering techniques with spatial coherence by incorporating the local information of the considered data points is far less common. This topic is primarily analyzed in the context of image segmentation problems [26–29]. The subject of spatial coherence can also be found in the literature of hyperspectral image analysis. Several NMF models with total variation regularization, which include the local neighborhood information of each data point, have been analyzed for the critical processing step of hyperspectral unmixing [30–32]. These articles can be considered as closest to our approach, since we also focus on the application of generalized NMF models to hyperspectral images. Further works also consider NMF models with different TV penalty terms for hyperspectral image denoising [33–35], some of which will be used in the later course of this work for the derivation of optimization algorithms of the respective clus-
tering models. Reference [36] uses a variant of the NMF, namely the nonnegative matrix underapproximation, which solves the NMF by identifying localized and sparse features sequentially. Similar to the considered approach in this work, the authors introduce a sparsity constraint, make use of the fact that it is more likely that neighboring pixels are contained in the same features, and apply the approach to hyperspectral datasets.

However, all the aforementioned works on ONMF or TV regularized NMF only include either orthogonality constraints or TV regularization into their NMF models, whereas we focus on combining both of these properties to obtain a spatially coherent clustering method for hyperspectral datasets.

The only work, which includes TV regularization, as well as penalty terms to enforce an orthogonality constraint on one of the matrices, is, to the best of our knowledge, the survey article of Reference [37], with a rather general focus on the development of algorithms based on surrogate functions leading to multiplicative update schemes.

1.2. Notation

Matrices will play a major role throughout this work and are denoted, unless otherwise stated, by capital Latin or Greek letters (e.g., $X, U, \Psi, \ldots$). The entry of a matrix $U$ in the $i$-th row and $j$-th column is indicated as $U_{ij}$. The same holds true for a matrix product, where its $ij$-th entry is given by $(UV)_{ij}$. Furthermore, we use a dot to indicate rows and columns of matrices. The $i$-th row and $j$-th column of $U$ are written as $U_{i, \bullet}$ and $U_{\bullet, j}$, respectively. Moreover, we denote the $i$-th iteration of a matrix $U$ in an algorithm by $U[i]$.

Moreover, we write $\|U\|_F$ and $\|U_{\bullet, j}\|_2$ for the Frobenius norm of a matrix $U$ and the usual Euclidean norm of a vector $U_{\bullet, j}$. We also use the notion of nonnegative matrices and write $U \geq 0$ or $U \in \mathbb{R}^{m \times n}_{\geq 0}$ with $\mathbb{R}_{\geq 0} := \{ x \in \mathbb{R} \mid x \geq 0 \}$ for an $m \times n$ matrix $U$, which has only nonnegative entries. The notation for the dimension of the matrices in the NMF problems are reused throughout the article and will be introduced in the following, Section 2.

2. Background

2.1. Orthogonal NMF and K-Means

Nonnegative Matrix Factorization (NMF), originally introduced by Paatero and Tapper in 1994 as positive matrix factorization [38], is a specific matrix factorization method designed to obtain a low-rank approximation of a given and typically large nonnegative data matrix. Different from the widely used Principal Component Analysis (PCA), which is based on the singular value decomposition and allows for computation of a best rank $K$ approximation of a given arbitrary matrix, the NMF constrains the matrix factors to be nonnegative. This property makes the NMF the method of choice where the considered data naturally fulfills a nonnegativity constraint so that the interpretability of the factor matrices is ensured. NMF has been widely used for data compression, source separation, feature extraction, clustering, or even for solving inverse problems. Possible application fields are hyperspectral unmixing [30–32], document clustering [8,39], and music analysis [40] but also medical imaging problems, such as dynamic computed tomography, to perform a joint reconstruction and low-rank decomposition of the corresponding dynamic inverse problem [41], or Matrix-Assisted Laser Desorption/Ionization (MALDI) imaging, where it can be used for tumor typing in the field of bioinformatics as a supervised classification method [42].

Mathematically, the standard NMF problem can be formulated as follows: For a given nonnegative matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, the task is to find two nonnegative matrices $U \in \mathbb{R}^{M \times K}_{\geq 0}$ and $V \in \mathbb{R}^{K \times N}_{\geq 0}$ with $K \ll \min\{M, N\}$, such that

$$X \approx UV = \sum_{k=1}^{K} U_{\bullet, k} V_{k, \bullet}.$$
This allows the approximation of the columns $X_{\bullet,n}$ and rows $X_{m,\bullet}$ via a superposition of just a few basis vectors $\{U_{\bullet,k}\}_k$ and $\{V_{k,\bullet}\}_k$, such that $X_{\bullet,n} \approx \sum_k V_{k,n} U_{\bullet,k}$ and $X_{m,\bullet} \approx \sum_k U_{m,k} V_{k,\bullet}$. In this way, the NMF can be seen as a basis learning tool with additional nonnegativity constraints.

The typical variational approach to tackle the NMF problem is to reformulate it as a minimization problem by defining a suitable discrepancy term $D$ according to the noise assumption of the underlying problem. The default case of Gaussian noise corresponds to the Frobenius norm on which we will focus on in this work. Further possible choices include the Kullback–Leibler divergence or more generalized divergence functions [25].

Moreover, NMF problems are non-linear and ill-conditioned [43,44]. Thus, they require stabilization techniques, which is typically done by adding regularization terms $R_j$ into the NMF cost function $F$. However, besides the use case of regularization, the penalty terms $R_j$ can also be used to enforce additional properties to the factorization matrices $U$ and $V$. The general NMF minimization problem can, therefore, be written as

$$\min_{U,V \geq 0} D(X, UV) + \sum_{j=1}^J \alpha_j R_j(U,V) = \min_{U,V \geq 0} F(U,V),$$

where $\alpha_j$ are regularization parameters. Common choices for $R_j$ are $\ell_1$ and $\ell_2$ penalty terms. Further possible options are total variation regularization or other penalty terms which enforce orthogonality or even allow a supervised classification scheme in case the NMF is used as a prior feature extraction step [37,42]. In this work, we will focus on the combination of orthogonality constraints and total variation penalty terms to construct an NMF model for spatially coherent clustering methods.

Another essential step for computing the NMF of a given matrix $X$ is the determination of an optimal number of features $K$. Typical techniques used throughout the literature are based on heuristic or approximative methods, including core consistency diagnostics via a PCA or residual analysis (also see Reference [25]). A straightforward technique used in Reference [45] is based on the analysis of the rank-one matrices $U_{\bullet,k} V_{k,\bullet}$. For a $K \in \mathbb{N}$ chosen sufficiently large, the considered NMF algorithm is executed to obtain the factorization $UV$. Afterwards, the norm of the rank-one matrices $U_{\bullet,k} V_{k,\bullet}$ for every $k \in \{1,\ldots,K\}$ is analyzed. By the choice of a large $K$, the NMF algorithm is forced to compute additional irrelevant features, which can be identified by a small norm of the corresponding rank-one matrices. By choosing a suitably defined threshold for the values of the norm, a suitable $K$ can be obtained.

This work, however, will not focus on methods to determine an optimal number of features. Hence, we assume in the numerical part in Section 4 that the true number of features in the considered dataset is known in advance so that $K$ can be set a priori.

On the other hand, $K$-means clustering is one of the most commonly used prototype-based, partitional clustering technique. As for any other clustering method, the main task is to partition a given set of objects into groups such that objects in the same group are more similar compared to the ones in other groups. These groups are usually referred to as clusters. In mathematical terms, the aim is to partition the index set $\{1,2,\ldots,M\}$ of a corresponding given dataset $\{x_m \in \mathbb{R}^N | m = 1,\ldots,M\}$ into disjoint sets $I_k \subset \{1,\ldots,M\}$, such that $\cup_{k=1}^K I_k = \{1,\ldots,M\}$.

Many different variations and generalizations of $K$-means have been proposed and analyzed (see, for instance, Reference [1,46] and the references therein), but we will focus in this section on the most common case. The method is based on two main ingredients. On the one hand, a similarity measure $\text{dist}(\cdot,\cdot)$ is needed to specify the similarity between data points. The default choice is the squared Euclidean distance $\text{dist}(x_i, x_j) := \|x_i - x_j\|_2^2$. On the other hand, so-called representative centroids $c_k \in \mathbb{R}^N$ are computed for each cluster $I_k$. The computation of the clusters and centroids is based on the minimization of the within-cluster variances given by $J = \sum_{k=1}^K \sum_{m \in I_k} \text{dist}(x_m, c_k)$. Due to the NP-hardness of the minimization problem [47], heuristic approaches are commonly used to
find an approximate solution. The K-means algorithm is the most common optimization technique which is based on an alternating minimization. After a suitable initialization, the first step is to assign each data point \( x_m \) to the cluster with the closest centroid with respect to the distance measure \( \text{dist}(\cdot, \cdot) \). In the case of the squared Euclidean distance, the centroids are recalculated in a second step based on the mean of its newly assigned data points to minimize the sum of the within-cluster variances. Both steps are repeated until the assignments do not change anymore.

The relationship between the NMF and K-means clustering can be easily seen by adding further constraints to both problems. First of all, from the point of view of K-means, we assume nonnegativity of the given data and write the vectors \( x_m \) row-wise to a data matrix \( X \) such that \( X = [x_1, \ldots, x_M]^\top \in \mathbb{R}^{M \times N} \). Furthermore, we define the so-called cluster membership matrix \( \hat{U} \in \{0, 1\}^{M \times K} \), such that

\[
\hat{U}_{mk} := \begin{cases} 
0 & \text{if } m \notin I_k \\
1 & \text{if } m \in I_k 
\end{cases}
\]

and the centroid matrix \( \hat{V} := [c_1, \ldots, c_K]^\top \in \mathbb{R}^{K \times N} \). With this, and by choosing the squared Euclidean distance function, it can be easily shown that the objective function \( J \) of K-means can be rewritten as \( J = \|X - \hat{U}\hat{V}\|_2^2 \), which has the same structure of the usual cost function of an NMF problem. However, the usual NMF does not constrain one of the matrices to have binary entries or, more importantly, to be row-orthogonal as it is the case for \( \hat{U} \). This ensures that each row of \( \hat{U} \) contains only one nonzero element which gives the needed clustering interpretability.

This gives rise to the problem of ONMF, which is given by

\[
\min_{U, V \geq 0} F(U, V), \quad \text{s.t. } U^\top U = I,
\]

where \( I \) is the identity matrix. The matrices \( U \) and \( V \) of an ONMF problem will be henceforth also referred to as cluster membership matrix and centroid matrix, respectively. In the case of the Frobenius norm as discrepancy term and without any regularization terms \( \mathcal{R}_j \), it can be shown that this problem is equivalent to weighted variant of the spherical K-means problem [16]. For further variants of the relations between ONMF and K-means, we refer to the works of Reference [2–4] and the review articles of Reference [23,24].

2.2. Algorithms for Orthogonal NMF

Due to the ill-posedness of NMF problems and possible constraints on the matrices, tailored minimization approaches are needed. In this section, we review shortly common optimization techniques of NMF and ONMF problems, which will also be used in this work for the derivation of algorithms for ONMF models, including spatial coherence.

For usual choices of \( \mathcal{D} \) and \( \mathcal{R}_j \) in the NMF problem (1), the corresponding cost function \( F \) is convex in each of the variables \( U \) and \( V \) but non-convex in \( \langle U, V \rangle \). Therefore, the majority of optimization algorithms for NMF and ONMF problems are based on alternating minimization schemes

\[
U_i^{[i+1]} = \arg \min_{U \geq 0} F(U, V_i^{[i]}), \\
V_i^{[i+1]} = \arg \min_{V \geq 0} F(U_i^{[i+1]}, V).
\]

One classical technique to tackle these minimization problems are alternating multiplicative algorithms, which only consist of summations and multiplications of matrices and, therefore, ensure the nonnegativity of \( U \) and \( V \) without any additional projection step provided that they are initialized appropriately. This approach was mainly popularized by the works of Lee and Seung [48,49], which also brought much attention to the NMF, in general. The update rules are usually derived by analyzing the Karush–Kuhn–Tucker
(KKT) first-order optimality conditions for each of the minimization problems in (2) and (3) or via the so-called Majorize-Minimization (MM) principle. The basic idea of the latter technique is to replace the NMF cost function \( F \) by a majorizing surrogate function \( Q_F : \text{dom}(F) \times \text{dom}(F) \to \mathbb{R} \), which is easier to minimize and whose tailored construction leads to the desired multiplicative updates rules defined by

\[
A^{[i+1]} := \arg \min_{A \in \text{dom}(F)} Q_F(A, A^{[i]}).
\]

With the defining properties of a surrogate function that \( Q_F \) majorizes \( F \) and \( Q_F(A, A) = F(A) \) for all \( A \in \text{dom}(F) \), it can be easily shown that the above update rule leads to a monotone decrease of the cost function \( F \) (also see Appendix A). However, the whole method is based on an appropriate construction of the surrogate functions, which is generally non-trivial. Possible techniques for common choices of \( D \) and \( R_j \) in the NMF cost function are based on the quadratic upper bound principle and Jensen’s inequality [37].

Overall, multiplicative algorithms offer a flexible approach to various choices of NMF cost functions and will also be used in this work for some of the proposed and comparative methods.

Another classical method is based on Alternating (nonnegative) Least Squares (ALS) algorithms. They are based on the estimation of the stationary points of the cost function with a corresponding fixed point approach and a subsequent projection step to ensure the nonnegativity of the matrices. An extension to this procedure is given by Hierarchical Alternating (nonnegative) Least Squares (HALS) algorithms, which solve nonnegative ALS problems column-wise for both matrices \( U \) and \( V \) [11,12] and will also be used as a comparative methodology.

An optimization approach, which was recently used for NMF problems, is the widely known Proximal Alternating Linearized Minimization (PALM) [14,15], together with its extensions, including stochastic gradients [50]. As a first step, the cost function is split up into a differentiable part \( F_1 \) and a non-differentiable part \( F_2 \). In its basic form, the PALM update rules consist of alternating gradient descent steps for \( U \) and \( V \) with learning rates based on the Lipschitz constants of the gradient of \( F_1 \) in combination with a subsequent computation of a proximal operator of the function \( F_2 \). Some of these techniques will be used for the proposed methods in this work and will be discussed in more detail in Section 3.

Further well-known techniques are, for example, projected gradient algorithms consisting of additive update rules, quasi-Newton approaches based on second-order derivatives of the cost function, or algorithms based on an augmented Lagrangian concept [16,25].

All these methods can be also used to derive suitable algorithms for ONMF problems. Common approaches to include the orthogonality constraints are the use of Lagrangian multipliers [3,5,6,11,12] or the replacement of the hard constraint \( U^TU = I \) by adding a suitable penalty term into the NMF cost function to enforce approximate row-orthogonality for \( U \) controlled by a regularization parameter [7,9,10,15,20,37]. Other methods include optimization algorithms on the Stiefel manifold [19], the use of sparsity and nuclear norm minimization [8,39], or other techniques [14,16,18].

In the next section, we will introduce the considered NMF models in this work and derive the corresponding optimization algorithms.

### 3. Orthogonal NMF with Spatial Coherence

In this section, we present the spatially coherent clustering methods based on ONMF models, together with the derivation of their optimization algorithms. Different from classical clustering approaches via ONMF, our proposed technique includes the local information of a measured data point into the clustering process. This is done by including a TV regularization procedure on the cluster membership matrix \( U \), which naturally leads to spatially coherent regions, while preserving their edges.
This is, for instance, especially helpful for hyperspectral datasets. If the neighborhood of a measured spectrum \( X_m \) is associated to one cluster \( I_k \), the inclusion of spatial coherence in the clustering model makes it more likely that \( X_m \) will be also classified to \( I_k \). This spatial coherence can be observed in many spectral imaging applications, such as Earth remote sensing or MALDI imaging, where locally close spectra have a higher probability to belong to the same class.

In the following, we divide our proposed techniques into so-called separated and combined methods.

### 3.1. Separated Methods

One straightforward approach to design a spatially coherent clustering method is to compute a clustering based on a classical ONMF model and subsequently perform a post-processing on the obtained cluster membership matrix based on total variation denoising. The general workflow is provided in Algorithm 1 and will be henceforth referred to as ONMF-TV.

**Algorithm 1 ONMF-TV**

1: Input: \( X \in \mathbb{R}^{M \times N}_\geq, K \in \mathbb{N}, \tau > 0, \theta \in \mathbb{R}^J \)
2: Initialize: \( U[0] \in \mathbb{R}^{M \times K}_\geq, V[0] \in \mathbb{R}^{K \times N}_\geq \)
3: \((U, V) \leftarrow \text{ONMF}_\theta(X, U[0], V[0]) \)
4: \( U \leftarrow \text{TVDENOISER}_\tau(U) \)

After a suitable initialization of the cluster membership matrix and the centroid matrix in Line 2, a classical ONMF model is used in Line 3 to perform a clustering of the given data \( X \), where \( \theta \) are possible hyperparameters which typically have to be chosen a priori. These ONMF models will be chosen based on some of the works described in Section 2.2 and will be specified in more detail in Section 4.2. Afterwards, a TV denoising algorithm is applied on the cluster membership matrix \( U \) obtained by ONMF\( _\theta \) to induce the spatial coherence in the clustering. In this work and for all considered separated methods, the denoising step is based on evaluating the proximal mapping column-wise on \( U \), which is defined by the minimization problem

\[
\text{prox}_{\tau \| \cdot \|_{TV}}(x) := \text{arg min}_{y \in \mathbb{R}^M} \left\{ \frac{1}{2} \| y - x \|^2 + \tau \| y \|_{TV} \right\}.
\] (4)

Here, \( \tau > 0 \) is a regularization parameter, and \( \| \cdot \|_{TV} \) denotes the classical isotropic TV \([51,52]\) and corresponds to the definition in (6) with \( \epsilon_{TV} = 0 \) if \( \| \cdot \|_{TV} \) is applied on a matrix \( U \). The algorithm used to solve the above minimization problem is based on a Fast Iterative Shrinkage/Thresholding Algorithm (FISTA) described in Reference [52], with a maximal iteration number of 100. Afterwards, every negative entry of \( U \) is projected to zero to ensure the corresponding nonnegativity constraint. For further details on the implementation of the TV denoising algorithm and the separated methods, in general, we refer the reader to the provided codes of all considered algorithms in our GitLab [53].

We consider this workflow as baseline for our comparison with the combined methods presented in the following Section 3.2. For the later numerical evaluation of the separated methods in Section 4.4, we will compare these approaches with and without the TV post-processing step to get an impression of the advantage of adding a TV regularization procedure to the clustering method.

The initialization methods, the stopping criteria and the choice of the hyperparameters will be specified in more detail in Section 4 of the numerical experiments of this work.

### 3.2. Combined Methods

In this section, we present the so-called combined methods, together with different optimization algorithms for their numerical solution. Different from the separated methods
in Section 3.1, this coupled approach includes a total variation penalty term into the ONMF model to induce the spatial coherence in the clustering. The combination of orthogonality constraints and a total variation penalty term in the NMF problem leads, in general, to a higher effort to derive suitable solution algorithms. However, the main motivation is that this joint workflow allows the clustering process to take advantage of the TV regularization, leading to better local minima, which could, therefore, enhance the quality of the clustering compared to classical approaches or the previous described separated methods, where the spatial coherence is just enforced in an independent subsequent TV denoising step.

In the following, we will present different multiplicative update rules and algorithms based on proximal gradient descent approaches.

As for the separated methods, the initialization and the stopping criteria, as well as the choice of the hyperparameters, for all considered approaches will be described in Section 4 in more detail.

3.2.1. Multiplicative Update Rules

Our first multiplicative algorithm is taken from the work of Reference [37] without any modification and is based on the ONMF model

\[
\min_{U,V \geq 0} \frac{1}{2} \|X - UV\|_F^2 + \frac{\sigma}{2} \|I - UTU\|_F^2 + \frac{\tau}{2} \|U\|_{TV^{\epsilon}}, \quad (5)
\]

where \(\sigma, \tau \geq 0\) are regularization parameters, and \(\| \cdot \|_{TV^{\epsilon}}\) is the smoothed, isotropic total variation [37,54] defined by

\[
\|U\|_{TV^{\epsilon}} := \sum_{k=1}^{K} \sum_{m=1}^{M} |\nabla_{mk} U| := \sum_{k=1}^{K} \sum_{m=1}^{M} \sqrt{\epsilon_{TV}^2 + \sum_{n \in N_m} (U_{mk} - U_{\tilde{m}k})^2}. \quad (6)
\]

Furthermore, \(\epsilon_{TV} > 0\) is a positive, small, predefined constant to ensure the differentiability of the TV penalty term, which is needed due to the MM principle for the optimization approach. Finally, \(N_m\) are index sets referring to spatially neighboring pixels. The default case in two dimensions for the neighborhood of a non-boundary pixel in \((i, j)\) is \(N_{(i,j)} = \{(i+1, j), (i, j+1)\}\) to obtain an estimate of the gradient components in both directions.

The derivation of the solution algorithm is described in Reference [37] and is based on the MM principle mentioned in Section 2.2. The surrogate function \(Q_F(x, a)\) of such approaches majorizes \(F\) and is typically quadratic in \(x\). In order to avoid complications in constructing a suitable surrogate function for the cost function in (5), the fourth order terms from \(\|I - UTU\|_F^2\) have to be avoided. In Reference [37], this problem is solved by introducing an auxiliary variable \(W \in \mathbb{R}^{MK}\) and by reformulating the minimization problem in (5) as

\[
\min_{U,V,W \geq 0} \frac{1}{2} \|X - UV\|_F^2 + \frac{\sigma_1}{2} \|I - WTU\|_F^2 + \frac{\sigma_2}{2} \|W - U\|_F^2 + \frac{\tau}{2} \|U\|_{TV^{\epsilon}}, \quad (7)
\]

For this problem, a suitable surrogate function and a multiplicative algorithm can be derived. The details of the derivation will not be discussed here and can be found in Reference [37] in all details. We also provide a short outline of the derivation in Appendix A.
In the following, we describe the final update rules obtained by the MM principle in Algorithm 2 and define for this purpose the matrices \( P(U), Z(U) \in \mathbb{R}_{\geq 0}^{M \times K} \) as

\[
P(U)_{mk} := \frac{1}{\|\nabla\epsilon TV_m U\|} + \sum_{\tilde{m} \in \mathcal{N}_m} \frac{1}{\|\nabla\epsilon TV_{\tilde{m}} U\|},
\]

(8)

\[
Z(U)_{mk} := \frac{1}{P(U)_{mk}} \left( \frac{1}{\|\nabla\epsilon TV_m U\|} \sum_{\tilde{m} \in \mathcal{N}_m} \frac{U_{mk} + U_{\tilde{m}k}}{2} + \sum_{\tilde{m} \in \mathcal{N}_m} \frac{U_{mk} + U_{\tilde{m}k}}{2\|\nabla\epsilon TV_{\tilde{m}} U\|} \right),
\]

(9)

where \( \mathcal{N}_m \) is the so-called adjoint neighborhood given by \( \tilde{m} \in \mathcal{N}_m \leftrightarrow m \in \mathcal{N}_{\tilde{m}} \).

**Algorithm 2 ONMFTV-MUL1**

1. **Input** \( X \in \mathbb{R}_{M<N}^{M \times N}, \ K \in \mathbb{N}, \ \sigma_1, \sigma_2, \tau > 0, \ i = 0 \)
2. **Initialize** \( U[0], W[0] \in \mathbb{R}_{>0}^{M \times K}, \ V[0] \in \mathbb{R}_{>0}^{K \times N} \)
3. **repeat**
   4. \( U[i+1] = \left[ U[i] \circ \left( \frac{XV[i]^T + \tau P(U[i]) \circ Z(U[i]) + (\sigma_1 + \sigma_2)W[i]}{\tau P(U[i]) \circ U[i]^T + \sigma_2U[i]^T + U[i]V[i]V[i]^T + \sigma_1W[i]W[i]^T U[i]^T} \right) \right]_{>0} \)
   5. \( V[i+1] = \left[ V[i] \circ \left( \frac{U[i+1]^T X}{U[i+1]^T U[i+1] V[i]} \right) \right]_{>0} \)
   6. \( W[i+1] = \left[ W[i] \circ \left( \frac{(\sigma_1 + \sigma_2)U[i+1]^T}{\sigma_1 U[i+1] U[i+1]^T W[i]^T + \sigma_2 W[i]} \right) \right]_{>0} \)
4. **until** Stopping criterion satisfied

We denote by \( \circ \), as well as the fraction line, the element-wise (Hadamard) multiplication and division, respectively. Due to the multiplicative structure of the update rules, the nonnegativity of the iterates is preserved. However, a strict positive initialization of \( U, V, \text{ and } W \) is needed to avoid numerical instabilities and the zero locking phenomenon caused by zero entries in the matrices, which is characteristic for all algorithms based on multiplicative update rules (see, e.g., Reference [9]). For the same reason, we perform a subsequent element-wise projection step for every matrix defined by \( |\lambda|_{>0} := \max\{\lambda, \epsilon P_1\} \) with \( \epsilon P_1 = 1 \times 10^{-16} \). Analogously, a projection step is applied for too large entries, with \( \epsilon P_2 = 1 \times 10^{35} \) being the corresponding parameter.

The asymptotic computational complexity of ONMFTV-MUL1 can be easily obtained by analyzing the performed matrix multiplications and the involved-for-loops in the algorithm (also see Reference [53]), which leads to a complexity of \( O(KMN + K^2N + K^2M) \). However, \( K \) is usually chosen such that \( K \ll \min\{M, N\} \). Hence, regarding the asymptotic computational complexity, we can consider \( K \) as a positive constant, so that we obtain \( O(MN) \).

Finally, the above algorithm ensures a monotone decrease of the cost function in (7) due to its construction based on the MM principle [37]. This leads to the convergence of the cost function values, since \( F(U, V, W) \) is bounded from below.

**Theorem 1 (ONMFTV-MUL1).** Algorithm 2 ensures a monotone decrease of the cost function defined by the NMF model in (7).

**ONMFTV-MUL2**

In this section, we derive another multiplicative algorithm, following the ideas in the work of Reference [35], based on a continuous formulation of an isotropic and differentiable version of the TV penalty term given by

\[
TV_{\epsilon TV}(u) := \int_{\Omega} \|\nabla u\|_{\epsilon TV} \, d(x_1, x_2)
\]

(10)
for $\Omega \subset \mathbb{R}^2$, a sufficiently smooth $u: \Omega \to \mathbb{R}$ with bounded variation (see, e.g., Reference [51]), and for

$$
\|\nabla u\|_{\text{TV}} := \sqrt{\left(\frac{\partial u}{\partial x_1}\right)^2 + \left(\frac{\partial u}{\partial x_2}\right)^2} + \epsilon_{\text{TV}}^2,
$$

with a small $\epsilon_{\text{TV}} > 0$. The application of the TV regularization on the discrete matrix $U$ is done via a subsequent discretization step, which is specified in more detail in Appendix B. Thus, we consider in this section the orthogonal NMF model

$$
\min_{U,V \geq 0} \frac{1}{2} \|X - UV\|^2 + \frac{\sigma_1}{4} \|I - UTU\|^2 + \tau \text{TV}_\epsilon(U),
$$

with the sloppy notation of $\text{TV}_\epsilon(U)$ for the matrix $U$, where the discretization step is implicitly included. Different from the model ONMFTV-MUL1 in the previous section, we do not include any auxiliary variable $W$.

The update rule for $U$ is based on a classical gradient descent approach

$$
U[i+1] := U[i] - \Gamma[i] \circ \nabla_u \mathcal{F}(U[i], V[i]),
$$

with a step size $\Gamma[i] \in \mathbb{R}_{\geq 0}^{M \times K}$. By computing the gradient $\nabla_u \mathcal{F}(U[i], V[i])$ and choosing an appropriate step size $\Gamma[i]$, this leads to the multiplicative update rule shown in Algorithm 3. For the minimization with respect to $V$, we simply choose the multiplicative update rule given in Algorithm 2. The term $\text{div} \left( \nabla U[i] / \|\nabla U[i]\|_{\text{TV}} \right)$ is again an abuse of notation and implicitly includes a discretization step (see Appendix B.2). It can be seen as the gradient of the TV penalty term and is obtained by analyzing the corresponding Euler-Lagrange equation (see Appendix B.1). Note that, in discretized form, it is a matrix of size $M \times K$ and can also contain negative entries. Hence, this update step is, strictly speaking, not a multiplicative update, since it cannot enforce the nonnegativity of the matrix $U$ by itself. However, as in Algorithm 2, the projection step given by $[\cdot]_{>0}$ is applied subsequently for both matrices to avoid numerical instabilities and to ensure the nonnegativity of $U$. Different from the approach in ONMFTV-MUL1, a monotone decrease of the cost function cannot be guaranteed based on the MM principle, since $\mathcal{F}$ also contains fourth-order terms due to the penalty term $\|I - UTU\|^2$.  

Algorithm 3 ONMFTV-MUL2

1. **Input** $X \in \mathbb{R}_{\geq 0}^{M \times N}$, $K \in \mathbb{N}$, $\sigma_1, \tau > 0$, $i = 0$
2. **Initialize** $U[0] \in \mathbb{R}_{\geq 0}^{M \times K}$, $V[0] \in \mathbb{R}_{\geq 0}^{K \times N}$
3. **repeat**
4. $U[i+1] = \left[ \frac{XV[i]\top + \tau \text{div} \left( \frac{\nabla U[i]}{\|\nabla U[i]\|_{\text{TV}}} \right) + \sigma_1 U[i]}{U[i+1]\top V[i]} \right]_{>0}$
5. $V[i+1] = \left[ \frac{U[i+1]\top X}{U[i+1]\top U[i+1]\top V[i]} \right]_{>0}$
6. $i \leftarrow i + 1$
7. **until** Stopping criterion satisfied

By analyzing the involved matrix multiplications in Algorithm 3, we can see that the asymptotic computational complexity is the same as in ONMFTV-MUL1. Hence, we obtain for ONMFTV-MUL2 $O(KMN + K^2N + K^2M)$.

Furthermore, note the similarity of the update rules given in Reference [9], where no total variation penalty term is considered. For more details on the derivation of Algorithm 3 and the discretization of the divergence term, we refer the reader to Appendix B.
3.2.2. Proximal Alternating Linearized Minimization

Adapting the optimization procedure of the very recent work of Reference [50], we consider in this section several Proximal Alternating Linearized Minimization (PALM) schemes. For all presented methods in this section, we consider the NMF model

$$\min_{U, V, W \geq 0} \frac{1}{2} \|X - UV\|_F^2 + \frac{\alpha_1}{2} \|I - WTU\|_F^2 + \frac{\alpha_2}{2} \|W - U\|_F^2 + \tau \|U\|_{TV},$$

(14)

where \(\|\cdot\|_{TV}\) is defined according to (4). One crucial step for the application of this optimization approach is to divide the whole cost function into a differentiable part \(\mathcal{F}\) and a non-differentiable part \(J\). The additional auxiliary variable \(W\) is needed to ensure the Lipschitz continuity of the partial gradients of \(\mathcal{F}\) and, hence, the convergence rates of the respective algorithms shown in Reference [50].

The general scheme of these algorithms are based on an alternating minimization procedure with a gradient descent step with respect to the differentiable function \(\mathcal{F}(U, V, W)\) via the computation of full gradients or gradient estimates and a subsequent column-wise application of the proximal operator for the non-differentiable part of the ONMF model. In the case of the algorithm for \(U\), the proximal operator with respect to the function \(J\) is evaluated, which leads to the update rule

$$U_{**k}^{[i+1]} := \left[\text{prox}_{\eta J}\left(U_{**k}^{[i]} - \eta (\nabla_U \mathcal{F}(U^{[i]}, V, W))_{**k}\right)\right]_{\geq 0},$$

(15)

for a suitable step size \(\eta > 0\). Furthermore, \(\nabla_U \mathcal{F}\) is either the partial derivative \(\nabla_U \mathcal{F}\) of \(\mathcal{F}\) with respect to \(U\) or some random gradient estimate (also see ONMFTV-SPRING). The nonnegativity constraint of the matrices is ensured by a final projection of all negative values to zero denoted in (15) by \([\cdot]_{\geq 0}\). In the following, we will write in short

$$U_{**}^{[i+1]} := \left[\text{prox}_{\eta J}\left(U_{**}^{[i]} - \eta \nabla_U \mathcal{F}(U^{[i]}, V, W)\right)\right]_{\geq 0},$$

for the whole matrix \(U\). The evaluation of the proximal mapping for all considered methods based on the PALM scheme in this section is based on the same FISTA algorithm [52] as it is the case for the separated approaches of Section 3.1, whereby the maximal iteration number is reduced to 5. As usual for PALM algorithms, adaptive step sizes based on the local Lipschitz constants of the partial gradients of \(\mathcal{F}\) are used, which will be approximated via the power iteration for all considered approaches. More information on the computation of these estimates and the choice of the step sizes are given in the subsequent descriptions of the specific algorithms and in Section 4 of the numerical experiments, as well as in Appendix C. Details on the derivation of the gradients and the Lipschitz constants for the specific NMF model in (14) are given in Appendix C.

Regarding the update rules for the matrices \(V\) and \(W\) for the considered ONMF model in (14), the application of the proximal operator can be neglected, since there is no non-differentiable part in (14) depending on either \(V\) or \(W\). Hence, only the corresponding gradient descent and projection steps are performed for both matrices (see Algorithms 4 and 5).

ONMFTV-PALM

The proximal alternating linearized minimization is based on the general algorithm scheme described above. The main steps are illustrated in Algorithm 4. For the gradient descent step, the full classical partial derivatives \(\nabla_U \mathcal{F}, \nabla_V \mathcal{F},\) and \(\nabla_W \mathcal{F}\) are computed without the consideration of any batch sizes. Furthermore, the function POWERIT denotes the power method to compute the needed step sizes. The derivation of the algorithm, together with the computation of the gradients and the Lipschitz constants for the step sizes, are described in Appendix C.1. The algorithm terminates until a suitable stopping
criterion is satisfied, which will be further specified in Section 4 and Appendix D. This approach will be henceforth referred to as ONMFTV-PALM.

As for the multiplicative algorithms ONMFTV-MUL1 and ONMFTV-MUL2, one can easily obtain the asymptotic computational complexity for Algorithm 4 by analyzing the involved matrix multiplications (also see Reference [53]), leading to the same complexity $O(KMN + K^2N + K^3M)$ as for proposed multiplicative update rules.

**Algorithm 4 ONMFTV-PALM**

1: Input $X \in \mathbb{R}^{M \times N}$, $K \in \mathbb{N}$, $\sigma_1, \sigma_2, \tau > 0$, $i = 0$
2: Initialize $U^{[0]}, W^{[0]} \in \mathbb{R}^{M \times K}$, $V^{[0]} \in \mathbb{R}^{K \times N}$
3: repeat
4: $\eta_{U^{[i]}} = \text{POWERIT}_U(V^{[i]}, W^{[i]})$
5: $U^{[i+1]} = \left[ \text{prox}_{\eta_{U^{[i]}} \mathcal{F}} \left( U^{[i]} - \eta_{U^{[i]}} \nabla_U \mathcal{F}(U^{[i]}, V^{[i]}, W^{[i]}) \right) \right]_{\geq 0}$
6: $\eta_{V^{[i]}} = \text{POWERIT}_V(U^{[i+1]})$
7: $V^{[i+1]} = \left[ V^{[i]} - \eta_{V^{[i]}} \nabla_V \mathcal{F}(U^{[i+1]}, V^{[i]}, W^{[i]}) \right]_{\geq 0}$
8: $\eta_{W^{[i]}} = \text{POWERIT}_W(U^{[i+1]})$
9: $W^{[i+1]} = \left[ W^{[i]} - \eta_{W^{[i]}} \nabla_W \mathcal{F}(U^{[i+1]}, V^{[i+1]}, W^{[i]}) \right]_{\geq 0}$
10: $i \leftarrow i + 1$
11: until Stopping criterion satisfied

**ONMFTV-iPALM**

A slightly extended version of ONMFTV-PALM is the so-called Inertial Proximal Alternating Linearized Minimization (iPALM) algorithm, which introduces an additional momentum term into Algorithm 4 and, hence, improves the convergence rate. Since the iPALM algorithm still follows the rough outline of ONMFTV-PALM and uses the classical partial gradients of $\mathcal{F}$, we will not present the whole algorithm at this point and refer the reader to the corresponding work in Reference [55], Appendix C.2, and our GitLab [53], where the corresponding codes are available online. This method will be referred to as ONMFTV-iPALM.

Since ONMFTV-iPALM follows the same major principles as ONMFTV-PALM, the asymptotic computational complexity of ONMFTV-iPALM is the same as in the case of ONMFTV-PALM.

**ONMFTV-SPRING**

The so-called Stochastic Proximal Alternating Linearized Minimization (SPRING) is an extended version of the PALM method, where the full gradients are replaced by random estimates. One basic assumption of this approach is that $\mathcal{F}$ is separable depending on the considered variable for which the cost function is minimized. In the case of the minimization with respect to $U$, the function $\mathcal{F}$ can be expressed as

$$
\mathcal{F}(U, V, W) = \sum_{n=1}^{N} \frac{1}{2} \| X_{*,n} - UV_{*,n} \|_2^2 + \sum_{k=1}^{K} \frac{\sigma_1}{2} \| I_{*,k} - U^TW_{*,k} \|_2^2 + \frac{\sigma_2}{2} \| W_{*,k} - W_{*,k} \|_2^2 .
$$

However, instead, we use the formulation

$$
\mathcal{F}(U, V, W) = \sum_{n=1}^{N} \left[ \frac{1}{2} \| X_{*,n} - UV_{*,n} \|_2^2 + \frac{1}{N} \mathcal{F}(U, W) \right] := \sum_{n=1}^{N} \mathcal{F}_n(U, V, W)
$$

to be able to compute the estimates of the gradients based on the functions $\mathcal{F}_n$. These random estimates are formed by using just a few indices of $\mathcal{F}_n$, which are the elements of the so-called mini-batch $n \in B_{ij}^T \subset \{1, \ldots, N\}$, where $i$ denotes the iteration number of the
SPRING algorithm, and \( j \in \{1, \ldots, s_r \} \) specifying the currently used subsample of indices, with \( 1/s_r \) being the subsample ratio. One classical example of a gradient estimator, which is also used in this work, is the Stochastic Gradient Descent (SGD) estimator given by

\[
\hat{\nabla}_{ij}^n \mathcal{F}(U, V, W) := \sum_{n \in B_{ij}^n} \nabla_{ij} \mathcal{F}_n(U, V, W).
\]

(16)

For other possible gradient estimators, such as the SAGA or SARAH estimator, we refer the reader to the work of Reference [50] and the references therein.

The case of the minimization with respect to \( V \) and \( W \) is more straightforward. For \( V \), the penalty terms for the orthogonality can be dropped, and we use the separability of the function with respect to the rows of \( X \) and \( U \). The computation of the SGD estimator is done analogously. For minimizing with respect to \( W \), the expression based on \( \mathcal{F}(U, W) \) can be used by omitting the data fidelity term. However, since \( K \ll \min\{M, N\} \), we use in this work the full gradient of \( \mathcal{F} \) for the minimization with respect to \( W \).

The main steps of the algorithm are presented in Algorithm 5, which will be referred to as ONMFTV-SPRING. For details on the choice of the hyperparameters and the computation of the gradients, as well as the step sizes, we refer the reader to Section 4 and Appendix C.

### Algorithm 5 ONMFTV-SPRING

1. **Input** \( X \in \mathbb{R}_{\geq 0}^{M \times N}, K \in \mathbb{N}, \sigma_1, \sigma_2, \tau > 0, s_r \in \mathbb{N}, i = 0 \)
2. **Initialize** \( U^{[i,0]}, W^{[i,0]}, V^{[i,0]} \in \mathbb{R}_{\geq 0}^{M \times K}, V^{[i,0]} \in \mathbb{R}_{\geq 0}^{K \times N} \)
3. **repeat**
   4. **for** \( j = 1, \ldots, s_r \) **do**
   5. \( \eta_{U[i,j]} = \text{POWERIT}_U(V^{[i,j]}, W^{[i,j]}) \)
   6. \( U^{[i,j+1]} = \left[ \text{prox}_{\eta_{U[i,j]} \mathcal{F}}(U^{[i,j]} - \eta_{U[i,j]} \hat{\nabla}_{ij}^n \mathcal{F}(U^{[i,j]}, V^{[i,j]}, W^{[i,j]})) \right]_{\geq 0} \)
   7. \( \eta_{V^{[i,j]}} = \text{POWERIT}_V(U^{[i,j+1]}) \)
   8. \( V^{[i,j+1]} = \left[ V^{[i,j]} - \eta_{V^{[i,j]}} \hat{\nabla}_{ij}^n \mathcal{F}(U^{[i,j+1]}, V^{[i,j]}, W^{[i,j]}) \right]_{\geq 0} \)
   9. \( \eta_{W^{[i,j]}} = \text{POWERIT}_W(U^{[i,j+1]}) \)
   10. \( W^{[i,j+1]} = \left[ W^{[i,j]} - \eta_{W^{[i,j]}} \nabla W \mathcal{F}(U^{[i,j+1]}, V^{[i,j+1]}, W^{[i,j]}) \right]_{\geq 0} \)
   11. **end for**
12. \( U^{[i+1,1]} = U^{[i,s_r+1]}, V^{[i+1,1]} = V^{[i,s_r+1]}, W^{[i+1,1]} = W^{[i,s_r+1]} \)
13. **i** \( \leftarrow i + 1 \)
14. **until** Stopping criterion satisfied

The asymptotic computational complexity of ONMFTV-SPRING can be obtained by analyzing the involved for-loops and matrix multiplications (see Reference [53]), leading to a complexity of \( O(KMN + s_r K^2 N + s_r K^2 M) \), which now also involves \( s_r \). However, note that \( s_r \) can be also seen as a positive constant, which in this case would lead to the same complexity as in ONMFTV-PALM.

### Convergence Analysis

While this work does not focus on a theoretical convergence analysis of the considered optimization methods of the ONMFT models, we provide in the following some brief information on the convergence properties of the PALM algorithms.

The convergence theory of PALM algorithms has been analyzed in various works [50,55,56]. The convergence properties depend heavily on the properties of the considered cost function. A non-standard extension of the usual PALM scheme in the case of our considered ONMF model in (14) is the introduction of a third auxiliary variable. However, according to Reference [50,56], the full convergence theory of the PALM and SPRING scheme easily extends to an arbitrary number of variables.
Furthermore, \( F \) and \( J \) in (14) have to satisfy specific properties to ensure basic convergence properties of the PALM algorithms. \( F \) needs to be a finite-valued, differentiable function with its gradient \( \nabla F \) being Lipschitz continuous on bounded sets of \( \mathbb{R}^{M \times K} \times \mathbb{R}^{K \times N} \times \mathbb{R}^{M \times K} \). This is obviously true in the case of (14) as a consequence of the mean value theorem, since \( F \) is a \( C^2 \) function. Furthermore, the partial gradients \( \nabla_U F, \nabla_V F, \) and \( \nabla_W F \) need to be Lipschitz continuous with modulus \( L_U, L_V, \) and \( L_W \), respectively, which is also true in our considered ONMF model (also see Appendix C).

Regarding the non-differentiable part in (14), \( J \) needs to be a proper lower semi-continuous function, which is bounded from below. Since the classic TV semi-norm used in (14) is lower semi-continuous (see Reference [51]), these properties also hold in our case.

To ensure the specific convergence property that a sequence of iterates \( Y^{[k]} := (U^{[k]}, V^{[k]}, W^{[k]}) \) converge to a critical point \( Y^* := (U^*, V^*, W^*) \) of the whole cost function \( G(U, V, W) := F(U, V, W) + J(U) \) in (14), additional requirements on \( G \) are needed. In the case of ONMFTV-PALM, \( G \) needs to be a so-called Kurdyka-Łojasiewicz (KL) function (see Reference [50,56]), whereas, for ONMFTV-SPRING, \( G \) has to be a semialgebraic function [50]. For a full-length treatise of the definition and properties of semialgebraic functions in the field of real algebraic geometry, we refer the reader to Reference [57]. The following Lemma is a nonsmooth version of the Łojasiewicz gradient inequality and can be found in Reference [56].

**Lemma 1.** Let \( f : \mathbb{R}^n \to (-\infty, +\infty) \) be a proper and lower semicontinuous function. If \( \sigma \) is semialgebraic, then \( \sigma \) fulfills the KL property at any point in the domain of \( \sigma \) and is, hence, a KL function.

Since \( G \) is obviously a proper and lower semicontinuous function, it is sufficient to show that \( G \) is semialgebraic. To do so, we note, first of all, that \( F(U, V, W) \) is a real polynomial function and, hence, semialgebraic. By using further basic properties of semialgebraic functions, it is also easy to see that \( J(U) \) is semialgebraic. Since \( J \) is given by the equation in (6) with \( \epsilon_{TV} = 0 \), it consists of a real polynomial function with a subsequent application of the square root function, which is still semialgebraic. Since finite sums of semialgebraic functions are semialgebraic, \( G \) is semialgebraic. Hence, the sequence of iterates \( Y^{[k]} \) produced by ONMFTV-PALM converge to a critical point of \( G \). In the case of ONMFTV-SPRING, the iterates converge in expectation to a critical point.

Similar arguments with additional assumptions on the momentum terms can be applied for ONMFTV-1PALM, so that we also achieve a convergence to a critical point for this algorithm. While the authors in Reference [55] do not prove improved convergence rates, they show that the additional momentum terms lead to an improved performance in practice. For further results on the convergence rates of the considered algorithms, we refer the reader to the aforementioned references.

### 4. Numerical Experiments

#### 4.1. Dataset

Concerning the numerical experiments of this work, we consider a hyperspectral dataset obtained from a Matrix-Assisted Laser Desorption/Ionization (MALDI) imaging measurement of a human colon tissue sample [58]. This technique is a Mass Spectrometry Imaging (MALDI-MSI) method, which is able to provide a spatial molecular profile of a given analyte. Together with the technological advancements in acquisition speed and robustness over the last decade, MALDI-MSI has become a standard tool in proteomics and applications in medical research, such as characterization of tumor subtypes and extraction of characteristic spectra [42], have become feasible.

In general, a measurement with a mass spectrometer can be subdivided into three main steps: the ionization of the sample, followed by the separation and detection of the ions. Considering MALDI imaging, the ionization part is characterized by the term MALDI. Different from other ionization techniques, such as the so-called Secondary-Ion
Mass Spectrometry (SIMS) or Desorption Electrospray Ionization (DESI), MALDI allows measurement of a wider mass range of the extracted ions. This is possible due to an application of a matrix onto the tissue sample to optimize the transfer of the ionization energy and to extract the molecules out of the analyte. The needed ionization energy is provided by a laser, which is shot on the tissue sample by following a grid pattern. The separation of the molecules is done by a mass analyzer. One typical method, which was also used for the dataset in this work, is to accelerate the ions by an electric field and to measure the Time-Of-Flight (TOF) of the particles. The final step of the whole measurement is the detection of the ions with the help of an ion detector.

Another major aspect of the whole MALDI-MSI workflow is the preparation of the analytes before the actual measurement with a MALDI-TOF device, which follows standardized protocols. For more information, we refer the reader to Reference [58,59].

For every point on the tissue slide, which is shot by the laser, a whole mass spectrum is acquired, leading to a hyperspectral dataset. The whole data is then written into the matrix $X \in \mathbb{R}^{M \times N}_{\geq 0}$, where each entry denotes the intensity of the detected particles of a specific mass-to-charge ratio (m/z-value). Typically, the measured spectra are ordered row-wise to the matrix, such that every column corresponds to an intensity plot of the whole tissue slide for a specific m/z-value (m/z-image).

Typical data sizes range from $1 \times 10^4$ to one million spectra and m/z-images, respectively. Furthermore, MALDI datasets are naturally nonnegative. These properties make the NMF an ideal analyzing tool for MALDI imaging datasets due to the nonnegativity constraints leading to a meaningful physical interpretation of the acquired matrices $U$ and $V$. For more details on the application of NMF models to MALDI imaging problems and the interpretation of the factorization matrices, we refer the reader to the works of Reference [37,42].

Figure 1 shows the human colon tissue dataset used for the subsequent numerical evaluation of the methods. Figure 1a is the histological image after the application of a so-called Hematoxylin and Eosin (H&E) staining, which allows a distinction between different tissue types. Figure 1b shows the histological annotation, which divides the dataset into six different classes and constitutes the ground truth for the subsequent numerical evaluation. The human colon dataset consists of 12,049 acquired spectra, each containing 20,000 measured m/z values covering a mass range of 600 Da to 4000 Da. However, we only consider the actual annotated spectra to ensure that each considered spectra can be reasonably classified in one of the 6 classes shown in Figure 1b. Hence, we restrict ourselves to 8725 spectra leaving out the zero and the non-annotated ones, which leads to the nonnegative data matrix $X \in \mathbb{R}^{8725 \times 20,000}_{\geq 0}$ for the numerical experiments.
4.2. Choice of Separated Methods

As discussed in Section 3.1, we consider the separated approaches based on the workflow given in Algorithm 1 as baseline for the comparison with the combined methods presented in Section 3.2. In this section, we specify shortly seven ONMF algorithms to compute the clustering in Line 3 of Algorithm 1, which are based on different works throughout the literature and will be used for the numerical comparison in Section 4.4 (see Table 1). Numerical tests for the optimization approach in the more recent work of Reference [8], which is based on a sparsity and nuclear norm minimization, did not lead to satisfactory results and, hence, will not be presented in this work.

Furthermore, we will evaluate the separated methods with and without the TV denoising step in Section 4.4. In such a way, we obtain a comparison between the clustering results of the classical ONMF approaches in Table 1 with and without the TV post-processing and by that an intuitive view on the advantage of adding a TV regularization procedure to the clustering method.

The choice of the hyperparameters and details on the initialization of the matrices, as well as the used stopping criteria of all separated methods, are discussed in Section 4.3 and Appendix D.1.

Table 1. Designations of the considered separated methods (left column) and short explanation of the corresponding ONMF algorithm (right column).

<table>
<thead>
<tr>
<th>Separated Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONMF-TV-Choi</td>
<td>Alternating multiplicative update rules based on Reference [5].</td>
</tr>
<tr>
<td>ONMF-TV-Ding</td>
<td>Alternating multiplicative update rules based on Reference [3].</td>
</tr>
<tr>
<td>ONMF-TV-Pompili1</td>
<td>Alternating expectation-maximization algorithm similar to the default spherical K-means algorithm [16,60].</td>
</tr>
<tr>
<td>ONMF-TV-Pompili2</td>
<td>Alternating algorithm based on an augmented Lagrangian approach with strict orthogonality constraints [16]. Nonnegativity is obtained asymptotically by using a quadratic penalty.</td>
</tr>
<tr>
<td>ONMF-TV-Kimura</td>
<td>Hierarchical alternating least squares algorithm, which is applied column-wise on $U$ and row-wise on $V$ [11].</td>
</tr>
<tr>
<td>ONMF-TV-Li</td>
<td>Hierarchical alternating least squares algorithm with approximate orthogonal-ity constraints and subsequent projection steps to ensure nonnegativity [12].</td>
</tr>
</tbody>
</table>

4.3. Setup

In this section, we describe the initialization methods and stopping criteria, the calculation of the final hard clustering, the choice of the various hyperparameters, and the used cluster validation measures. Furthermore, we give some further details on the general numerical setup.

For every considered separated and combined method in Sections 3.1 and 3.2, we perform 30 replicates of the experiment to get an impression of the performance stability, since the used initialization methods are partially based on randomly selected data points or matrices. For each method, we use the same random seed in the beginning of the experiment. Furthermore, we measure for each replicate the computational time, including the time for the calculation of the initialization of the factorization matrices.

For the evaluation of the clusterings, we use several different clustering validation measures discussed in Reference [61]. Due to the known ground truth of the data, we restrict ourselves to external clustering validation measures. Based on the results in this work, we primarily consider the so-called normalized Van Dongen criterion ($VD_n$) and the normalized Variation of Information ($VI_n$), since they give the most representative, quantitative measures in most of the general cases. Furthermore, they are normalized into the $[0,1]$ range and give reasonable results in cases there are clusters without any corresponding data points, which is different from classical measures, such as the purity. In addition, we consider as a secondary measure the Entropy ($E$). For all considered measures, it holds that a lower value indicates a better clustering performance.
To provide the definition of the clustering validation measures, we denote $n_{kk}$ as the number of data points in cluster $I_k$ from class $C_k$ for $k, \tilde{k} \in \{1, \ldots, K\}$ and define

$$n := \sum_{k, \tilde{k}=1}^{K} n_{kk}, \quad n_{k\bullet} := \sum_{k=1}^{K} n_{kk}, \quad n_{\bullet\bullet} := \sum_{k=1}^{K} n_{kk}, \quad p_{kk} := \frac{n_{kk}}{n}, \quad p_k := \frac{n_{k\bullet}}{n}, \quad \tilde{p}_k := \frac{n_{\bullet\bullet}}{n}.$$ 

Using this notation, the definition of all considered clustering validation measures are given in Table 2.

**Table 2. Definitions of all considered clustering validation measures.**

<table>
<thead>
<tr>
<th>Measure</th>
<th>Definition</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entropy (E)</td>
<td>$-\sum_{k=1}^{K} p_k \left( \sum_{k=1}^{K} \frac{p_{kk}}{p_k} \log \left( \frac{p_{kk}}{p_k} \right) \right)$</td>
<td>$[0, \log(K)]$</td>
</tr>
<tr>
<td>Normalized Variation of Information (Vln)</td>
<td>$1 + 2 \cdot \frac{\sum_{k=1}^{K} p_k \log(p_k) + \sum_{k=1}^{K} \tilde{p}_k \log(\tilde{p}<em>k)}{\sum</em>{k=1}^{K} p_k \log(p_k)}$</td>
<td>$[0, 1]$</td>
</tr>
<tr>
<td>Normalized Van Dongen criterion (VDn)</td>
<td>$\frac{2n - \sum_{k=1}^{K} \max_k {n_{kk}} - \sum_{k=1}^{K} \max_k {n_{kk}}}{2n - \max_k {n_{k\bullet}} - \max_k {n_{\bullet\bullet}}}$</td>
<td>$[0, 1]$</td>
</tr>
</tbody>
</table>

Concerning the initialization approaches, we consider either the classical K-means++ method or an initialization based on the singular value decomposition of the data matrix $X$ by following the works of Reference [62,63]. In short, the latter method is based on the computation of a truncated Singular Value Decomposition (SVD) of the data matrix following a Krylov approach described in Reference [62] and performing specific projection, as well as normalization steps, based on the algorithm described in Reference [63]. For further details on the initialization, we refer the reader to the provided code in the GitLab [53]. For K-means-TV, we use K-means++ as the typical initialization method for the K-means algorithm. To achieve the optimal results for each of the remaining proposed and comparative methods, both initialization methods are tested. The initialization method which leads to a better clustering stability and performance in terms of the VDn is chosen. However, it turns out that, for all methods, except K-means-TV, ONMF-TV-Ding, and ONMF-TV-Pompili11, the initialization based on the SVD of $X$ leads to better clustering results. The concrete choices for every ONMF model are described in Appendix D.

Regarding the stopping criteria, we simply set for the considered ONMF models a maximal iteration number until a sufficient convergence is reached, except for K-means-TV, ONMF-TV-Pompili11, and ONMF-TV-Pompili12, where we use the internal stopping criteria of the respective algorithms. For more information, we refer the reader to Appendix D and the work of Reference [16].

Another aspect is the computation of the final clustering based on the cluster membership matrix $U$. Most of the considered ONMF models yield a cluster membership matrix $U$ having multiple positive entries in each row, which is related to a soft clustering. To obtain the final hard clustering, we assign every data point to the cluster, which corresponds to the column in $U$, where the maximal value in the row is attained. In the case that there are two or more equal entries in one row, we choose the cluster by a random choice.

A main part of the whole workflow of the numerical evaluation is the choice of the various hyperparameters of the considered ONMF models. In particular, these include the regularization parameters $c_1, c_2$, and $\tau$ of the combined methods and the parameter $\tau$ of the TV post-processing of the separated methods. For all considered methods, we perform a grid search of the corresponding parameters. An appropriate subset of the parameter space is chosen and experiments for a wide range of possible combinations of parameters are performed. For each considered method, the parameter configuration leading to the
best performance stability and $\text{VD}_n$ is chosen. More details and the specific selection of the hyperparameters are given in Appendix D.

Furthermore, we perform several projection steps to enhance the numerical stability. For all considered methods, we project the data matrix $X$ by applying $\|\cdot\|_0$ defined as in Section 3.2.1. Furthermore, we perform specifically for the multiplicative update rules and for ONMF-TV-Kimura the same projection step after the initialization of the matrices. For the combined method ONMFTV-SPRING, we perform an additional projection step of the parameter $\tau U_{ij}$ used for the application of the proximal operator described in Algorithm 5 to avoid too large parameters for the TV regularization. A similar projection is done for the step size in each gradient descent step. More details are given in Appendix C.3.

The algorithms were implemented with MATLAB® R2021a and executed on an Intel® Core™ i7-7700K quad core CPU @4.20 GHz with 32 GB of RAM. The corresponding MATLAB® codes can be found in our GitLab [53].

4.4. Results and Discussion

In this section, we present and discuss the numerical results obtained in the evaluation discussed in the previous sections.

Figure 2 shows the clusterings of the separated methods without the subsequent TV regularization step of the best performed replicate with respect to the $\text{VD}_n$, which was obtained after the TV post-processing. In general, every separated method is able to identify all classes shown in Figure 1b, except the distinction between the tumor and mucosa. Since the combined methods are also not able to distinguish between both classes (see Figure 3), this is probably due to the fundamental, underlying NMF model with the orthogonality constraints, which are not able to identify the regions in an unsupervised workflow. In the case that annotated datasets are available for training, supervised machine learning methods could lead to different results. However, this work does not focus on these kind of approaches.

Figure 2. Ground truth and clusterings of all considered separated methods without TV regularization. The best-performing replicate, including the TV regularization, is chosen based on the normalized van Dongen criterion.
Figure 3. Ground truth and clusterings of all considered methods, including the TV regularization. The best-performing replicate, including the TV regularization, is chosen based on the normalized van Dongen criterion.

Furthermore, every separated method is not able to guarantee a spatially coherent clustering in the region of the muscularis. Moreover, every method, except K-means-TV, does not provide any spatial coherence in any of the classes of the tissue slide. This is in contrast to the results in Figure 3, which shows the clusterings of all considered methods of the best performed replicate with respect to the VDn, including the TV regularization. Every method is able to provide a spatially coherent clustering with some few exceptions in the region of the muscularis and, hence, leads to significantly improved clusterings, in general. Comparing the results within Figure 3, the clustering of ONMFTV-SPRING seems to be the one, which best reproduces the given annotations. Furthermore, we note that some methods lead to clusterings with clusters, which do not contain any data points (see, i.e., K-means-TV, ONMF-TV-Pompili1, ONMFTV-PALM). However, this is also owing to the fact that the class of the lymphocytes is significantly underrepresented compared to the other
ones (see Figure 1b). Furthermore, this behavior is also dependent on the choice of the TV regularization parameter $\tau$.

For a quantitative evaluation, we provide in Figure 4 box plots of all replicates of every performed experiment and for all considered methods with respect to the $VD_n$ and $VI_n$. For each combined method, one box plot is plotted, which visualizes the corresponding clustering validation measure of all 30 replicates of the experiment through its quartiles. As usual, the line in between the box indicates the median of the validation measure of all replicates. Furthermore, the vertical lines are the so-called whiskers, which indicate the variability outside the lower and upper quartile. Finally, each plotted point in the graph represents one replicate of the corresponding experiment indicating the obtained validation measure.

For every separated method, two box plots are plotted in different colors, each representing the obtained clustering validation measure with or without the TV post-processing (see the legend). In such a way, a comparison of the classical ONMF algorithms of Table 1 with and without the total variation regularization is possible.

Based on such a visualization, a quantitative evaluation and comparison of all considered methods by considering different clustering validation measures is possible. Furthermore, the box plots visualize the performance variability of every method and, hence, give some indications on the stability of the approaches. As described in Section 4.3, lower values of the $VD_n$ and $VI_n$ indicate better clustering results.

For both measures, the observations of the qualitative evaluation above can be confirmed. First, we note that, for all separated methods, the TV post-processing does indeed lead to clusterings with better cluster validation measures. Moreover, the combined methods based on the PALM scheme achieve the best results with respect to both measures, from which ONMFTV-PALM and ONMFTV-iPALM achieve the highest performance stability. While some experiments of ONMFTV-SPRING attain the best values compared to all other methods, this approach is less stable than the non stochastic approaches ONMFTV-PALM and ONMFTV-iPALM. Furthermore, we note that both combined methods ONMFTV-MUL1 and ONMFTV-MUL2 based on the multiplicative update rules do not perform better than some of the other separated methods. Comparing the separated methods with each other, we see that ONMFTV-L1 performs remarkably well, with a high stability compared to the other approaches. Note that the stability also seems to depend on the initialization procedure. Regarding this, the SVD seems to favor more stable results than the $K$-means++ initialization.

Similar to the cluster validation measures $VD_n$ and $VI_n$ shown in Figure 4, Figure 5 displays the entropy of all performed experiments for all methods. As described in Section 4.3, lower values of the entropy measure indicate better clustering results. This measure also confirms the observation, that the combined methods ONMFTV-SPRING, ONMFTV-PALM, and ONMFTV-iPALM achieve the best results. Concerning the other methods, the outcomes are similar to the ones of the $VD_n$ and $VI_n$ and shall not be discussed in detail.

Figure 6 shows the box plots of the computational times of all replicates for every method. For each considered approach, one box plot is plotted visualizing the computational cost of all performed replicates of the experiment. As in the previous figures, each plotted point corresponds to a replicate showing the specific needed computational time.

The combined methods ONMFTV-PALM and ONMFTV-iPALM are one of the fastest methods, together with $K$-means-TV, which requires the least time to compute the experiments. Furthermore, we note that ONMFTV-SPRING needs significantly more time compared to the other PALM algorithms, which can be also seen by the slightly different asymptotic computational complexities (see Section 3.2.2). The other separated methods are faster than ONMFTV-SPRING, except ONMFTV-Pompilj12, which needs significantly more time than every other considered approach.

All in all, based on the experiments performed on the MALDI dataset, we recommend the methods based on the PALM scheme, particularly ONMFTV-PALM, as well as
ONMFTV-iPALM, since they give the most stable results, while achieving comparatively good results with low computational effort.

Figure 4. Box plots of the normalized van Dongen criterion ($V_{Dn}$) and the normalized variation of information ($V_{In}$) of all performed experiments.

Figure 5. Box plot of the Entropy (E) of all performed experiments.
Figure 6. Box plot of the computational times in minutes of all performed experiments.

5. Conclusions

In this work, we have considered various orthogonal nonnegative matrix factorization (ONMF) models, together with different optimization approaches for clustering hyperspectral data, as the main application field. Furthermore, we have introduced total variation regularization in the proposed ONMF models to ensure spatial coherence in the obtained clusterings constituting the main innovation in this paper motivated by numerous spectral imaging applications, which naturally satisfy the spatial coherence in the data.

After a brief description of the main principles of ONMF models, their relation to classical clustering methods, and different optimization techniques, we have proposed so-called separated methods, which apply the TV denoising step after the computation of a classical ONMF algorithm. Furthermore, we have introduced more sophisticated combined methods with different optimization procedures, which include the TV regularization into the considered ONMF model.

For the numerical evaluation, we have compared 12 different TV regularized ONMF methods on a MALDI-MSI human colon hyperspectral dataset with six different spatially coherent tissue regions, which constitute the ground truth for the clustering problem. The qualitative and quantitative results confirmed our expectation that the TV regularization significantly improves the clustering performance. Furthermore, the combined methods based on the proximal alternating linearized minimization have led to the best clustering outcomes and performance stability. Hence, based on the numerical evaluation of the MALDI dataset, we recommend the methods ONMFTV-PALM and ONMFTV-iPALM, as well as ONMFTV-SPRING.

Several further research directions could be of interest. One limitation of the presented approaches is the need of a manual a-priori choice of the needed hyperparameters. Hence, a useful extension of the proposed methods could be to introduce an automated way to choose suitable parameters. Another aspect is the analysis and the derivation of optimization algorithms for the case of discrepancy terms different from the Frobenius norm. Moreover, further gradient estimators different from the SGD could be examined for the method ONMFTV-SPRING. Furthermore, another major point is the consideration of more hyperspectral datasets from different application fields and a more thorough numerical evaluation of the different ONMF methods.

A more theoretical research direction could be an extended convergence analysis in particular for the multiplicative algorithms ONMFTV-MUL1 and ONMFTV-MUL2. Finally, the investigation of spatially coherent clustering models in infinite dimension space leading to “continuous” factorization problems with gradient based penalty terms could be interesting. In this setting, the analysis of first order conditions could lead to connections to corresponding K-means clustering models and partial differential equations, whose solutions give insight to the according distance measures and clusters. A first step for such an investigation could be to start with a finite dimensional space based on ONMF models.
Funding: This project was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) within the framework of RTG “π³: Parameter Identification—Analysis, Algorithms, Applications”—Project number 281474342/GRK2224/1.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The MATLAB® codes of the considered algorithms in this work are available in our GitLab [53] for general nonnegative hyperspectral datasets. Restrictions apply to the availability of the hyperspectral dataset used in this work as it was obtained with the permission of the third party SCiLS (see Reference [58] and www.scils.de (accessed on 27 September 2021)).

Acknowledgments: The author would like to thank Sibylle Hess (Eindhoven University of Technology) for fruitful discussions. The used tissue sample is a SCiLS example dataset provided by Axel Walch (German Research Center for Environmental Health, Helmholtz Zentrum München).

Conflicts of Interest: The author declares that there is no conflict of interest regarding the publication of this paper.

Abbreviations
The following abbreviations are used in this manuscript:

H&E Hematoxylin and Eosin
E Entropy
iPALM Inertial PALM
MALDI Matrix-Assisted Laser Desorption/Ionization
MM Majorize-Minimization
NMF Nonnegative Matrix Factorization
ONMF Orthogonal NMF
PALM Proximal Alternating Linearized Minimization
SVD Singular Value Decomposition
SGD Stochastic Gradient Descent
SPRING Stochastic Proximal Alternating Linearized Minimization
TV Total Variation
VDn Normalized Van Dongen Criterion
VIn Normalized Variation of Information

Appendix A. Derivation of the Algorithm ONMFTV-MUL1
In this section, we give an outline of the proof of Theorem 1. For more details on the derivation, we refer the reader to the work of Reference [37].

The update rules in Algorithm 2 are based on the Majorize-Minimization (MM) principle [64]. The basic idea behind this concept is to replace the cost function $\mathcal{F}$ by a so-called surrogate function $Q_F$, whose minimization is simplified and leads to the desired multiplicative algorithms.

Definition A1. Let $\Omega \subset \mathbb{R}^{M \times K}$ be an open set, and $\mathcal{F}: \Omega \to \mathbb{R}$ a given cost function. A function $Q_F : \Omega \times \Omega \to \mathbb{R}$ is called a surrogate function, if it satisfies the following properties:

(i) $Q_F(U, A) \geq \mathcal{F}(U)$ for all $U, A \in \Omega$,
(ii) $Q_F(U, U) = \mathcal{F}(U)$ for all $U \in \Omega$.

The minimization procedure based on the MM approach is defined by

$$U^{[i+1]} := \arg \min_{U \in \Omega} Q_F(U, U^{[i]}).$$
Together with the properties of $Q_F$, this leads directly to the monotone decrease of the cost function $F$, since

$$F(U^{[i+1]}) \leq Q_F(U^{[i+1]}, U^{[i]}) \leq Q_F(U^{[i]}, U^{[i]}) = F(U^{[i]}).$$

Tailored construction techniques for surrogate functions leads additionally to the desired multiplicative structure of the update rules and are typically based on Jensen’s inequality or the so-called quadratic upper bound principle [37,64].

We first focus on the minimization with respect to $U$. It can be shown that, for each term of the cost function $F$ in (7), a surrogate function can be constructed, which finally results to a suitable surrogate $Q_F := Q_{F_1} + Q_{F_2} + Q_{F_3}$ defined by

$$Q_{F_1}(U, A) := \frac{1}{2} \sum_{m=1}^{M} \sum_{n=1}^{N} \frac{1}{(AV)_{mn}} \sum_{k=1}^{K} A_{ik} V_{kn} \left( X_{mn} - \frac{U_{mk}}{A_{mk}} (AV)_{mn} \right)^2,$$

$$Q_{F_2}(U, A) := \frac{\sigma_2}{2} \sum_{k=1}^{K} \sum_{i=1}^{M} \sum_{m=1}^{M} W_{mk} A_{ml} \left( \delta_{ik} - \frac{U_{mk}}{A_{mk}} (W^T A)_{ik} \right)^2 + \frac{\sigma_2}{2} \| W - U \|^2_2,$$

$$Q_{F_3}(U, A) := \frac{\tau}{2} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{i=1}^{M} \left[ P(A)_{mk} (U_{mk} - Z(A)_{mk})^2 \right] + C(A),$$

where $C(A)$ is some function depending only on $A$, and with $P(A)$ and $Z(A)$ given as in (8) and (9). Computing the first order condition $\nabla_U Q_F(U, A) = 0$ and applying classical calculation rules for derivatives leads then to the desired update rule given in Line 4 of Algorithm 2.

The update rules for $V$ and $W$ are treated similarly.

**Appendix B. Details on the Algorithm**

**ONMFTV-MUL2**

**Appendix B.1. Derivation of the Update Rules**

Algorithm 3 is based on a classical gradient descent approach with a suitably chosen step size to ensure a multiplicative structure of the update rules. We will discuss here only the derivation for the minimization with respect to $U$. The update rules for $V$ are classical results and can be found in various works [37,49].

Regarding the update rule for $U$, we consider the classical gradient descent step

$$U^{[i+1]} := U^{[i]} - \Gamma^{[i]} \circ \nabla_U F(U^{[i]}, V),$$

given also in (13). Thus, we need an explicit expression for $\nabla_U F(U, V)$. Classical calculation rules for derivatives yields

$$\nabla_U F(U, V) = U V V^T - X V^T + \sigma_1 (U U^T U - U) + \nabla_U TV_{\varepsilon TV}(U).$$

The gradient of $TV_{\varepsilon TV}(U)$ can be acquired via the Euler-Lagrange equation and is a classical result. By considering the continuous case with the function $u : \Omega \to \mathbb{R}$ of Section 3.2.1 and defining $L(x_1, x_2, u, u_1, u_2) := \| \nabla u \|_{\varepsilon TV}$ with $u_i := \partial u / \partial x_i$ for $i \in \{1, 2\}$, the Euler Lagrange Equation

$$\frac{\partial L}{\partial u} - \sum_{i=1}^{2} \frac{\partial}{\partial x_i} \left( \frac{\partial L}{\partial u_i} \right) = 0$$

gives the formal relationship $\nabla_u TV_{\varepsilon TV}(u) := -\text{div}(\nabla u / \| \nabla u \|_{\varepsilon TV})$. Thus, the gradient descent step is given by

$$U^{[i+1]} := U^{[i]} - \Gamma^{[i]} \circ \left( U^{[i]} V V^T - X V^T - \tau \text{div} \left( \frac{\nabla U^{[i]}}{\| \nabla U^{[i]} \|_{\varepsilon TV}} \right) + \sigma_1 U^{[i]} U^{[i]T} U^{[i]} - \sigma_1 U^{[i]} \right).$$
To ensure a multiplicative structure of the update rules, we set the step size to be

$$\Gamma[i] := \frac{U[i]}{U[i]V^T + \sigma_1 U[i]^T U[i]^T},$$

which leads directly to the update rule in Line 4 of Algorithm 3.

**Appendix B.2. Discretization of the TV Gradient**

In this section, we describe the discretization procedure of the divergence term $\text{div} \left( \frac{\nabla U[i]}{\| \nabla U[i] \|_{TV}} \right)$, which occurs in Line 4 of Algorithm 3.

To perform such a discretization, it is needed for express the divergence term in terms of sums and products of first and second order derivatives. To simplify the notation, we stick in this first step to the continuous case and consider the function $u$ defined in (10). By considering the definition in (11) and applying classical calculation rules for derivatives, we get

$$\text{div} \left( \frac{\nabla u}{\| \nabla u \|_{TV}} \right) = \frac{\varepsilon_{TV}^2 \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial y} \right)^2 - 2 \frac{\partial u}{\partial x} \frac{\partial u}{\partial y} \frac{\partial^2 u}{\partial x \partial y}}{\| \nabla u \|_{TV}^2}.$$

For the discretization of this expression, we assume that $u$ is a given discretized image of size $d_1 \times d_2$. Note that, in the NMF models, which are considered in this work, this would correspond to one (reshaped) column of the matrix $U$. To approximate the partial derivatives in the above expression, we use in the following a central differencing scheme and interpret the $x$ and $y$ directions as the vertical and horizontal axis in the image $u$, respectively. Thus, we define

$$\begin{align*}
(\Delta_x u)_{ij} &:= \frac{u_{i+1,j} - u_{i-1,j}}{2}, \\
(\Delta_{xx} u)_{ij} &:= \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{2}, \\
(\Delta_{xy} u)_{ij} &:= \frac{u_{i,j+1} + u_{i,j-1} - u_{i-1,j} + u_{i+1,j}}{2}.
\end{align*}$$

The discretized gradients can also be interpreted as matrices of size $d_1 \times d_2$, which finally leads to the discretization of the divergence term

$$\text{div} \left( \frac{\nabla u}{\| \nabla u \|_{TV}} \right) = \frac{\varepsilon_{TV}^2 \left( \Delta_{xx} u + \Delta_{yy} u \right) + (\Delta_{xx} u)^2 \circ \Delta_{yy} u + (\Delta_{yy} u)^2 \circ \Delta_{xx} u - 2 \Delta_{xx} u \circ \Delta_{yy} u \circ \Delta_{xy} u}{(\| \Delta_{xx} u \|_2^2 + (\| \Delta_{yy} u \|_2^2 + \varepsilon_{TV}^2 1_{d_1 \times d_2})^{3/2}}$$

where $1_{d_1 \times d_2}$ is a matrix of size $d_1 \times d_2$ with ones in every entry.

**Appendix C. Algorithmic Details on the Proximal Gradient Descent Approach**

In this section, we give some information about the involved gradients, Lipschitz constants, and the computation of the step sizes of all algorithms based on the proximal gradient descent approach (see Section 3.2.2).

**Appendix C.1. ONMFTV-PALM**

We start in this section with details on the gradients of the algorithm ONMFTV-PALM. Since the computations are straightforward, we will only state the final results of all three gradients:

$$\begin{align*}
\nabla_U F(U, V, W) &= UVV^T - XVT + \sigma_1 (WW^T U - W) + \sigma_2 (U - W), \\
\nabla_V F(U, V, W) &= U^T UV - U^T X, \\
\nabla_W F(U, V, W) &= \sigma_1 (UU^T W - U) + \sigma_2 (W - U).
\end{align*}$$
For the calculation of the Lipschitz constants, we compute for the minimization with respect to \( \mathcal{U} \) and arbitrary matrices \( \mathcal{U}_1, \mathcal{U}_2 \in \mathbb{R}^{M \times k} \)

\[
\| \nabla_{\mathcal{U}} \mathcal{F}(\mathcal{U}_1, \mathcal{V}, \mathcal{W}) - \nabla_{\mathcal{U}} \mathcal{F}(\mathcal{U}_2, \mathcal{V}, \mathcal{W}) \| = \| (\mathcal{U}_1 - \mathcal{U}_2) \mathcal{V} \mathcal{V}^\top + c_1 \mathcal{W} \mathcal{W}^\top (\mathcal{U}_1 - \mathcal{U}_2) + c_2 (\mathcal{U}_1 - \mathcal{U}_2) \| \\
\leq (\| \mathcal{V} \mathcal{V}^\top \| + \| c_1 \mathcal{W} \mathcal{W}^\top + c_2 I_{M \times M} \|) \cdot \| \mathcal{U}_1 - \mathcal{U}_2 \| \\
= (\lambda_{1,\mathcal{U}} + \lambda_{2,\mathcal{U}}) \cdot \| \mathcal{U}_1 - \mathcal{U}_2 \|,
\]

where \( \| \cdot \| \) is the spectral norm, and \( \lambda_{1,\mathcal{U}}, \lambda_{2,\mathcal{U}} \) the maximal absolute eigenvalue of the symmetric matrices \( \mathcal{V} \mathcal{V}^\top \) and \( c_1 \mathcal{W} \mathcal{W}^\top + c_2 I_{M \times M} \), respectively, with \( I_{M \times M} \) being the identity matrix of size \( M \times M \). The other cases are treated analogously and result in

\[
\| \nabla_{\mathcal{V}} \mathcal{F}(\mathcal{U}, \mathcal{V}_1, \mathcal{W}) - \nabla_{\mathcal{V}} \mathcal{F}(\mathcal{U}, \mathcal{V}_2, \mathcal{W}) \| \leq \| \mathcal{U} \mathcal{U}^\top \| \cdot \| \mathcal{V}_1 - \mathcal{V}_2 \| = \lambda_{\mathcal{V}} \| \mathcal{V}_1 - \mathcal{V}_2 \|,
\]

\[
\| \nabla_{\mathcal{W}} \mathcal{F}(\mathcal{U}, \mathcal{V}, \mathcal{W}_1) - \nabla_{\mathcal{W}} \mathcal{F}(\mathcal{U}, \mathcal{V}, \mathcal{W}_2) \| \leq \| c_1 \mathcal{U} \mathcal{U}^\top + c_2 I_{M \times M} \| \cdot \| \mathcal{W}_1 - \mathcal{W}_2 \| = \lambda_{\mathcal{W}} \| \mathcal{W}_1 - \mathcal{W}_2 \|.
\]

Numerically, the eigenvalues of the matrices are approximated via the power iteration with 5 iterations. The step sizes \( \eta_{\mathcal{U}}, \eta_{\mathcal{V}}, \) and \( \eta_{\mathcal{W}} \) described in Algorithm 4 are computed based on the Lipschitz constants, which are given by \( \mathcal{L}_{\mathcal{U}} := \lambda_{1,\mathcal{U}} + \lambda_{2,\mathcal{U}}, \mathcal{L}_{\mathcal{V}} := \lambda_{\mathcal{V}}, \) and \( \mathcal{L}_{\mathcal{W}} := \lambda_{\mathcal{W}} \) with the computation above. The standard choice for the step size of the ONMFTV-iPALM algorithm are simply the inverse Lipschitz constants, i.e. \( \eta_{\mathcal{U}} = 1/\mathcal{L}_{\mathcal{U}}, \eta_{\mathcal{V}} = 1/\mathcal{L}_{\mathcal{V}}, \) and \( \eta_{\mathcal{W}} = 1/\mathcal{L}_{\mathcal{W}} \) [50,56].

**Appendix C.2. ONMFTV-iPALM**

The calculation of the gradients and Lipschitz constants for ONMFTV-iPALM are the same as in the case of ONMFTV-PALM with the slight difference that the gradients are evaluated at other points. Furthermore, the step sizes are set by \( \eta_{\mathcal{U}} = 0.9/\mathcal{L}_{\mathcal{U}}, \eta_{\mathcal{V}} = 0.9/\mathcal{L}_{\mathcal{V}}, \) and \( \eta_{\mathcal{W}} = 0.9/\mathcal{L}_{\mathcal{W}} \) according to Reference [50]. For details, we refer the reader to Reference [50,55] and to the codes available in our GitLab [53].

**Appendix C.3. ONMFTV-SPRING**

The computation of the gradients for ONMFTV-SPRING are based on the SGD estimator given in (16). As in the case of ONMFTV-PALM, the computations are straightforward, and we get, by defining the mini-batch \( B_{ij}^V \subseteq \{ 1, \ldots, M \} \) for the minimization with respect to \( V \) and with \( \mathcal{F}_m(\mathcal{U}, \mathcal{V}, \mathcal{W}) := 1/2 \| X_{m, \bullet} - \mathcal{U}_{m, \bullet} \mathcal{V} \|_2^2 \) the gradients

\[
\hat{\nabla}_m^V \mathcal{F}(\mathcal{U}, \mathcal{V}, \mathcal{W}) = \sum_{m \in B_{ij}^V} \left[ \left( (\mathcal{U} \mathcal{V})_{m, \bullet} \mathcal{V}_{m, \bullet} - X_{m, \bullet} \mathcal{V}_{m, \bullet} + 1/N (c_1 (\mathcal{W} \mathcal{W}^\top \mathcal{U} - \mathcal{W}) + c_2 (\mathcal{U} - \mathcal{W})) \right) \right]
\]

\[
= \mathcal{U} \mathcal{V}_{\bullet, B_{ij}^U} \mathcal{V}_{\bullet, B_{ij}^V} - X_{\bullet, B_{ij}^V} \mathcal{V}_{\bullet, B_{ij}^V} + \frac{|B_{ij}^U|}{N} (c_1 (\mathcal{W} \mathcal{W}^\top \mathcal{U} - \mathcal{W}) + c_2 (\mathcal{U} - \mathcal{W})),
\]

\[
\hat{\nabla}_m^U \mathcal{F}(\mathcal{U}, \mathcal{V}, \mathcal{W}) = \sum_{m \in B_{ij}^U} \nabla \mathcal{F}_m(\mathcal{U}, \mathcal{V}, \mathcal{W}) = \sum_{m \in B_{ij}^U} \mathcal{U}_{m, \bullet} \mathcal{V}_m - \mathcal{U}_{m, \bullet} X_{m, \bullet}
\]

\[
= \mathcal{U}_{\bullet, B_{ij}^U} \mathcal{V}_{B_{ij}^U} \mathcal{V} - \mathcal{U}_{\bullet, B_{ij}^U} \mathcal{V}_{B_{ij}^U} X_{B_{ij}^U, \bullet},
\]

where \( V_{\bullet, B_{ij}^U}, X_{\bullet, B_{ij}^U}, \) and \( U_{\bullet, B_{ij}^U}, X_{\bullet, B_{ij}^U} \) are the submatrices of \( U, V, \) and \( X, \) which are constrained column-wise and row-wise based on the index sets \( B_{ij}^U \) and \( B_{ij}^V, \) respectively. For the minimization with respect to \( W, \) the full partial gradient of \( \mathcal{F} \) is used, which was already computed in Appendix C.1.
The computation of the Lipschitz constants of the partial gradients of \( F \) with respect to \( U \) and \( V \) is based on the SGD estimator and goes analogously to the steps in Appendix C.1. Hence, we have

\[
\| \nabla_U^i F(U, V, W) - \nabla_U^i F(U_2, V, W) \| \leq \left( \| V \cdot B^i_v \| + \frac{|B^i_v|}{N} \| v_1 W W^T + \sigma I_{M \times M} \| \right) \| U_1 - U_2 \|,
\]

\[
\| \nabla_U^i F(U, V, W) - \nabla_U^i F(U, V_2, W) \| \leq \| U^T J_{\nabla_U^i} Y \| \| U_1 - U_2 \| = \lambda_{1, U} \| V_1 - V_2 \|,
\]

together with the Lipschitz constants \( L_U, L_V \) analogously to Appendix C.1. Since we consider the full partial gradient of \( F \) with respect to \( W \), we also get the same Lipschitz constant \( L_W \) as in Appendix C.1.

The choice of \( \eta_{U, \text{stopCrit}} \) and \( \eta_{V, \text{initMethod}} \) are chosen according to the work of Reference [50] by defining

\[
\eta_{U, \text{stopCrit}} := \min \left\{ \frac{1}{\sqrt{[i : \| B^i_u \| / N] \cdot L_U}}, \frac{1}{L_U} \right\}, \quad \eta_{V, \text{initMethod}} := \min \left\{ \frac{1}{\sqrt{[i : \| B^i_v \| / M] \cdot L_V}}, \frac{1}{L_V} \right\}.
\]

Furthermore, as described in Section 4.3, we perform an additional projection step of the parameter \( \| \nabla_U^i \| \) (see Line 6 in Algorithm 5 and the application of the prox\(_{\eta_{i,U}J} \) operator) to avoid too large regularization parameters of the TV penalty term by applying prox\(_{\eta_{i,U}J} \) with \( \tau_i := \min \{ \tau_{i,U,\text{stopCrit}}, 1 \times 10^{-3} \} \). For more details on the algorithm, we refer the reader to the codes available in our GitLab [53].

### Appendix D. Parameter Choice

As described in Section 4.3, we choose the regularization parameters for the numerical experiments of every considered method empirically by performing multiple experiments for different parameters and choosing the ones which lead to stable experiments and the best VDn. For the separated methods, we also follow, partially, the recommendations of the corresponding works. In the remaining sections, we describe the choice of these and other hyperparameters in more detail.

#### Appendix D.1. Separated Methods

In this section, we describe the choice of the main hyperparameters of all separated methods. Table A1 shows the selected parameters for the numerical experiments in Section 4, along with the used stopping criteria (stopCrit) and initialization methods (initMethod).

<table>
<thead>
<tr>
<th>Method</th>
<th>( \tau )</th>
<th>stopCrit</th>
<th>( i_{\text{max}} )</th>
<th>initMethod</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means-TV</td>
<td>1</td>
<td>Cluster assignment</td>
<td>-</td>
<td>K-means++</td>
</tr>
<tr>
<td>ONMF-TV-Choi</td>
<td>( 2 \times 10^{-2} )</td>
<td>maxIt</td>
<td>( 6 \times 10^2 )</td>
<td>SVD</td>
</tr>
<tr>
<td>ONMF-TV-Ding</td>
<td>( 2 \times 10^{-2} )</td>
<td>maxIt</td>
<td>( 8 \times 10^2 )</td>
<td>K-means++</td>
</tr>
<tr>
<td>ONMF-TV-Pompili1</td>
<td>1</td>
<td>Cluster assignment</td>
<td>-</td>
<td>K-means++</td>
</tr>
<tr>
<td>ONMF-TV-Pompili2</td>
<td>( 4 \times 10^{-2} )</td>
<td>Pompili Internal</td>
<td>-</td>
<td>SVD</td>
</tr>
<tr>
<td>ONMF-TV-Kimura</td>
<td>( 2 \times 10^{-2} )</td>
<td>maxIt</td>
<td>( 7 \times 10^2 )</td>
<td>SVD</td>
</tr>
<tr>
<td>ONMF-TV-Li</td>
<td>( 3 \times 10^{-2} )</td>
<td>maxIt</td>
<td>( 2 \times 10^2 )</td>
<td>SVD</td>
</tr>
</tbody>
</table>
All separated methods are initialized either based on the SVD approach described in Section 4.3 or via K-means++. Both methods were tried out for every separated method in the numerical experiments, and one of them was chosen based on the stability and quality measures of the results. For the specific case of ONMF-TV-Pompili1, we use a mixture of the K-means++ method and the internal initialization procedure of Reference [16].

Furthermore, different stopping criteria were used. Besides the classical stopping criterion based on a maximal iteration number \( i_{\text{max}} \) (maxIt), the clustering algorithms K-means-TV and ONMF-TV-Pompili1 (i.e., Line 3 in Algorithm 1) are stopped until the cluster assignments in the cluster membership matrix \( U \) do not change anymore. For the special case of ONMF-TV-Pompili2, the algorithm stops until the current iterates are “sufficiently” nonnegative (see Reference [16]).

Finally, the TV denoising algorithm in Line 4 of all separated methods is based on a Fast Iterative Shrinkage-Thresholding Algorithm (FISTA; see Reference [65]) with a maximal iteration number of 100.

Appendix D.2. Combined Methods

In this section, we describe the choice of the main hyperparameters of all combined methods. As in the previous section, Table A2 shows the selected parameters for the numerical experiments in Section 4, along with the used stopping criteria (stopCrit) and initialization methods (initMethod).

<table>
<thead>
<tr>
<th>Method</th>
<th>( \sigma_1 )</th>
<th>( \sigma_2 )</th>
<th>( \tau )</th>
<th>( \epsilon_{TV} )</th>
<th>( s_r )</th>
<th>stopCrit</th>
<th>( i_{\text{max}} )</th>
<th>initMethod</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONMF-TV-MUL1</td>
<td>0.5</td>
<td>0.5</td>
<td>( 5 \times 10^{-3} )</td>
<td>( \sqrt{1 \times 10^{-5}} )</td>
<td>–</td>
<td>maxIt</td>
<td>( 8 \times 10^2 )</td>
<td>SVD</td>
</tr>
<tr>
<td>ONMF-TV-MUL2</td>
<td>1</td>
<td>( 1 \times 10^{-3} )</td>
<td>( \sqrt{1 \times 10^{-5}} )</td>
<td>–</td>
<td>maxIt</td>
<td>( 7 \times 10^2 )</td>
<td>SVD</td>
<td></td>
</tr>
<tr>
<td>ONMF-TV-PALM</td>
<td>0.1</td>
<td>0.1</td>
<td>( 1 \times 10^{-1} )</td>
<td>–</td>
<td>maxIt</td>
<td>( 4 \times 10^2 )</td>
<td>SVD</td>
<td></td>
</tr>
<tr>
<td>ONMF-TV-iPALM</td>
<td>0.1</td>
<td>0.1</td>
<td>( 1 \times 10^{-1} )</td>
<td>–</td>
<td>–</td>
<td>( 3 \times 10^2 )</td>
<td>SVD</td>
<td></td>
</tr>
<tr>
<td>ONMF-TV-SPRING</td>
<td>0.1</td>
<td>0.1</td>
<td>( 1 \times 10^{-4} )</td>
<td>–</td>
<td>40</td>
<td>maxIt</td>
<td>( 1 \times 10^2 )</td>
<td>SVD</td>
</tr>
</tbody>
</table>

All combined methods are initialized based on the SVD approach described in Section 4.3 and are stopped until \( i_{\text{max}} \) iterations are reached (maxIt). Regarding the initialization methods, kmeans++ and the SVD method were tried out, and one of them was finally chosen based on the quality of the results as it is the case for the separated methods. For ONMF-TV-SPRING, we choose \( s_r = 40 \), according to Reference [50].

References


Bioimage informatics

Supervised non-negative matrix factorization methods for MALDI imaging applications

Johannes Leuschner, Maximilian Schmidt, Pascal Fernsel, Delf Lachmund, Tobias Boskamp* and Peter Maass

Department of Mathematics, Center for Industrial Mathematics, University of Bremen, Bremen 28359, Germany

*To whom correspondence should be addressed.

Associate Editor: Robert Murphy

Received and revised on May 25, 2018; editorial decision on October 28, 2018; accepted on November 2, 2018

Abstract

Motivation: Non-negative matrix factorization (NMF) is a common tool for obtaining low-rank approximations of non-negative data matrices and has been widely used in machine learning, e.g. for supporting feature extraction in high-dimensional classification tasks. In its classical form, NMF is an unsupervised method, i.e. the class labels of the training data are not used when computing the NMF. However, incorporating the classification labels into the NMF algorithms allows to specifically guide them toward the extraction of data patterns relevant for discriminating the respective classes. This approach is particularly suited for the analysis of mass spectrometry imaging (MSI) data in clinical applications, such as tumor typing and classification, which are among the most challenging tasks in pathology. Thus, we investigate algorithms for extracting tumor-specific spectral patterns from MSI data by NMF methods.

Results: In this article, we incorporate a priori class labels into the NMF cost functional by adding appropriate supervised penalty terms. Numerical experiments on a MALDI imaging dataset confirm that the novel supervised NMF methods lead to significantly better classification accuracy and stability as compared with other standard approaches.

Availability and implementation: https://gitlab.informatik.uni-bremen.de/digipath/Supervised_NMF_Methods_for_MALDI.git

Contact: tboskamp@uni-bremen.de

Supplementary information: Supplementary data are available at Bioinformatics online.

1 Introduction

The pathological diagnosis of a tumor found in a tissue specimen, including the determination of the tumor origin and genetic subtype, is crucial for individualized therapy decision and accurate prognosis. This task, often referred to as tumor typing, is conventionally based on microscopic examination of stained tissue sections, as well as molecular or genetic tests (Klauschen et al., 2015; Kruglyak et al., 2014), and often is subtle and requires extensive training and experience.

Naturally, since the early stages of digital pathology, large efforts have been devoted to support expert diagnosis by statistical and computational methods (Pantanoitz et al., 2010). Moreover, mass spectrometry imaging (MSI), which has been established over the last decade as a routine methodology in analytical chemistry and proteomics research, has demonstrated a high potential for a variety of tasks in clinical pathology (Kriegsmann et al., 2015; Schwamborn and Caprioli, 2010). However, the high complexity and large data volumes of MSI experiments demand for appropriate, dedicated computational analysis tools.

Recent reports on applications of MSI to tumor typing are the starting point for our present research (Casadonte et al., 2014; Kriegsmann et al., 2015; Veselkov et al., 2014). The proposed methods primarily rely on the extraction of spectral features (single peaks or more complex spectral patterns) from large sets of training data. In the mentioned sources, both statistical tests for detecting discriminative spectral features, as well as computational methods, such as
Supervised NMF for MALDI imaging

principal component analysis (PCA), probabilistic latent semantic analysis (PLSA) or non-negative matrix factorization (NMF) have been applied. These spectral features then form the basis for constructing a subsequent classification scheme (LDA, logistic regression, etc.). In particular, spectral patterns based on NMF decomposition have been demonstrated to result in competitive or even improved classification schemes for different tumor typing tasks (Bokskamp et al., 2016).

These results on NMF-based tumor typing rely on a first unsupervised step for extracting spectral patterns followed by a subsequent supervised training of a classification model. In this article, however, we aim at (i) motivating and analyzing classification schemes based on supervised NMF, and (ii) evaluating the potential of such methods for tumor typing using matrix-assisted laser desorption/ionization (MALDI) MSI.

Concerning the first aim, we will combine the NMF data decomposition with the construction of the classification scheme in a unified approach. We will do this in the context of regularized NMF decompositions, where the classification error for either logistic regression or linear discriminant analysis (LDA) is added as a separate penalty term.

Concerning the second aim, we will evaluate the derived classification schemes using MALDI MSI data from a collection of tissue microarrays (TMAs) of different types of lung cancer tissue samples. Due to the comparatively large number of patients (N = 304), this challenging dataset exhibits a large biological variation, as well as apparent technical variation between the different measurements. According to our main hypothesis, we expect that the spectral patterns determined by the supervised NMF methods are more relevant for the respective classification task, as compared with the unsupervised NMF approach, and hence result in a higher classification accuracy.

The article is organized as follows: Section 2 describes a general and well-established approach to regularized NMF functionals and their use in classification contexts. In Section 3, we introduce our proposed supervised NMF models and corresponding algorithms. In Section 4, we present the results of extensive numerical experiments evaluating the different variants of supervised NMF and classification schemes. The final section is devoted to a discussion of the results and an outlook on future research directions.

2 Non-negative matrix factorization

Matrix factorization methods address the task of computing a low-rank approximation to a given, typically large data matrix \( Y \) of dimension \( n \times m \). In matrix notation, this requires to determine matrices \( K \) and \( X \) of dimensions \( n \times p \) and \( p \times m \), respectively, such that \( p \ll m, n \) and \( Y \approx KX \). Such methods are the basis for a large variety of applications including compression, feature extraction or basis learning (Golub and Van Loan, 1996; Lee and Seung, 2001). If the data are non-negative, as is the case for MALDI MSI, it is often desirable to request that \( K \) and \( X \) are also non-negative, leading to a NMF.

In a standard situation, the data matrix \( Y \) combines \( n \) data vectors of length \( m \) (rows of data matrix \( Y \)), which are related to \( n \) entities under consideration (e.g. MALDI spectra). Computing an NMF helps to determine \( p \) characteristic non-negative basis vectors that allow to approximate the full data matrix \( Y \) (basis learning). Computing such an NMF decomposition based only on the data matrix \( Y \) is usually only a first step in a more complex processing pipeline for analyzing the data.

In this article we assume that the individual entities are labeled, e.g. ‘tumor’ or ‘non-tumor’ in a tissue typing application. In a classical two-step classification pipeline, the computation of an NMF decomposition is followed by constructing a linear or non-linear classification model (e.g. LDA or logistic regression). The label information is only used in the classification step, which is why this step is called a ‘supervised’ learning step, whereas the NMF decomposition is performed ‘unsupervised’.

In the following, we will first motivate the application of NMF methods for MSI-based tumor typing, summarize the basic known results on unsupervised NMF methods and corresponding algorithms, and outline the role of NMF in typical two-step classification schemes. The supervised NMF approach will then be described in Section 3.

2.1 Non-negative matrix factorization for tumor typing

A typical MALDI MSI experiment results in a data matrix \( Y \) consisting of measured spectra for different spatial locations of a tissue section or TMA (Fig. 1). The data matrix consists of data vectors \( Y_{i*} \in \mathbb{R}^m_+ \), \( i = 1, \ldots, n \), each one representing the spectrum measured at the spatial position with index \( i \) (see Supplementary Appendix C, for details).

The application of NMF methods for tumor typing is motivated by the assumption that only a comparatively small number \( p \) of metabolic processes or protein structures are represented in the dataset. It is hence feasible to assume that \( p \) spectral patterns \( X_{k*} \in \mathbb{R}^m_+, k = 1, \ldots, p \) with \( p < \min (n, m) \) are sufficient for approximating the full dataset, and that there exist coefficients \( K_{k*} \in \mathbb{R}_{\geq 0} \) such that \( Y_{i*} \approx \sum_{k=1}^{p} K_{k*} X_{k*} \). This results in a low-rank approximation of the data, i.e. \( Y \approx KX \). Even if not measured directly, the pseudo spectra \( X_{k*} \) (the rows of \( X \)) can be interpreted as mass spectra, the spatial distribution of which is given by the respective pseudo channels \( K_{k*} \) (columns of \( K \)). Hence, the non-negativity of \( K \)

![Fig. 1. Schematic MALDI MSI workflow. Tissue sections (a) are subjected to sample preparation including deparaffination, antigen retrieval, on-tissue tryptic digestion and matrix application (b). Prepared tissue sections are inserted into the MALDI MSI instrument (c) and mass spectra (d) are acquired. When fixing single m/z values, the intensities in the measurement area can be visualized as m/z images (e), reflecting the molecular distribution of peptides with corresponding masses.](https://academic.oup.com/bioinformatics/article/35/11/1940/5161075)
and X supports the biological interpretability, making NMF methods an ideal tool for analyzing MSI datasets and for extracting characteristic spectral patterns as a basis for classification and tumor typing.

### 2.2 Tikhonov functionals for NMF

The NMF problem, i.e., computing factors K, X such that Y ≈ KX, is commonly formulated as a minimization problem with a suitable discrepancy term. In this article, the discrepancy measure will always be the standard Frobenius norm, i.e., the sum of squared matrix coefficients, denoted by $||·||_F^2$.

To deal with the non-uniqueness of the NMF, with numerical instabilities and scaling issues, as well as to equip the matrices K and X with additional desirable properties, we incorporate additional penalty terms $\phi_i$ into the cost function. Thus, the general minimization problems is of the form

$$\min_{K,X} \frac{1}{2} ||Y - KX||^2_F + \sum_i z_i \phi_i(K, X).$$  

(1)

with regularization parameters $z_i \in \mathbb{R}_{\geq 0}$ controlling the influence of the penalty terms.

The variety of potentially useful penalty terms $\phi$ is huge and needs to be guided by the application in mind. Here, we introduce $\ell_2$-regularization terms for K and X (Frobenius norm) as the simplest form of Tikhonov regularization, which avoids scaling issues and results in a better condition of the minimization problem. Moreover, an $\ell_1$-penalty term on $X$ is introduced, increasing the sparsity of $X$, thus resulting in sparser pseudo spectra with more characteristic m/z peaks.

More precisely, this leads to the unsupervised NMF model

$$\min_{K,X} \frac{1}{2} ||Y - KX||^2_F + \lambda ||X||_1 + \mu ||K||_F^2 + \nu ||X||_F^2$$  

(2)

with suitable regularization parameters $\lambda, \mu, \nu \geq 0$, which will be abbreviated as FR (Frobenius, regularized).

Furthermore, we intend to introduce orthogonality constraints on the pseudo spectra $X_a$, such that $X^T \approx 1$, which results in less correlated and—together with the non-negativity of $X$—sparser pseudo spectra.

However, instead of directly adding the fourth-order penalty term $||I - XX^T||_F^2$ in the cost functional, we introduce a third auxiliary variable $W \in \mathbb{R}^{m \times m}$ and split the constraint in two penalty terms. Thus, the problem is transformed to

$$\min_{K,X,W} \frac{1}{2} ||Y - KX||^2_F + \lambda ||X||_1 + \frac{\mu}{2} ||K||_F^2 + \frac{\nu}{2} ||X||_F^2 + \sigma_1 ||I - XX^T||_F^2 + \sigma_2 ||W - X||_F^2$$  

(3)

with additional regularization parameters $\sigma_{1,2} \geq 0$. This NMF model is abbreviated as FRO (Frobenius, regularized, orthogonal). The special case without $\ell_1$-regularization ($\lambda, \mu, \nu = 0$) will be referred to as FO (Frobenius, orthogonal).

### 2.3 Algorithms

All models presented in the previous section are formulated as minimization problems. These models include multiple matrix variables, i.e., $K, X$, as well as $W$ for the FO and FRO models. The cost functionals are convex as long as only a single matrix variable is varied, but they are non-convex in the Cartesian product spaces for $(K, X)$ and $(K, X, W)$, respectively.

We follow the classical approach of majorize-minimization (MM) algorithms, (Lange, 2016) leading to alternating matrix variable updates. The key idea is to shift the minimization to surrogate functions that majorize the original cost function locally and are easier to minimize. Suitable surrogate functions for NMF-Tikhonov models as well as the resulting update rules are summarized in Supplementary Appendix A.

Rescaling the rows of $X$ further improves the stability of the algorithm as well as the interpretability/comparability of the resulting spectral patterns $X_a$. Incorporating rescaling is equivalent to multiplying $X$ respectively $K$, with a diagonal matrix $D = \text{diag}((|X_a|, k = 1, \ldots, p))$ from the left, respectively with $D^{-1}$ from the right. While rescaling is often found to be necessary with the FR model, it is less relevant with the FRO model, as the orthogonalization of $X$ in the NMF decomposition implies $\ell_2$-normalization.

### 2.4 Classification methods based on NMF decompositions

Typical classification methods are based on two steps: In the first step, the original data vectors are transformed into some feature vectors, usually of much lower dimension. In the second step, the actual classification scheme is applied to the feature data, resulting in a binary or multi-class assignment.

Depending on the nature of the original data, different approaches for defining features can be suitable. In cases where the data vectors represent measurements of some physical quantity, correlations with suitably defined or computed basic pattern vectors are widely used. In the context of MALDI MSI data analysis, we consider features obtained by correlating a spectrum $y$ (row vector of length $n$) with a set of pseudo spectra constructed by NMF methods, i.e. the row vectors of $X$. Hence, the data vector $y$ is mapped to the feature vector $f$ (row vector of length $p$) by $f = yX^T$.

For generating the classification model, we assume that a sufficiently large set of training data $Y$ is given together with corresponding class labels $u$ (ground truth), such that $u(i) \in \{0, 1\}$ denotes the class label of the $i$th data vector $Y_i$. Furthermore, the feature vectors corresponding to the full set of training data $Y$ are computed by $F := YX^T$, yielding a matrix $F \in \mathbb{R}^{n \times p}$ of row feature vectors. Note that we only consider the case of binary classification here, although the general concepts are easily extended to the multi-class case.

In this article, we consider two different binary regression models as classification methods: LDA and logistic regression. Both models are optimized by solving a minimization problem $\min \Phi(u, F, \beta)$, where $\Phi$ denotes some cost function measuring the discrepancy between the true class labels and the assignments of the classification model, and $\beta \in \mathbb{R}^p$ denotes the set of model parameters (for details, see Supplementary Appendix B, as well as Chapter 4 in Bishop, 2006).

The overall approach of constructing and using a two-step classification method is visualized in Figure 2. For constructing the model, an annotated set of training data is used. An unsupervised NMF is computed from the training data, yielding the pseudo spectra that define the feature map. Feature vectors are then computed by applying the feature map to the training data, and are used together with the label annotations for optimizing the classification model. Once the feature map and the classification model are computed, the classification of a new data vector consists of applying the feature map to obtain a feature vector as the input for the classification model.

Combining the different unsupervised NMF decomposition schemes (FR, FO, FRO) described in this section with either LDA or logistic regression results in six different classification schemes:
FR lda, FR log, FO lda, FO log, FRO lda and FRO log. In addition, the computation of the NMF decompositions may or may not incorporate a rescaling of X and K in each iteration, leading to 12 different classification schemes. A characteristic subset of these methods will be evaluated in Section 4 in comparison to the supervised approaches presented in the next section.

3 Supervised NMF methods

3.1 Motivation

The NMF methods described in the previous section yield decompositions of the data matrix Y that provide a good approximation using only a small number p of basis patterns (rows of X) and channels (columns of K). Thus, we can expect X to consist of those spectral patterns that were found to be, in some sense, most dominant in the data. In the context of classification problems, however, we are not primarily interested in the most dominant components or in an accurate data approximation.

Instead, we wish to extract those spectral patterns that allow to well discriminate between spectra acquired from different tissue phenotypes, such as, for example tumor and normal tissue. The spectral features reflecting these different tissue types, however, are often subtle and much less expressed than the most dominant spectral components. Hence, they may well be considered irrelevant by a standard NMF decomposition, causing them to be suppressed in the resulting feature map Y→YX^T and resulting in a decreased classification accuracy.

In the following, we will propose an extension of the standard NMF methods that allows to incorporate the a-priori label information associated with the different tissue types into the NMF minimization problem, guiding the NMF algorithm to spectral basis patterns that are most informative and relevant with respect to the classification task. The term ‘supervised NMF’ will be used to distinguish between these extended NMF methods and the standard, unsupervised methods.

3.2 Models and algorithms

In this section, we discuss the different supervised NMF models which are used in this article and describe shortly the respective update rules and their derivation.

The supervised NMF methods differ from the standard unsupervised approach in that the classification task is done in parallel to the feature extraction. This is realized by integrating the cost function of the classification method into the NMF cost functional. Here, we consider either LDA or logistic regression as classification scheme (see Section 2.4 and Supplementary Appendix B). Using the notation of the previous sections and the cost functionals in Equations (5) and (7) in Supplementary Appendix B, leads to the basic supervised NMF models Flda

\[
\min_{X,K,b} \frac{1}{2} ||Y - KX||_F^2 + \frac{\gamma}{2} ||a - YX^T||_F^2 \quad \text{Flda} \quad (4)
\]

and Flog

\[
\min_{X,K,b} \frac{1}{2} ||Y - KX||_F^2 + \frac{1}{n} \sum_{i=1}^{n} \log (1 + e^{Y^{T}X_i} - u_i)^2 ||YX^T||_F^2 \quad \text{Flog} \quad (5)
\]

with the regularization parameter \(\gamma \geq 0\). Note that Flda (4) requires the non-negativity constraint on b since we assume in this model that a is a superposition of the correlation images YX_k^T. However, from model Flog (5) and Equation (6) in Supplementary Appendix B.2, it is necessary for b to assume negative values in the logistic regression case in order to be able to model probabilities smaller than 0.5.

Analogously to Section 2.2, we also consider the combination of the supervised Flda NMF model with \(\ell_1\)- and \(\ell_2\)-regularization terms for K and X, as well as terms enforcing the orthogonality of X. These additional penalty terms have the same structure as the corresponding terms in the unsupervised case. Thus, iterative algorithms can again be determined following the MM scheme and constructing suitable surrogate functionals (see Supplementary Appendix A). This leads to the regularized supervised NMF models FRlda and FROlda with the following update rules:

\[
K \leftarrow K - \frac{XY^T}{KX X^T + \mu K} \mu K
\]

\[
D \leftarrow K'X + \sigma_1 X W^T + \sigma_2 X + \gamma \beta Y X^T Y + \nu X + \lambda
\]

\[
X \leftarrow X - \frac{K'Y + (\sigma_1 + \sigma_2) W + \gamma \mu Y}{D}
\]

\[
W \leftarrow W - \frac{(\sigma_1 + \sigma_2) X}{W (\sigma_1 X + \sigma_2 I)}
\]

\[
\beta \leftarrow \beta - \frac{YX^T \mu}{YX^TY \beta}
\]

For FRlda, let \(\sigma_1 = \sigma_2 = 0\) and ignore the rule for W. Note that the above rules are multiplicative and preserve non-negativity as long as the matrices are initialized non-negative.

As regards model Flog (5), minimization with respect to K leads to the same cost functional as in Flda (4), and therefore to the same update rules for K. Due to the different structure of the logistic regression term, however, we need to follow a different approach for the updates of X and \(\beta\). More specifically, we use a special variant of the stochastic gradient descent approach method ADADELTA.

![Fig. 2. Standard process of NMF-based data classification](https://academic.oup.com/bioinformatics/article/35/11/1940/5161075)
This adaptive method uses the information of previously computed gradients to evaluate the next step size and introduces two parameters to be chosen before performing the iterations: one parameter representing a decay rate and the second one for stability reasons. The required gradients of the cost functional in model Flog (5) with respect to $X$ and $\beta$ can be easily calculated analytically for this approach, which leads to ADADELTA updates $\Delta X$ and $\Delta \beta$ (see Zeiler, 2012 for details). To ensure the non-negativity of $X$, a projection step (denoted by proj$(X)$) is included that replaces all entries $x_{ij} \leq 0$ with a small positive constant. This finally results in the Flog update rules:

$$
K \leftarrow K - \frac{Y X^T}{K X X^T} \\
X \leftarrow \text{proj}(X + \Delta X) \\
\beta \leftarrow \beta + \Delta \beta
$$

Note that we do not add further regularization terms to the Flog functional. This has two reasons: First, adding additional regularization terms with respect to $K$ could be handled by the same update rules as for Flda, whereas adding additional regularization with respect to $X$ would require adapting the minimization scheme for functionals mixing logarithmic and quadratic penalty terms, which is not the focus of this article. Second, numerical tests indicate that the logistic regression term added to the Frobenius discrepancy term already yields sufficient regularization. In fact, the Flog algorithm will produce the best classification results, as seen in Section 4.

### 3.3 Combination with classification

We now assume that a supervised NMF decomposition using a variant of Flda or Flog as described in the previous section has been computed. The next step is the construction of a classifier based on these NMF vectors, i.e. a function mapping new data $\tilde{Y} \in \mathbb{R}^{n \times m}$ to a class prediction $\tilde{u} \in \{0, 1\}^k$. For that purpose we discuss two approaches.

#### Integrated approach

Directly utilizing the coefficient vector $\beta$ from the supervised NMF model (Flda (4) or Flog (5), respectively) yields the first version, which we call integrated classifier. It will be indicated by an _int suffix in the model names (see below). For the Flda variants one computes $Y X^T \beta \in \mathbb{R}^n$, which assigns a real value to each new spectrum in $\hat{Y}$. Next, entries below some threshold $t$ are mapped to class 0, all other entries to class 1. The threshold $t$ is determined from the training data by calculating $Y X^T \beta \in \mathbb{R}^n$ and choosing $t$ such that the target performance measure (see Section 4.1) is optimized.

For the Flog variants, the regression values, computed by (8) in Supplementary Appendix B.2, can directly be interpreted as probabilities for class 1. It is hence natural to apply the threshold 0.5, i.e. entries with a probability less than 50% are predicted to belong to class 0, entries with a higher probability to class 1.

#### Optimized approach

The second approach performs the classifier training in a separate step after solving the supervised NMF model. We keep the characteristic patterns $X$ determined by the supervised NMF, but we ignore the coefficient vector $\beta$. Instead we use a subsequent optimization for training an independent classifier (LDA or logistic regression) using the feature data $Y X^T$ and the class labels $\tilde{u}$. New MALDI data $\tilde{Y}$ is then classified by applying the classifier on the feature data $Y X^T$. These variants will be indicated by an _lda or _log suffix.

Combining the different supervised NMF models with the above integrated or optimized classification methods leads to a large variety of classification schemes. Our main aim is to demonstrate the effect of including supervised terms in the NMF construction directly as opposed to the classical two-step approach (unsupervised NMF, classifier). To this end, we have selected the following models for comparison and evaluation: FRlda_lda, FRlda_log, FRlda_int, FROlda_lda, FROlda_log, FROlda_int, Flog_lda, Flog_log and Flog_int.

### 4 Results and discussion

In the following, we present the numerical results obtained by applying the classification schemes introduced in the previous sections to the MALDI MSI dataset described in Supplementary Appendix C. This dataset contains 4667 spectra of length 1699, it is obtained from eight TMAs of lung tumor tissues, denoted by L1 to L8. Each TMA consists of a collection of adenocarcinoma (ADC) and squamous cell carcinoma (SqCC) biopsies, proportions of both tissue types were roughly similar in all TMAs. MALDI MSI data were acquired in separate experiments for each TMA. The classification task consists in distinguishing between the tumor types ADC and SqCC, corresponding to class labels $u = 0$ and $u = 1$, respectively.

A comparison of the performance achieved on this task by the different classification schemes are given below. Moreover, the characteristics of the different NMF decompositions are investigated and compared with biological interpretations available from previous work.

#### 4.1 Classification performance

In order to obtain a realistic estimation of a classification model’s performance, the available data have to be divided into a training and a test set (see Fig. 2). The results presented below were obtained by applying a standard cross-validation (CV) scheme known as $k$-fold CV. Since we wanted to evaluate the robustness of the classification methods toward technical variation between different MALDI MSI experiments, we chose to perform an 8-fold CV on TMA level.

More specifically, from the total set of TMAs L1 to L8, eight different training subsets were formed, each one consisting of seven TMAs. In each of the eight CV folds, a classification model was trained on seven training TMAs and tested on the respective remaining test TMA not included in the training set. Thus, each CV fold yields a classification result (prediction) for one of the TMAs, and the union of all predictions represents the complete prediction obtained using the respective classification scheme. Note that this CV scheme covers both the feature extraction and the classification step, as the NMF decompositions are computed on the training data only.

The accuracy of a classification result is evaluated by computing the sensitivities for both classes, i.e. the number of correctly classified spectra in each class divided by the total number of spectra in this class, and taking the average of both sensitivities. This metric, known as balanced accuracy, has the advantage of being independent of the relative proportions of the classes within the respective test data (Sokolova and Lapalme, 2009).

In total, 13 different classification schemes were evaluated. Since the number of features $p$ has a strong influence on the accuracy of many models, we varied this hyperparameter from 10 to 100 in steps of 10. The balanced accuracies for a subset of the most relevant methods is shown in Figure 3 (left), results for all methods as well as details on the regularization parameters used are presented in Supplementary Appendix D.
As can be observed, the methods Flog_int and Flog_log based on the supervised NMF model Flog (5) achieve the highest performance of approximately 90%, independent of the number of features $p$. They are followed by the FR lda method achieving a maximum performance close to 85%. Most of the other methods achieve balanced accuracy values below 80%. Furthermore, the following detailed observations can be made (see Supplementary Appendix D):

- Among the unsupervised methods, FR is superior to the FRO model. Both achieve better results when combined with LDA classification as compared with logistic regression.
- While the supervised methods Flog_int and Flog_log outperform all others, Flog lda performs significantly worse.
- Similarly, FR(O)lda_int and FR(O)lda_log perform better than FR(O)lda lda, although the effect is less expressed.
- While FRlda_int and FRO_int show very similar performance, FR0lda lda and FR0lda_log perform worse than their FRlda lda/log counterparts.

In addition to the overall classification performance, we investigated the performance variation between the eight CV folds for each classification schemes. In particular where two methods show similar overall performance, it is of interest to compare the respective performance variation, as a lower variation indicates a method’s higher robustness toward biological and technical variability. As can be seen in Figure 3 (right), performance variation of Flog_int is significantly smaller than that of Flog_log, although both show almost the same stable overall performance. On the other hand, no significant differences in performance variation are noted between FRlda_int and FRlda_log.

As can be observed, the methods Flog_int and Flog_log based on the supervised NMF model Flog (5) achieve the highest performance of approximately 90%, independent of the number of features $p$. They are followed by the FR lda method achieving a maximum performance close to 85%. Most of the other methods achieve balanced accuracy values below 80%. Furthermore, the following detailed observations can be made (see Supplementary Appendix D):

- Among the unsupervised methods, FR is superior to the FRO model. Both achieve better results when combined with LDA classification as compared with logistic regression.
- While the supervised methods Flog_int and Flog_log outperform all others, Flog lda performs significantly worse.
- Similarly, FR(O)lda_int and FR(O)lda_log perform better than FR(O)lda lda, although the effect is less expressed.
- While FRlda_int and FRO_int show very similar performance, FR0lda lda and FR0lda_log perform worse than their FRlda lda/log counterparts.

In addition to the overall classification performance, we investigated the performance variation between the eight CV folds for each classification schemes. In particular where two methods show similar overall performance, it is of interest to compare the respective performance variation, as a lower variation indicates a method’s higher robustness toward biological and technical variability. As can be seen in Figure 3 (right), performance variation of Flog_int is significantly smaller than that of Flog_log, although both show almost the same stable overall performance. On the other hand, no significant differences in performance variation are noted between FRlda_int and FRlda_log.

4.2 Characteristic spectral patterns
The above results demonstrate that classification schemes based on the supervised NMF model Flog achieve the highest balanced accuracy values and lowest variation among all investigated methods. Most notably, these results are achieved with even a small number of features and are stable across the whole range of feature counts. This motivates the assumption that in these methods only a small number of basis vectors $X_i$ is actually relevant, and that there is only little variation in these vectors between different training sets.

To reduce complexity, we replace the 8-fold CV from the previous section by a 2-fold CV, in which the full dataset is split in two subsets, $A = \{L1 \ldots L4\}$ and $B = \{L5 \ldots L8\}$. Balanced accuracy
results for some of the classification schemes are shown in Figure 4. Note that in this scenario, each classifier is trained on only four out of eight TMAs, which explains the performance decrease as compared with 8-fold CV. Moreover, we extended the variation of the number of features to include $p \in \{1 \ldots 10\}$.

As discussed in Section 3.2, the vector of regression weights $\beta$ in Flog [5] reflects the level of influence of the spectral patterns on the logistic regression term in the NMF model. More specifically, the $i$-th entry in $\beta$ is related to the $i$-th pseudo spectrum $X_{i:}$. Thus, we can interpret the values $\beta_i$ as the relevance of the respective pseudo spectra for the classification model.

Figure 5 shows how the number of active weights, i.e. weights significantly different from zero, changes with the number of features $p$. While for Flog the number of active weights remains almost unchanged at a certain level, it continues to increase with $p$ in the traditional approach FR_log and is often very close to the total number of weights. Obviously, the supervised NMF model tends to concentrate the information relevant for the classification task on a few pseudo spectra. This is not possible with the unsupervised NMF models, on the other hand, as these are independent of the subsequent classification.

A closer investigation of the pseudo spectra $\vec{X}$ generated by the Flog model reveals a high correlation among the basis vectors. Moreover, these patterns exhibit little dependency on the number of features and thus can be interpreted as being characteristic for the two classes ADC and SqCC.

To illustrate this, Figure 6 shows the weighted linear combination of pseudo spectra, given by $x_y = X^T \beta_y$, which is the discriminatory pattern underlying the Flog_int classification scheme. In fact, the dominant features in this pattern align well with results published previously on the discrimination of ADC- and SqCC tissue of the lung (see Supplementary Appendix D, for further details).

5 Conclusion
In the past years, NMF has been established as a valuable tool for generating low-rank approximations of large datasets. Such methods have recently been applied to classification problems, where the NMF serves as a feature extraction step prior to the training of a classification model. In the context of such applications, we have presented an extension of the classical NMF framework by incorporating the class information on the training data into the NMF feature extraction step. Formally, this has been achieved in a natural way by introducing additional penalty terms into the NMF objective functional, thus being able to solve the modified problem with algorithms that are similar to those for the original problem. Moreover, this approach allows to unify the feature extraction and classifier training steps, reducing computation time and algorithmic complexity.

We have evaluated several variants of this supervised NMF approach on a challenging classification task related to MALDI MSI and its application to tumor typing in pathology. The comparison of the novel methods based on a supervised NMF decomposition with more conventional NMF-based classification schemes reveals an improved classification accuracy in some of the investigated methods. In particular, methods based on a logarithmic regression-type extension of the NMF decomposition significantly outperform all other methods in our experiments.

An in-depth analysis of the pseudo spectra and discriminatory patterns generated by the supervised NMF decompositions yields a high stability of the method with respect to the training subset and the feature space dimension. Moreover, the generated patterns are amenable to a biological interpretation, thus allowing to confirm the hypothesized discriminative markers by complementary analysis techniques. In our application, we were able to extract discriminative markers that nicely match mass spectrometric markers identified by other researchers.

In future work, we plan to extend the application of this method to multi-class classification problems, as well as developing methods for making an optimal choice of the method’s hyperparameters.

Acknowledgements
Tissue samples were kindly provided by Dr M. Kriegsmann (Institute of Pathology, Heidelberg University Hospital), Dr A. Warth (Thoracic Pathology, Heidelberg University Hospital), Prof. Dr H. Dienemann (Thoracic Surgery, Heidelberg University Hospital) and Prof. Dr W. Weichert (Institute of Pathology, Technical University of Munich) through the tissue bank of the National Center for Tumor Diseases (NCT, Heidelberg, Germany) in accordance with the regulations of the tissue bank and the approval of the ethics committee of Heidelberg University, MALDI MSI data was kindly provided by Dr. Rita Casadonte (Proteopath, Trier, Germany). The authors acknowledge that the used protocol for processing FFPE tissue,
as well as the method of MS-based differentiation of tissue states are subject to patents held by or exclusively licensed by Bruker Daltonik GmbH, Bremen, Germany.

**Funding**

The authors gratefully acknowledge the financial support from the German Federal Ministry of Education and Research (‘KMU-innovativ: Medizintechnik’ program, contract number 13GW0081) and the German Science Foundation within the framework of GRK 2224/1 2224 ‘Parameter Identification—Analysis, Algorithms, Applications’.

**Conflict of Interest:** none declared.

**References**


Appendix A  Algorithmic details

A.1  Surrogate functions

All models presented in this paper are formulated as minimization problems. Many models include multiple variables, e.g. the NMF models involve at least $K$ and $X$, and for the supervised NMF models $\beta$ is added. This constitutes a difficulty in finding optimal values, especially since the discussed cost functions in general are non-convex in the space given by the Cartesian product of the variable spaces. We approach this by separating over the variables within each minimization step of the iterative algorithms. Hereby all variables are updated alternatingly, aiming for a set of variable values minimizing the cost function.

MM algorithms (Majorize-Minimization algorithms), elaborated in Lange [2016], are a commonly used optimization strategy. The key idea is to shift the minimization to surrogate functions that majorize the original cost function locally and are desirably easier to minimize.

For a cost function $f$, a surrogate function $g_{a}(x)$ at point $a$ is characterized by fulfilling the two conditions

(i) \[ g_{a}(x) \geq f(x) \quad \forall x, \]

(ii) \[ g_{a}(a) = f(a). \]

By this it is easy to prove that iterating the update rule

\[ x^{(k+1)} := \arg \min \ g_{x^{(k)}}(x), \quad k = 0, 1, 2, \ldots \]  

(1)

guarantees a monotonic decrease of $f(x^{(k)})$. While this implies convergence of $f(x^{(k)})$, as stated in Lee and Seung [2001], it does not imply convergence of $x^{(k)}$, the minimizing variable itself. Also, this method possibly does not locate a global but only a local minimum. In practice, however, MM algorithms have been found to yield good solutions.

In order to use this strategy for our minimum problems, the challenge now is to find surrogate functions for the employed cost functions. There are some general approaches for obtaining surrogate functions, including applications of Jensen’s inequality, the Cauchy-Schwarz inequality, and the quadratic upper bound principle. We are going to utilize the latter.

A.2  Quadratic upper bound principle

As a starting point consider the second-order Taylor polynomial of $f(x)$ around $a$,

\[ T_{2,a}f(x) = f(a) + (x-a)^{T}\nabla f(a) + \frac{1}{2}(x-a)^{T}Hf(a)(x-a). \]

*tboskamp@uni-bremen.de
Replacing the Hessian matrix $H_f(a)$ by another matrix $M$, whose difference to $H_f(a)$ is positive semidefinite, i.e.

$$v^T(M - H_f(a))v \geq 0 \quad \forall v, a,$$

generates an upper bound for $T_{2,a}f(x)$,

$$g_a(x) := f(a) + (x - a)^T \nabla f(a) + \frac{1}{2}(x - a)^T M(x - a) \geq T_{2,a}f(x). \quad (2)$$

The surrogate function condition (i) $g_a(x) \geq f(x)$ hence is fulfilled if $T_{2,a}f(x) \geq f(x)$. Since all of our cost functions are polynomials of at most second degree, the Taylor polynomials $T_{2,a}f(x)$ are equal to the functions itself. As (ii) $g_a(a) = f(a)$ obviously holds, we can use (2) as a surrogate function template.

If $M$ is positive definite, then $g_a(x)$ is strictly convex and the calculation of its global minimum point, which is required by the update rule (1), results from solving $\nabla g_a(x) = 0$,

$$\nabla f(a) + M(x - a) = 0$$

$$\Leftrightarrow \quad x = a - M^{-1} \nabla f(a). \quad (3)$$

Note that the inverse $M^{-1}$ is clear to exist by the positive definiteness of $M$.

When choosing $M$, it is reasonable to require $M^{-1}$ to be straightforward to compute. Another quite important aim originates from the non-negativity constraint on $x$ in (3) imposed when updating $K$ or $X$. This can be easily satisfied if the update rule (3) is multiplicative, in the sense that it can be rewritten by only addition, multiplication and division of the used variables, and particularly not by subtraction. If this holds for all alternating update rules, the non-negativity of all variables is automatically maintained throughout the whole algorithm after choosing non-negative initial values. This avoids the need for projections into the valid spaces, which are commonly used in other optimization algorithms.

For our applications the choice of the diagonal matrix

$$M = \left[ \delta_{i,j} \left( \frac{H_f(a) a_i + \lambda}{a_i} \right)_{i,j} \right]_{i,j}, \quad (4)$$

whose inversion is naturally trivial, will lead to multiplicative update rules. This is owing to the special relation between the gradients $\nabla f(a)$ and the Hessians $H_f(a)$. Furthermore, in our cases $H_f$ has non-negative components, a property that ensures positive semidefiniteness of $M - H_f$ for the proposed choice of $M$ (Blondel et al. [2008]).
In (4) the parameter $\lambda$ covers the case of $\ell_1$-regularization on the updated variable. Nevertheless for $M$ to exist and to be positive definite, we need both the numerator and the denominator in (4) to be non-zero, which leads us to require not only non-negativity but rather strictly positiveness for $K$ and $X$. This is not sensible from the perspective of NMF or MALDI imaging, especially since we prefer sparse pseudo spectra, but is expedient to assume during the algorithm. We technically ensure the positiveness by adding a small value to each variable component after its regular update. When finally interpreting the results, a small threshold can be used to gain actual sparsity.

A.3 Update rules
To obtain the update rules for the required variables, firstly the Hessian is calculated and substituted in (4). This is then substituted in (3), which is rearranged conveniently. The resulting update rules for models $F$, $FRO$, $FROlda$, and $Flog$ are listed below.

**F**

$$K \leftarrow K \odot \frac{YX^\top}{KXX^\top}$$

$$X \leftarrow X \odot \frac{K^\top Y}{K^\top KX}$$

**FRO**

$$K \leftarrow K \odot \frac{YX^\top}{KXX^\top + \mu K}$$

$$X \leftarrow X \odot \frac{K^\top Y + (\sigma_1 + \sigma_2)W}{K^\top KX + \sigma_1 XW^\top W + \sigma_2 X + \nu X + \lambda}$$

$$W \leftarrow W \odot \frac{(\sigma_1 + \sigma_2)X}{W(\sigma_1 X^\top X + \sigma_2 I)}$$

**FROlda**

$$K \leftarrow K \odot \frac{YX^\top}{KXX^\top + \mu K}$$

$$X \leftarrow X \odot \frac{K^\top Y + (\sigma_1 + \sigma_2)W + \gamma \beta u^\top Y}{K^\top KX + \sigma_1 XW^\top W + \sigma_2 X + \gamma \beta \beta^\top XY^\top Y + \lambda}$$

$$W \leftarrow W \odot \frac{(\sigma_1 + \sigma_2)X}{W(\sigma_1 X^\top X + \sigma_2 I)}$$

$$\beta \leftarrow \beta \odot \frac{XY^\top u}{XY^\top Y + \sigma_2 I}$$

**Flog**

$$K \leftarrow K \odot \frac{YX^\top}{KXX^\top}$$

$$X \leftarrow \text{proj} \left( X + \Delta X \right) \quad \text{(Gradient based update, projection on } \mathbb{R}_+ \text{)}$$

$$\beta \leftarrow \beta + \Delta \beta \quad \text{(Gradient based update)}$$
Appendix B  Linear and logistic regression for binary classification

B.1 Linear discriminant analysis

A standard procedure for generating a classification model based on an NMF decomposition, as mentioned in Section 2.4 (main article), is to approximate given binary class labels $u \in \{0, 1\}^n$ by a linear combination of the correlations $Y_{Xk}^\top$ and leads to the matrix $F := YX^\top \in \mathbb{R}^n_{\geq 0}^p$, which contains row-wise the features vectors $Y_{1,k}X^\top$ of length $p$. Thus, coefficients $\beta \in \mathbb{R}^p$ have to be determined, such that $u \approx \sum_{k=1}^p \beta_k F_{:,k} = YX^\top \beta$. Due to the optimization method and the multiplicative update rules, these coefficients are automatically non-negative in the case of the LDA, so that $u$ is a superposition of the correlations $F_{:,k}$. To also allow the modeling of class labels equal to zero, we therefore omit a strict positive bias and consider the unbiased case in contrast to default LDA regression models.

The estimation of $\beta$ is typically done with a least squares method and leads to the minimization problem

$$\min_{\beta} \|u - YX^\top \beta\|_F^2. \quad (5)$$

The minimization using the available annotated training data yields a suitable parameter set $\hat{\beta}$ and a corresponding characteristic vector $\hat{x} = X^\top \hat{\beta}$. The actual classification of a new data set $y$ is now straightforward, it just requires to compute the scalar product $c = y \hat{x}$ and the resulting classification is obtained by a binary threshold on $c$.

One great advantage of the linear regression model is its simplicity and the manageability. However, the fundamental drawback of this method is the fact that $u$ is a binary coded target variable which is in contrast to the input data $F$, whose entries lie typically in $\mathbb{R}_{\geq 0}$. The linear regression model is commonly not the natural choice to approximate binary output data with a continuous input. We therefore consider logistic regression as an alternative approach.

B.2 Logistic regression

The logistic regression method represents a more natural approach to model binary output variables. It is a special case of the general logistic regression for two classes and is also known as the logit model. Here, we use the standard biased case of the logit approach and define accordingly $F := [1|YX^\top]$. The basic idea is to model the posterior probabilities $\pi_i = P(u_i = 1)$ for the occurrence of $u_i = 1$ by applying the standard logistic function $h(x) = (1 + e^{-x})^{-1} \in [0, 1]$ to the so called linear predictors $F_{i,\bullet} \beta$, yielding the regression model

$$\pi_i = h(F_{i,\bullet} \beta) = \frac{1}{1 + \exp(-F_{i,\bullet} \beta)}. \quad (6)$$

As mentioned in Section 3.2 (main article), we will also allow negative entries in $\beta$ to be able to model probabilities lower than 0.5.

The estimation of the parameter $\beta$ is typically done by applying the maximum likelihood method. Since the output variable $u_i$ underlies the Bernoulli distribution, it follows for the likelihood function $L(\beta) = \prod_{i=1}^n \pi_i^{u_i}(1 - \pi_i)^{1-u_i}$. This leads finally to the minimization of the negative log-likelihood $l^-$ to get an estimator for the parameter $\beta$, such that

$$\min_{\beta} l^-(\beta) = \min_{\beta} \sum_{i=1}^n \log(1 + \exp(F_{i,\bullet} \beta)) - u^T F \beta. \quad (7)$$

The prediction on new datasets $\hat{Y}$ is done by using (6) and calculating

$$\hat{\pi}_i = \frac{1}{1 + \exp(-[1|\hat{Y}X^\top]_{i,\bullet} \beta)}. \quad (8)$$

The main advantage over the linear regression method is that the values of the prediction can be directly interpreted as probabilities for the occurrence of tumorous regions in the tissue sample.
Figure C2: Schematic data structure. HE images of 8 TMAs (left), each containing multiple core biopsies of ADC or SqCC patients (middle). For each core several MALDI TOF spectra are collected (upper right). The close-up shows an isotopic pattern for a peptide at m/z 1790.9 (lower right).

Appendix C MALDI imaging and test data

Mass spectrometry imaging (MSI), in particular matrix-assisted laser desorption/ionization (MALDI) MSI, is a label-free technique for spatially resolved molecular analysis of biological tissue samples with a broad range of applications in pharmaceutical and biomedical research. With recent technological advances in acquisition speed and robustness, applications of MALDI MSI in pathological diagnostics became feasible, where this method can help to, for example, characterize tumor tissue, delineate tumor regions or identify tumor subtypes (Groseclose et al. [2008], Seeley and Caprioli [2011]).

In MALDI MSI, a pulsed laser beam is focused at a series of spots (pixels) covering a tissue sample with a spatial resolution ranging from 5 to 250 µm. Biomolecules over a wide mass range are desorbed from the tissue surface, ionized and fed into a mass spectrometer. The resulting dataset consists of a complete mass spectrum with up to 1 million spectral intensity values for each tissue spot, yielding extremely high-dimensional multispectral images for arbitrary ion masses.

The determination of spatial tissue regions with different biological characteristics requires the extraction of biologically meaningful information out of the spectral data, i.e. the identification of “spectral fingerprints” associated with different tissue phenotypes. A successful extraction of such biological information heavily depends on a careful sample preparation and data acquisition, including sample collection and fixation, several preparation steps, as well as matrix deposition and the actual MALDI measurement. For details on this process we refer to Oetjen et al. [2016] and references therein.

In a common workflow, formalin-fixed paraffin-embedded (FFPE) tissue is subjected to a tryptic digestion step, resulting in the decomposition of proteins into smaller fragments (peptides). As a consequence, the spectral fingerprint of a protein biomarker typically consists of a characteristic pattern of spectral peaks corresponding to different peptides, as well as molecular modifications and isotopes. This observation, together with the fact that spectral intensities are strictly non-negative, motivates the use of non-negative matrix factorization (NMF) algorithms for analyzing MALDI MSI datasets.

In the following we evaluate the proposed methods on a collection of FFPE lung tumor tissue samples, including biopsies of adenocarcinoma (ADC) and squamous cell carcinoma (SqCC) (Kriegsmann et al. [2016]). Differentiation between these two frequent subtypes of non-small cell lung cancer is of high relevance for therapy selection and patient management. Diagnostic accuracy of current procedures based on immunohistochemical (IHC) stainings, however, is limited, resulting in a demand for novel analytical tumor typing methods. The applicability of MALDI MSI to this task has been demonstrated in Kriegsmann et al. [2016], where the presence of five different marker peptides belonging to four proteins has been validated.

Tissue samples for our study were provided by the tissue bank of the National Center for Tumor Diseases (NCT, Heidelberg, Germany) in accordance with the regulations of the local ethics committee. Cylindrical tissue cores of 168 ADC and 136 SqCC were assembled to 8 tissue microarray (TMA) blocks. Tissue sections were cut from all TMA blocks and processed according to a typical tissue preparation protocol established for tryptic peptide imaging (Casadonte and Caprioli [2011]).
MALDI imaging data was acquired on an Autoflex speed MALDI-TOF/TOF mass spectrometer (Bruker Daltonik) in positive ion reflector mode. Spectra were measured in the mass range 500–5000 m/z at 150 µm spatial resolution (Figure C2). After MALDI MSI data acquisition, the sections were washed and hematoxilyn-eosin (HE) stained. Tumor status and typing for all cores were confirmed by standard histopathological examination and detailed annotations were created indicating subregions containing tumor cells.

Spectral data of all TMAs was loaded into SCiLS Lab (version 2016b, Bruker Daltonik) and the software’s default baseline correction as well as total ion count (TIC) normalization was applied. For subsequent processing, the data was exported to MATLAB (version 2016b, Mathworks) and spectra outside of the annotated tumor subregions were discarded. The remaining 4667 spectra were cropped to the mass range 800–2500 m/z and resampled to intervals of 0.36 Da width centered around expected peptide masses according to the averagine model (Senko et al. [1995]). Thus, the dimensionality of the data was reduced to 1699 intensity values per spectrum. In the following, the complete spectral dataset is represented by a matrix $Y$, where spectra are stored as rows and columns correspond to m/z images. Such MSI data sets are a special case of hyperspectral imaging data, where the channels of the hyperspectral data are the m/z images of the MSI data.

### Appendix D  Detail results and correlation analysis

Detailed performance results for all investigated classification schemes are shown in Figure D3. The regularization parameters used for the different NMF models are listed in Table D1.

Classification schemes based on the supervised NMF methods (in particular with the $Flog$ model) require a surprisingly low number of features for reaching convergence in terms of classification accuracy. Indeed, a closer investigation of the pseudo spectra $X$ generated by the $Flog$ model reveals a high correlation among the basis vectors, as demonstrated in Figure D4 for the case of $p = 20$ features. In fact, as much as 15 rows of $X$ essentially represent two distinct patterns, clearly associated with positive and negative signs of $\beta$. Moreover, these patterns exhibit little dependency on the number of features (Figure D5), and thus can be interpreted as being characteristic for the two tissue classes ADC ($\beta < 0$) and SqCC ($\beta > 0$).

**Table D1: Regularization parameters for the NMF models**

<table>
<thead>
<tr>
<th>Model</th>
<th>$\lambda$</th>
<th>$\mu$</th>
<th>$\nu$</th>
<th>$\sigma_{1/2}$</th>
<th>$\gamma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FRO</td>
<td>$1.0 \times 10^{-4}$</td>
<td>$1.0 \times 10^{-6}$</td>
<td>$1.0 \times 10^{-9}$</td>
<td>0.1</td>
<td>–</td>
</tr>
<tr>
<td>FR (rescale)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>FRlda</td>
<td>$1.0 \times 10^{-5}$</td>
<td>$1.0 \times 10^{-4}$</td>
<td>$1.0 \times 10^{-7}$</td>
<td>–</td>
<td>1</td>
</tr>
<tr>
<td>FROlda</td>
<td>$1.0 \times 10^{-3}$</td>
<td>$1.0 \times 10^{-4}$</td>
<td>$1.0 \times 10^{-7}$</td>
<td>0.3</td>
<td>1</td>
</tr>
<tr>
<td>Flog</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>–</td>
<td>1</td>
</tr>
</tbody>
</table>
For comparison of the observed characteristic patterns with results published previously on the discrimination of adeno- and squamous cell carcinoma tissue of the lung, we investigate prominent features in the weighted linear combination of pseudo spectra, given by $x_\beta = X^T \hat{\beta}$. The resulting discriminatory patterns are shown in Figures 6 (main article) and D6 for the supervised NMF models FRlog and Flda, resp. Note that for the Flda model, we can let $\check{\beta} = \beta$, whereas in the Flog model, the vector $\beta$ includes one leading component representing a constant offset (see equation Flog (5) in the main article.) Thus, for Flog, $\check{\beta}$ equals $\beta$ with the first component removed.

The model FRlda includes an $\ell_1$-penalty term on $X$ that induces sparsity in the pseudo spectra. As a result, the discriminatory patterns for this model feature only very few peaks, and only the peaks at $m/z = 1411$ and 1412 occur in both patterns learned on data A and B, resp. By comparison with results published in Kriegsmann et al. [2016], we attribute these to the monoisotopic peak of a peptide of the CK5 protein ($m/z = 1410.7$) and its second isotopic peak. Moreover, the peaks at $m/z = 1879$ and 1907 that occur in only one of the two patterns can most likely be attributed to peptides of the proteins CK15 (monoisotopic $m/z = 1877.9$) and HSP27 (monoisotopic $m/z = 1905.9$). In fact, all three proteins have been shown to be indicative for squamous cell carcinoma as opposed to adenocarcinoma in the lung, matching well the strong expression in the above discriminatory patterns.

The discriminatory patterns computed based on the Flog model, on the other hand, show a different
Figure D5: Comparison of the two opposite pseudo spectra $X$ associated with the largest (positive, indicative for SqCC) and smallest (negative, ADC) weights $\beta$ for different numbers of features, generated by the *Flog* NMF decomposition of training data $A$ and $B$.

Figure D6: Discriminatory patterns learned by the method *FRlda* on data $A$ and $B$.

characteristic and contain a significant background noise, which is most likely due to the missing sparsity regularization term in this method. Moreover, since in the *Flog* model $\beta$ is allowed to be negative, both positive (indicative for SqCC) and negative (indicative for ADC) peaks are seen. Only little difference can be observed between the discriminatory patterns learned on data $A$ and $B$. In addition to peaks at $m/z = 1411/1412$ as well as $m/z = 810.4$ and 1879 also occurring in the *FRlda* patterns, peaks at $m/z = 1407$ and 1822 can be observed. The former, having a negative direction, is most likely related to a peptide of the CK7 protein (monoisotopic $m/z = 1406.7$), which was demonstrated to be indicative for adenocarcinoma in the lung (Kriegsmann et al. [2016]). The other peak can be attributed to a different...
peptide of the above mentioned CK15 protein (monoisotopic m/z = 1821.9) indicative for squamous cell carcinoma.

References


