Non-negative matrix factorization using constrained optimization with applications

Mir Mohammad Nazmul Arefin

Doctor of Philosophy
2015

Swinburne University of Technology
Melbourne, Australia
Abstract

This research work provides a new method for solving non-negative matrix factorization (NMF) problems, which has applications to image processing, text clustering and data mining. Matrix factorization has been a popular topic of research for some decades and several methods and algorithms have been developed for matrix factorization. NMF has received much attention recently, as many of the practical problems inhere the property of being non-negative naturally. In NMF, a high dimensional non-negative matrix is factorized into two lower dimensional non-negative matrices, sometimes referred to as the basis matrix and the co-efficient matrix respectively. NMF has applications in the field of parts-based learning of images, the analysis of hyper-spectral images, text and data mining, artificial neural networks, dimension reduction and many others, where a factorization of a data matrix is non-negative by nature. Factorizing these data matrices provides important information and reduced dimensions of the data, which helps further analysis.

Due to the lack of information about the factors with the exception of the property of non-negativity, it is hard to find a global solution for an NMF problem. However, local solutions are useful for numerous applications. Researchers have proposed different methods to factorize the original data matrix. The problem itself has non-convexity, and finding the solution alternately may introduce convexity in the sub-problems. To find a feasible solution for a particular problem, some specific constraints have been introduced in the previously-developed methods. Approaches include multiplicative updates of the factor matrices, non-negative alternative least squares, gradient-based solving methods and geometric approaches to the problem.

In this thesis, a new alternating optimization method is proposed. Following from the revisited and new interpretations of the geometric properties of NMF, we
propose alternately minimizing the Frobenius norm of the basis matrix while max-
imizing the Frobenius norm of the co-efficient matrix in a unitized $\ell_1$-space. Mini-
mizing the Frobenius norm of the basis matrix provides a smaller convex conic hull
and maximizing the Frobenius norm of the co-efficient matrix provides a sparser
solution. The proposed alternating solving converges faster than the methods cur-
rently available and provides better performance at the same time.

The proposed method is applied to the field of image processing to separate sig-
ificant parts of the images, and to the field of text clustering to separate similar
words and documents, based on their appearance parameter. It is also be applied to
the field of dimension reduction to reduce the dimensions of a huge document data
set as part of the preprocessing of artificial neural networks. In these applications,
the proposed new method has shown better performance with faster convergence
to feasible solutions.

In summary, this thesis studies the optimal solutions and proposes a new method
for solving NMF problems. The geometric properties and convex optimization of
NMF have been studied and applied to provide a better optimum solution than
other popular methods. The claims of better performance and faster convergence
have been verified by a number of computational results.
This thesis is dedicated to my parents,
Mir Akhterul Alam and Begum Shahanara Akter.
Acknowledgements

PhD is a long journey and it is not be possible to reach the final stage without the help of many. I would like to take the opportunity to thank everyone who have supported and encouraged me throughout this period. First of all, I would like to show my gratitude to my creator, the almighty Allah for his blessings. I am very much grateful to my principal supervisor Professor Cishen Zhang. He has been resourceful, patient, and supportive. He was very motivating during the hard times of my PhD. I would like to thank Professor Zhihong Man for his contribution as co-supervisor. I am thankful to Swinburne University of Technology for their resource and financial support as scholarship and Stamford University Bangladesh for granting me leave of absence for study. I am grateful to the helpful community in the internet, who shared many knowledgeable resources, open-source software which helped to expedite my work. I am also grateful to my previous educational institutions and their faculties for sharing their knowledge. I would like to thank my colleagues, especially Wen Hou and Ifat Al Baqee for their continuous support and help. I gratefully acknowledge the work of Dr. Alex McKnight in proofreading the thesis with respect to language and grammar.

My parents have always encouraged me for higher education, I am grateful to them for giving me the freedom to do things that I like to do. My elder brother Samsul Arefin has always guided and motivated me. I am lucky to have my uncle Md. Fozle Elahi, who is like a mentor to me. He has guided me from my childhood and I am very much grateful to him. I would like to thank Dr. ASM Shihavuddin for inspiring me to join the research arena especially in digital signal processing. I
am thankful to my friend Maruf Islam for inspiring me to pursue PhD in Australia, Nazmul Mahmud and Mir Nahidul Ambia for his continuous motivations.

Living in abroad, away from your family, is always a difficult task. I am grateful to ASM Mohsin, Dr. Ayaz Chowdhury, Dr. Khalid Imran, Nasrin Chowdhury, Maruf Billah, Shirin Sultana, AHM Sayem, Seemin Ahmed, Dr. Tanveer Ahmed Chowdhury, Saima Sharmin, Tashfiq Rahman, Kazi Shariful Islam and Dr. Ziaul Haque. They have made life enjoyable. They have always supported me with their knowledge and experience when I faced any problem, whether it was personal or study related. I am also thankful to Dr. Morshed Alam, Dr. Moniruddin Ahmed, Dr. Saad bin Kashem and Dr. Imrul Hassan.

Finally, I would like to thank all the Bangladeshis in Swinburne University, my family, friends and well-wishers for their support, knowledge and prayers.

Mir Mohammad Nazmul Arefin
Melbourne, 2015
Declaration

I declare that this thesis contains no material that has been accepted for the award of any other degree or diploma and to the best of my knowledge contains no material previously published or written by another person except where due reference is made in the text of this thesis.

Mir Mohammad Nazmul Arefin
Melbourne, 2015
Contents

1 Introduction .............................................. 1
1.1 Motivation ........................................... 1
1.2 Main contributions .................................... 4
1.3 Thesis outline ........................................ 5

2 Linear algebra and matrix factorization .................. 8
2.1 Introduction .......................................... 8
2.1.1 Matrix operations and definitions ................. 9
2.1.2 Vector subspaces .................................. 10
2.1.3 Eigenvalues and eigenvectors ....................... 11
2.1.4 Vector and matrix norm ............................ 11
2.2 Optimization .......................................... 13
2.2.1 Classes of optimization problem .................... 13
2.2.2 Solving optimization problems ...................... 14
2.2.3 Karush-Kuhn-Tucker conditions for optimization .... 16
2.2.4 Convex optimization for non-convex problems ....... 17
2.3 Matrix Factorization .................................. 18
2.3.1 Low rank matrix approximation ..................... 19
2.4 Conclusion ................................................. 20

3 Non-negative matrix factorization ........................................... 22
  3.1 Introduction ............................................. 22
    3.1.1 Non-negative matrix factorization .................. 23
  3.2 Principal component analysis and NMF ..................... 24
  3.3 Fundamental Algorithms of NMF ....................... 26
    3.3.1 Multiplication updates ............................ 27
    3.3.2 Alternating Non-negative Least Square .......... 29
    3.3.3 Projected Gradient .............................. 31
    3.3.4 Other popular methods ........................... 32
  3.4 Initialization of decomposed matrices ................. 32
  3.5 Variants and extensions of NMF ..................... 33
  3.6 Applications of NMF .................................. 34
    3.6.1 Image processing ................................ 34
    3.6.2 Text and document clustering ................. 34
    3.6.3 Computational Biology ......................... 35
    3.6.4 Music analysis ................................ 35
    3.6.5 Other applications .............................. 36
  3.7 Conclusion ............................................. 36

4 Optimizing the NMF problem subject to the $\ell_1$-constraint ........................................... 37
  4.1 Introduction ............................................. 37
  4.2 Important definitions and concepts ..................... 38
  4.3 Geometric interpretation of NMF ..................... 41
  4.4 Non-uniqueness of solution ........................... 44
4.5 Optimization of Frobenius norm with constrained $\ell_1$-norm to find the feasible solution. ........................................ 46
4.6 Algorithm ................................................................. 52
4.7 Stopping Criteria ....................................................... 54
  4.7.1 Minimum error condition ........................................ 54
  4.7.2 Similar solution condition ....................................... 55
4.8 Experimental results .................................................. 55
4.9 Conclusion ............................................................... 60

5 Application to parts-based learning of images 62
  5.1 Introduction ............................................................. 62
  5.2 Previous methods for image processing .......................... 63
  5.3 NMF for parts-based learning ....................................... 64
  5.4 MinMaxNMF for parts-based imaging ............................ 65
    5.4.1 Facial Recognition .............................................. 66
    5.4.2 Medical Imaging .................................................. 74
  5.5 Conclusion ............................................................ 75

6 Text and document clustering using $\ell_1$-constrained NMF 80
  6.1 Introduction ............................................................. 80
  6.2 Popular methods for clustering .................................... 81
  6.3 Clustering using NMF ................................................ 81
  6.4 $\ell_1$-constrained NMF applied to text classification .............. 82
    6.4.1 Document clustering .......................................... 85
  6.5 Conclusion ............................................................ 87
List of Figures

3.1 Visualization of NMF ................................................. 23

4.1 At the left hexagon-shaped set is a convex set which includes its boundary shown in darker line. At the right the kidney-shaped set is a non-convex set as the line segment between two points is not contained by the set [1].................................................. 39

4.2 At the left the shaded pentagon is the convex hull of fifteen points shown as dots. The shaded illustration at the right is the convex hull of the kidney shaped set shown in Figure 4.1 [1]. ..................... 40

4.3 Conic hull for the convex set shown in Figure(4.2) [1]................. 41

4.4 Simplicial cone containing data points ................................. 42

4.5 Example of multiple convex cone available containing data points . 43

4.6 Visualization of simplicial cone on a unitized \( \ell_1 \)-space. We can see three different orientations of simplicial cones with the same shape and size for the same data matrix \( X \) (Many other orientations are also possible with the same shape and size) ......................... 45

4.7 Representation of locking and unlocking situation. .................... 46

4.8 Unitized data matrix with comparison with the original. ............. 47
List of Figures

4.9  Unitized vectors in $\ell_1$-space. .............................................. 49
4.10 Representation of column vectors of $X$, $A$ and $\tilde{A}$ in $\mathbb{R}_+^2$ .... 49
4.11 Effect of basis matrix over coefficient matrix. *Left* illustration shows a larger simplicial cone formed by basis matrix. *Right* illustration shows comparatively smaller simplicial cone with sparser (higher Frobenius norm) coefficient matrix. ......................................................... 51
4.12 Convex conic hulls created by the basis matrices .............................. 57
4.13 Decrease of error in every iteration for MU, HALS, PG and Min-MaxNMF algorithms. ................................................................. 58
4.14 Decrease of error in every iteration for variable $r$ for MinMaxNMF algorithm ................................................................. 60
5.1  Visualization of parts-based imaging using NMF ............................... 65
5.2  Probabilistic hidden variables model underlying non-negative matrix factorization [2]. ................................................................. 66
5.3  Sample images from CBCL face database ...................................... 67
5.4  Basis extract from CBCL face database by (a) MU, (b) HALS, (c) PG and (d) MinMaxNMF algorithm ......................................................... 68
5.5  Decrease of error in every iteration for MU, HALS, PG and Min-MaxNMF algorithms while extracting basis of CBCL face database. ................................. 69
5.6  Samples from the reconstructed images of CBCL face database for (a) MU, (b) PG, (c) HALS and (d) MinMaxNMF algorithms. ............................. 70
5.7  Sample images from AT&T face database ...................................... 71
5.8  Basis extracted from AT&T face database by (a) MU, (b) HALS, (c) PG and (d) MinMaxNMF algorithm ......................................................... 72
List of Figures

5.9 Decrease of error in every iteration for MU, HALS, PG and Min-
MaxNMF algorithms while extracting basis of AT&T face database. . 73
5.10 TAC for four sources, extracted from raw PET image using MinMaxNMF
algorithm. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 75
5.11 Spatial images of four different sources extracted as basis from raw
PET image using MinMaxNMF algorithm (source 1 and 2) . . . . . . . 78
5.12 Spatial images of four different sources extracted as basis from raw
PET image using MinMaxNMF algorithm (source 3 and 4) . . . . . . . 79
6.1 Visualization of document clustering using NMF . . . . . . . . . . . . . 82
6.2 Decrease of error in every iteration for NMF algorithms while cluster-
ing MEDLARS data. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 85
6.3 variation of error with respect to number of categories for reuters10
database in MinMaxNMF algorithm. . . . . . . . . . . . . . . . . . . . . 87
List of Tables

3.1 MU Algorithm ......................................................... 28
3.2 ALS Algorithm ......................................................... 29
3.3 PG Algorithm ......................................................... 32

4.1 Summary of our Algorithm ............................................ 53
4.2 Error comparison between NMF algorithms ...................... 56
4.3 Error comparison with variable $r$ .................................. 59

5.1 Final error of NMF algorithms for CBCL face database ........ 67
5.2 Final error of NMF algorithms for AT&T face database .......... 71
5.3 Face detection accuracy using basis extracted by NMF algorithms for
AT&T face database ..................................................... 73

6.1 Final error of NMF algorithms for MEDLARS text database .... 83
6.2 Top ten words in each category in MEDLARS dataset, clustered by
MinMaxNMF when $r = 10$. Here $a_i$ ($i = 1, \ldots, 10$) represents each
column of $A$. ........................................................... 84
List of Tables

6.3 Top ten words in each category in email database clustered by Min-MaxNMF when \( r = 3 \). Here \( a_i \ (i = 1, \ldots, 3) \) represents each column of \( A \). ......................................................... 86

6.4 Final error of NMF algorithms for email text database . . . . . . . . . . . . . . . . . . . . . . . . . 86

6.5 Final error of NMF algorithms for reuters10 database . . . . . . . . . . . . . . . . . . . . . . . . . 87
Symbols

\[\mathbb{R}\] \hspace{0.5cm} \text{Set of real numbers}

\[\mathbb{R}_+\] \hspace{0.5cm} \text{Set of non-negative real numbers}

\[\mathbb{C}\] \hspace{0.5cm} \text{Set of complex numbers}

\[\mathbf{X}\] \hspace{0.5cm} \text{Data matrix}

\[\mathbf{A}\] \hspace{0.5cm} \text{Basis matrix}

\[\mathbf{S}\] \hspace{0.5cm} \text{Co-efficient matrix}

\[\mathbf{I}\] \hspace{0.5cm} \text{Identity matrix}

\[m\] \hspace{0.5cm} \text{Number of data vectors}

\[n\] \hspace{0.5cm} \text{Samples in each data vectors}

\[r\] \hspace{0.5cm} \text{Reduced rank, number of categories, number of basis}

\[\mathbf{x}_i\] \hspace{0.5cm} \text{The } i^{th} \text{ data vector}

\[x_{ij}\] \hspace{0.5cm} \text{The } i^{th} \text{ row and the } j^{th} \text{ element of matrix } \mathbf{X}

\[\mathbf{A}^0, \mathbf{S}^0\] \hspace{0.5cm} \text{Initial value of } \mathbf{A} \text{ and } \mathbf{S}

\[\mathbf{A}^k, \mathbf{S}^k\] \hspace{0.5cm} \text{Value of } \mathbf{A} \text{ and } \mathbf{S} \text{ in the } k^{th} \text{ iteration}

\[\Phi\] \hspace{0.5cm} \text{Subspace of a set of vectors}

\[\mathbf{\lambda}\] \hspace{0.5cm} \text{Eigenvector}

\[\lambda\] \hspace{0.5cm} \text{Eigenvalue}

xvi
Symbols

\[ \| \cdot \| \quad \text{Norm of a vector or matrix} \]
\[ \| \cdot \|_2 \quad \ell_2\text{-norm of a vector or matrix} \]
\[ \| \cdot \|_F \quad \text{Frobenius norm of a vector or matrix} \]
\[ \| \cdot \|_1 \quad \ell_1\text{-norm of a vector or matrix} \]
\[ f(x), g(x), h(x) \quad \text{Functions of a variable } x \]
\[ x^* \quad \text{Optimal or solution of an optimization problem} \]
\[ \alpha, \beta, \gamma, \theta \quad \text{Constants} \]
\[ \sigma \quad \text{Singular value} \]
\[ \mathcal{S} \quad \text{Simplicial cone} \]
Acronyms

NMF  Non-negative matrix factorization
SVD  Singular value decomposition
PCA  Principal component analysis
ICA  Independent component analysis
SCA  Sparse component analysis
MU   Multiplicative update
KKT  Karush-Kuhn-Tucker
ALS  Alternating least squares
PG   Projected gradient
HALS Hierarchical alternating least squares
EEG  Electroencephalogram
MinMaxNMF Minimization and Maximization of NMF
FLDA Fisher linear discriminant analysis
PET  Positron emission tomography
CBCL Center for biological and computational learning
LSI  Latent semantic indexing
PLSA Probabilistic latent semantic analysis
MEDLARS Medical literature analysis and retrieval system
<table>
<thead>
<tr>
<th>Acronyms</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLM</td>
<td>National library of medicine</td>
</tr>
<tr>
<td>NBSS</td>
<td>Non-negative blind source separation</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial neural network</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Motivation

In recent times, data acquisition and processing techniques have been dealing with huge amounts of data, and it sometimes becomes overwhelming to process such large amounts of data. Moreover, in many cases data are scrambled over the acquisition and transmission processes, which make it difficult to have an idea of the original source. To deal with these problems, matrix factorization is applied to decompose large data sets or separate mixed or scrambled data from unknown sources. Using matrix factorization techniques, large data matrices can be decomposed into low-rank matrices to enable faster processing. Matrix factorization also opens up the opportunity to analyze different aspects of unknown sources of data. Several popular methods including LU decomposition [3], QR-decomposition, Cholesky decomposition, and Singular Value Decomposition (SVD) [4], are available for matrix factorization. Many other algorithms, like Principal Component Analysis (PCA) [5], Independent Component Analysis (ICA) [6], Sparse Component Analysis (SCA) [7] etc., have also been developed over time. These algorithms are sometimes constrained to specific applications and are sometimes too general to be implemented.
Chapter 1: Introduction

In many applications it is observed that the data and source matrices are naturally non-negative. Using this property of data, a new type of matrix factorization technique named, non-negative matrix factorization (NMF) has attracted the attention of researchers. It is a very recent mathematical idea, originally introduced by Paatero and Tapper in 1994 [8]. Lee and Seung [2] made it popular to solve and decompose a matrix $X$ into two non-negative matrices, $A$ and $S$, both of which are unknown initially. In some cases $A$ is referred to as the basis matrix and $S$ is referred to as the coefficient matrix. In other applications, like blind source separation, they are termed the mixing matrix and source matrix respectively. Although the resultant matrices are unknown, their characteristic of being non-negative is the first criterion for NMF:

$$X \approx AS$$

A fundamental characteristic of NMF is the ability to extract inherent features as basis vectors in $A$, which features are used for recognition and categorization in many applications including image and data analysis. NMF enables only additive combination of parts to form a whole [2] by allowing non-negative entries only, in $A$ and $S$. Parts of faces in image data, topics or categories of text and document, specific absorption characteristics in hyper-spectral data are examples of inherent features. NMF has applications in a range of fields, including parts based learning [2], text mining [9, 10], dimension reduction [11, 12, 13, 10], blind source separation (BSS) [11, 14, 15], multimedia data analysis [16], pattern recognition [17] and chemometrics [18] to name a few.

Main obstacles affecting the optimization of NMF problems include the existence of local solutions, and possibly, more important is the lack of a unique solution. Considering the only constraint, the non-negativity, unique solutions are NP hard
Chapter 1: Introduction

to find. However, local solutions are also useful in many practical cases including feature extraction, data compression and data mining applications. Therefore, NMF is quite attractive to the researchers in those fields. Several research studies are underway to find the exact solution to NMF problems and researchers have introduced many other constraints to find an optimum solution [19].

Since Lee and Seung [2] have popularized NMF problem and its optimization, a great number of research works has been dedicated to the improvement, analysis, extension and application of algorithms to solve NMF problems in various fields of science and engineering. Many authors introduced alternating solving methods for the NMF problem. Lee and Seung [20] proposed multiplicative updating (MU) rule based on the Kullback Leibler divergence to update $A$ and $S$ alternately. Following their method, many other extensions of MU rules are introduced. For example, based on Csiszar's cp-divergence, Cichocki et al. [14] proposed new cost functions. Wang et al. [21] enforced constraints based on Fisher linear discriminant analysis in his formulation for improved extraction of spatially localized features. Using the diagonal weight matrix $Q$ in a new factorization model, $XQ \approx ASQ$, was suggested by Guillamet et al. [22] to compensate for feature redundancy in the column vectors of the basis matrix. By introducing column-stochastic constraints on co-efficient matrix, the problem can be improved further [23]. Lin [24] has recently proposed the use of a projected gradient-bound constrained optimization method that is computationally competitive and appears to have better convergence properties than the standard (multiplicative update rule) approach. Use of certain auxiliary constraints may, however, break down the bound-constrained optimization assumption, limiting the applicability of projected gradient methods. Gonzalez and Zhang [25] proposed accelerating the standard approach based on an interior-point gradient method. Zdunek and Cichocki [26] proposed a quasi-Newton optimization
Chapter 1: Introduction

approach for updating \( A \) and \( S \) where negative values are replaced with small \( e > 0 \) to enforce non-negativity, at the expense of a significant increase in computation time per iteration. Several other approaches propose alternative cost function formulations can be found in \([27, 28]\). Further studies related to convergence of the standard NMF algorithm can be found in the publications by Chu et al., Lin and Salakhutdinov et al. among others \([23]\). Catral et al. \([29]\) provides information about NMF of symmetric matrices including the theoretical analysis.

With all these recent studies and opportunities for further research, NMF has become the centre of interest of our research. Given the scope for applying mathematical optimization knowledge and geometric concepts, the motivation has been developed to approach NMF from a different perspective and find a solution for the problem. Numerous practical applications of NMF have also inspired this research work.

1.2 Main contributions

This thesis work has summarized the development of NMF to date and proposes a novel method for solving NMF problems. Contributions of this research study can be summarized as follows:

- In order to provide a novel approach to the solution of NMF problems, all the previous ideas and concepts are summarized. The algorithms are analyzed and their performances are compared to provide the information needed for a novel algorithm.

- A novel method of solution is proposed, where the mathematical optimization and geometric concepts of NMF are revisited. A new objective function
is proposed along with an algorithm that solves the problem alternately. This algorithm differs from other currently available methods in terms of optimization function, and performs better.

- The proposed algorithm is applied to parts-based learning of images. NMF has the ability of decomposing important features from any set of images without any prior information. Our algorithm provides better performance and feature extraction for parts-based learning of facial and medical images.

- The newly developed algorithm has been used for text and data clustering and compared with other traditional and NMF-based algorithms to prove the versatility of the algorithm. The proposed method outperforms its competitors and provides better clustering.

- Further guidelines for research on NMF are provided at the end of the thesis. This study provides a new outlook on the problem and develops a new optimization function, which in future may have an important role in the field of NMF.

1.3 Thesis outline

Chapter One: Introduction

The first chapter of the thesis explores the history of matrix factorization and non-negative matrix factorization. This chapter provides information about previous research over the time. The motivation for the thesis is also explained here. The contributions of this thesis are provided in brief. The structure of the thesis is provided at the end of the chapter.
Chapter One: Introduction

Chapter Two: Linear algebra and matrix factorization

Some important fundamentals of linear algebra and matrix factorization are presented in this chapter. The mathematics behind matrix manipulation and factorization is discussed.

Chapter Three: Non-negative matrix factorization

Fundamental theories of non-negative matrix factorization are explained in this chapter. Details about the previously-developed algorithms are provided, along with their properties and performance. The drawbacks of the methods are also highlighted. Brief information regarding the applications of NMF is provided in this chapter.

Chapter Four: Optimizing the NMF problem subject to the $\ell_1$-constraint

This chapter provides more descriptions including basic definitions and motivations based on geometric analysis, applications and concepts of convex optimization and NMF problems. The mathematical background to the need for a new optimization is explained. The formulation of a new cost function subject to constrained $\ell_1$-norm of matrices is discussed to introduce the new method that will minimize the Frobenius norm of the basis matrix and maximize the Frobenius norm of the co-efficient matrix in every iteration. An analysis of the advantages of the proposed algorithm over existing algorithms is presented; results and simulations of the performance are also discussed.
Chapter 1: Introduction

Chapter Five: Application to parts-based learning of images

Parts-based learning of images is explained with its usefulness for practical applications. Brief information is provided regarding the popular methods for parts-based learning and facial image processing. The newly-developed method is then applied to face recognition, facial expression recognition and medical image processing. The results are compared with those for popular methods and it is shown that our algorithm performs better.

Chapter Six: Text and document clustering using $\ell_1$-constrained NMF

The field of text clustering and data mining is studied in this chapter. Previously-developed and popular algorithms in this field are explained briefly, with their advantages and disadvantages. The application of NMF in the field of text and data processing is discussed to provide the basis for application of the proposed novel optimization method. The proposed method is then applied to a medical abstracts data set, an email data set and a document dataset. The results are explained and compared with other NMF-based algorithms. It is observed that our algorithm performs better than other algorithms in several aspects.

Chapter Seven: Conclusions and Recommendations

In this chapter the study is summarized and outcomes are outlined in brief for an overall understanding of the achievements. A guideline for future work is provided with explanations which will help to expand this work in future.
Chapter 2

Linear algebra and matrix
factorization

2.1 Introduction

A matrix is a rectangular array of numbers. If the array has \( n \) rows and \( m \) columns, then it is recognized as \( n \times m \) matrix, and \( n, m \) are called the dimensions of the matrix. If a matrix has only one row or column, then it is called a vector. In this thesis, matrices are denoted by capital boldface letters and vectors are denoted by lower case boldface letters. For example \( X \) is a matrix and \( x_j \) is the \( j^{th} \) column of \( X \), where scalar \( x_{ij} \) is the value of the \( i^{th} \) row and the \( j^{th} \) column. When all the elements of a matrix are real scalar, the matrix is called a real matrix. A set of \( n \times m \) real valued matrices are denoted by \( \mathbb{R}^{n \times m} \). In this thesis, all the matrices are real. Simple algebraic operations such as addition, subtraction, multiplication, division etc. are also applicable over matrices and vectors. However, they have to follow certain rules, depending on the types of matrices. Matrices are often categorized depending on their individual elements, the orientation of rows and columns, dimensions and many other properties. Some other operations like transformation,
rotation, inversion etc. are also available for matrix algebra. Details about elementary matrix algebra can be found in [4]. The following sections of this chapter will provide some of the important operations, properties and theories about matrices; these will be used frequently throughout this thesis.

2.1.1 Matrix operations and definitions

Considering the matrix $X \in \mathbb{R}^{n \times m}$, some of the basic but useful matrix operations, properties and definitions are presented here.

- **Matrix transpose**: Transpose of a matrix $X$ is denoted by $X^T$ where $(X^T)_{ij} = X_{ji}$. If $X$ is a symmetric matrix, then $X^T = X$.

- **Matrix vectorization**: The matrix $X \in \mathbb{R}^{n \times m}$ can be transformed into a column vector by stacking its columns into one.

$$\text{vec}(X) = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix}$$

- **Rank of matrix**: The number of linearly independent rows in a matrix is called the row rank of a matrix, and the number of linearly independent columns in a matrix is called the column rank of a matrix. Following the fundamental results, in linear algebra the row rank and the column rank are equal and simply termed as rank of a matrix.

Therefore, the rank of matrix $X$ will be

$$\text{rank}(X) \leq \min(n, m)$$
• Trace of a matrix: The trace of a square matrix $X \in \mathbb{R}^{n \times n}$ is the sum of its main diagonal elements. Trace can be defined as

$$trace(X) = \sum_{i=1}^{n} x_{ii}$$

Trace can also be interpreted as the sum of eigenvalues of a square matrix. If $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of $X$ then

$$trace(X) = \sum_{i=1}^{n} \lambda_i$$

### 2.1.2 Vector subspaces

The set of all linear combinations of a set of vectors $X = \{x_1, x_2, \ldots, x_m\}$ of $\mathbb{R}^n$ is called the linear subspace of $\mathbb{R}^n$. If we denote the subspace as $\Phi$ then,

$$\Phi = \left\{ \sum_{i=1}^{m} \alpha_i x_i : \alpha_i \in \mathbb{R} \right\}$$

$\Phi$ is also called the span of $X$ and $X$ is called the spanning set of $\Phi$. For a subspace $\Phi$ there can be many spanning sets. If no vector can be removed from a spanning set without changing the span, that spanning set is called linearly independent and a basis of $\Phi$. The cardinality of a basis of a span is fixed and is called the dimension of that span.

The rank of a matrix $A \in \mathbb{R}^{n \times m}$ can be expressed by the dimension of the subspace spanned by the column vectors of $A = \{a_1, a_2, \ldots, a_m\}$. If this subspace is denoted as $\Phi_A$ then:

$$rank(A) = dim(\Phi_A) \leq min(n, m)$$

A span is closed under the addition and scalar multiplication, meaning:
Chapter 2: Linear algebra and matrix factorization

2.1.3 Eigenvalues and eigenvectors

Eigenvalues and eigenvectors of a square matrix are very important concepts in matrix analysis as they carry essential information about the matrix. Similar concepts for rectangular matrices are known as singular values and vectors. They have dominating characteristics of the original matrix and are very important for matrix factorization.

**Definition:** A scalar \( \lambda \in \mathbb{C} \) is an eigenvalue of the matrix \( X \in \mathbb{C}^{n \times n} \) if there exist a non-zero vector \( v \in \mathbb{C}^n \) such that \( Xv = \lambda v \). The vector \( v \) is called the associated eigenvector of the eigenvalue \( \lambda \).

For a real symmetric matrix \( A \in \mathbb{R}^{n \times n} \), the eigenvalues and the eigenvectors are also real. Moreover, for this real matrix if all the eigenvalues are non-negative, \( A \) is said to be **positive semidefinite**. If the eigenvalues are strictly positive then it is called **positive definite**.

2.1.4 Vector and matrix norm

The measurement of the length or magnitude of a vector or a matrix is termed as *norm*. A norm of \( x \in \mathbb{R}^n \) or \( X \in \mathbb{R}^{n \times m} \) is a real function \( || \cdot || \) that satisfies the following conditions:

- \( ||x|| \geq 0, \quad \forall x \in \mathbb{R}^n \) or \( ||X|| \geq 0, \quad \forall X \in \mathbb{R}^{n \times m} \)
- \( ||x|| = 0 \iff x = 0 \)
- \( ||\alpha x|| = |\alpha||x| \quad \forall x \in \mathbb{R}^n \) and \( \forall \alpha \in \mathbb{R} \)
Chapter 2: Linear algebra and matrix factorization

- \|x + y\| \leq \|x\| + \|y\| \quad \forall x, y \in \mathbb{R}^n

The Euclidean norm or the Frobenius norm denoted by \| \cdot \|_F is the most common norm in practice. This can be derived from the inner product:

\[ \|x\|_F = \sqrt{\langle x, x \rangle} \]

Above formulation is applicable to both vectors and matrices. In least squares problems, one tries to minimize an error which is usually measured by Frobenius norm. Other popular norms are generalized by the Holder norms (\(p\)-norms) as

\[ \|x\|_p = \left( \sum_{i=1}^{n} |x_i|^p \right)^{1/p} \quad p = 1, 2, \ldots \]

where the most commonly used are \(p = 1, p = 2\) and \(p = \infty\):

- 1-norm: \(\|x\|_1 = |x_1| + |x_2| + \ldots + |x_n| \)
- 2-norm: \(\|x\|_2 = \sqrt{|x_1|^2 + |x_2|^2 + \ldots + |x_n|^2} \)
- \(\infty\)-norm: \(\|x\|_\infty = \max_i |x_i| \)

For vectors, 2-norm (\(\|\cdot\|_2\)) is equal to its Frobenius (\(\|\cdot\|_F\)) norm. However, this does not hold for matrices. For the case of matrices, the Frobenius norm can be expressed in terms of trace as follows:

\[ \|X\|_F = \sqrt{\text{trace}(X^*X)} \]

here \(X^*\) means the conjugate transpose of \(X\). The Frobenius norm of the matrix is extensively used in this thesis as the main cost function for optimization utilizes this norm. Other norms are also be used to introduce constraints over the solution.
2.2 Optimization

In this section, some basics of mathematical optimization are presented. A mathematical optimization problem has the form [1]:

\[
\begin{align*}
\text{minimize} & \quad f_0(x) \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m.
\end{align*}
\]

(2.1)

here the vector \( x = (x_1, \ldots, x_n) \) is the optimization variable of the problem, the function \( f_0 : \mathbb{R}^n \to \mathbb{R} \) is the objective function and the functions \( f_i : \mathbb{R}^n \to \mathbb{R}, i = 1, \ldots, m \) are the constraint functions and the constants \( b_1, \ldots b_m \) are the limits or bounds for the constraints. A vector \( x^* \) is called optimal or a solution of the problem (2.1) if it has the smallest objective value among all vectors that satisfies the constraints [1]. For any \( z \) with \( f_i(z) \leq b_i, f_0(z) \geq f_0(x^*) \). A function can also be maximized; a maximization problem has the following form

\[
\begin{align*}
\text{maximize} & \quad f_0(x) \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m.
\end{align*}
\]

(2.2)

where maximizing an objective function is simply minimizing the negative of that function. Therefore, the maximization of \( f_0(x) \) is the same as the following optimization problem

\[
\begin{align*}
\text{minimize} & \quad (-f_0(x)) \\
\text{subject to} & \quad f_i(x) \leq b_i, \quad i = 1, \ldots, m.
\end{align*}
\]

(2.3)

2.2.1 Classes of optimization problem

Optimization problems are classified into categories characterized by particular forms of the objective and constraint functions. For example, the optimization problem presented in (2.1) is a linear program if the objective and constraint functions
Chapter 2: Linear algebra and matrix factorization

$f_0, \ldots, f_m$ are linear. These functions are linear if they satisfy

\[
f_i(ax + \beta y) = af_i(x) + \beta f_i(y)
\]

(2.4)

for all $x, y \in \mathbb{R}^n$ and all $\alpha, \beta \in \mathbb{R}$. If the optimization problem does not satisfy the condition of (2.4) then it is called a nonlinear program. Another class of optimization problems are called convex optimization problems when the objective and constraint functions are convex, meaning that they satisfy the following condition:

\[
f_i(ax + \beta y) \leq af_i(x) + \beta f_i(y)
\]

(2.5)

for all $x, y \in \mathbb{R}^n$ and all $\alpha, \beta \in \mathbb{R}$ with $\alpha + \beta = 1, \alpha \geq 0, \beta \geq 0$. Comparing the conditions presented in (2.4) and (2.5), it can be said that the convex problems are more general than linear problems. Since any linear program is also a convex optimization problem, convex optimization can be considered as the generalization of linear programming [1]. More definitions and properties of convex optimization will be provided later in the thesis.

2.2.2 Solving optimization problems

A class of optimization problems is usually solved using a solution method or algorithm that computes the solution of the problem to some given accuracy. During the last century, many algorithms were developed to solve and analyze various classes of optimization problems. The effectiveness of these algorithms depends on factors such as the form of a particular objective function, the constraint function, number of variables and constraints and special structures. Although the general form of the optimization problem presented in (2.1) is surprisingly difficult to solve, there are some problem classes which have effective algorithms to solve even large problems with many variables. Least-squares problems and linear programs are two of the
most well-known and important examples of these types of problem classes. Details about these problems and their solutions are available in [1].

In the present thesis the concentration is on nonlinear programs and convex optimizations. There is no general or analytical formula for solving convex optimization problems. Recognizing a convex function is more difficult than the least-squares problem. The challenge in solving nonlinear convex optimization problems is to recognize and formulate the problem. There are some very effective methods to solve them such as the interior-point method.

A solution method for optimization problems may produce two types of solutions: i) local ii) global.

**Local solution**

A local solution is a point which is only locally optimal, meaning that it minimizes the objective function among the feasible points which are near or referred to as neighbouring points, but does not guarantee a lower objective value than all feasible points. Since local solutions only require information about the neighbouring points, the differentiability of the objective and constraint functions; local solution methods are faster, capable of handling large-scale problems, and widely applicable.

Local solutions are not the best solution. While solving for a local optimal value, any method requires an initial guess of the optimization variable. This initial guess is critical and greatly affects the objective value of the local solution obtained. Local optimization methods are sometimes sensitive to algorithm parameter values, which need to be adjusted for particular problem classes.
Global solution

A global solution is the point that minimizes the objective function among all the feasible points. A global solution method returns the best solution where the compromise is efficiency. The computational complexity and time consumption of a global solution method are much higher than any local solution methods. In worst cases, the complexity of a global solution method grows exponentially with the problem sizes $n, m$. However, for particular problems the method can be faster than for other problems.

Due to the difficulties of finding global solutions, local solutions are sometimes the feasible choice. The necessary conditions for local solutions can be derived easily by differential calculation. In the next chapter, more information will be provided, indicating the reasons for choosing local solutions for NMF problems.

2.2.3 Karush-Kuhn-Tucker conditions for optimization

Karush-Kuhn-Tucker (KKT) conditions are first order conditions for a nonlinear programming problem to produce optimal solutions. Consider the following constrained optimization problem:

$$
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0, \quad g_j(x) \leq 0
\end{align*}
$$

(2.6)

where $h_i(x) = 0, (i = 1, 2, \ldots, k)$ are the $k$ equality constraints and $g_j(x) \leq 0, (j = 1, 2, \ldots, m)$ are the $m$ inequality constraints. The KKT conditions for the above problem can be stated as follows:

**Proposition [30]:** Let $x^\ast$ be the local solution of the above problem; the objective
and constraint functions are continuously differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}$, $\nabla h_i(x^*)$ and $\nabla g_i(x^*)$ are linearly independent. Then there exist unique constants $\mu_i$ and $\lambda_j$ such that:

$$\nabla f(x^*) + \sum_{i=1}^{k} \mu_i \nabla h_i(x^*) + \sum_{j=1}^{m} \lambda_j \nabla g_i(x^*) = 0$$

$$\lambda_j \geq 0$$

$$\lambda_j g_j(x^*) = 0$$

where $\nabla$ represents the gradient of a function. This constrained problem is sometimes written in its associated Lagrange function:

$$L(x, \mu_i, \lambda_j) = f(x) + \sum_{i=1}^{k} \mu_i h_i(x^*) + \sum_{i=1}^{m} \nabla g_i(x^*)$$

where $\mu_i$ and $\lambda_j$ are the same as KKT conditions. However, they are called Lagrange multipliers.

### 2.2.4 Convex optimization for non-convex problems

While it is easier to solve convex problems using convex optimization, it also plays an important part in solving non-convex problems. A non-convex problem can be formulated to an approximate but convex problem. By solving these approximate problems using convex optimization, the exact solution can be found for these convex problems without any initial guess. Later, this exact solution can be used as the starting point of a local solution and then optimize the non-convex problem. Alternating optimization algorithms utilizes this approach by formulating a non-convex problem into multiple convex sub-problems and solving them alternately in every iteration.
2.3 Matrix Factorization

Matrix factorization is a fundamental idea of linear algebra, where a matrix is represented as product of two or more simpler matrices. For example, these matrices can be triangular, diagonal, orthogonal etc. These kinds of factorization are the basic tool for describing and analyzing other numerical algorithms. Several fundamental matrix factorizations are available and popular among mathematicians, some being centuries old. Singular value decomposition (SVD), QR, LU (LU stands for Lower Upper) and Cholesky factorizations are popular methods, and several other methods are also available [4]. If $X$ is a square matrix, in LU factorization $X$ will be decomposed into two matrices, a lower triangular matrix $L$ and an upper triangular matrix $U$ as

$$X = LU$$

$$
\begin{bmatrix}
x_{11} & x_{12} & x_{13} \\
x_{21} & x_{22} & x_{23} \\
x_{31} & x_{32} & x_{33}
\end{bmatrix}
= 
\begin{bmatrix}
l_{11} & 0 & 0 \\
l_{21} & l_{22} & 0 \\
l_{31} & l_{32} & l_{33}
\end{bmatrix}
\begin{bmatrix}
u_{11} & u_{12} & u_{13} \\
u_{22} & u_{22} & u_{23} \\
u_{32} & 0 & u_{33}
\end{bmatrix}
$$

In this particular example the decomposed matrices are of the same dimensions as the original matrix. Another type of matrix decomposition provides low rank factors of the original matrix. This particular class of matrix decomposition or factorization has been studied in this thesis, and this type helps to reduce the dimensions of large data.
2.3.1 Low rank matrix approximation

Low rank approximation is a special case of optimization problem, where a reduced rank constrained is introduced while approximating factors of a matrix. The correctness of approximation is measured with respect to the Frobenius norm. SVD, principal component analysis (PCA) and independent component analysis (ICA) are examples of low rank matrix factorization methods. For example, in SVD the results follow the well-known Eckart-Young Theorem [31].

Theorem 2.1 (Eckart-Young). Let \( X \in \mathbb{R}^{n \times m} (n \geq m) \) have the following singular value decomposition:

\[
X = U \Sigma V^T
\]  

(2.8)

\[
\Sigma = \begin{bmatrix}
\sigma_1 & 0 & \ldots & 0 \\
0 & \sigma_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \sigma_m \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 0
\end{bmatrix}
\]

where \( \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m \geq 0 \) are the singular values of \( X \), \( U \in \mathbb{R}^{n \times n} \) and \( V \in \mathbb{R}^{m \times m} \) are orthogonal matrices. Then for \( 1 \leq r \leq m \), the matrix

\[
X_r = U \Sigma_r V^T
\]
is a global optimizer of the problem (2.9)

\[
\min_{W \in \mathbb{R}^{n \times m}} \frac{1}{2} \|X - W\|_F^2
\]

subject to \( \text{rank}(W) \leq r \) \hspace{1cm} (2.9)

and the error is

\[
\frac{1}{2} \|X - W\|_F^2 = \frac{1}{2} \sum_{i=r+1}^{m} \sigma_i^2
\]

Proof of this theorem and other implications can be found in [32].

None of the methods named so far have the non-negativity constraint while decomposing matrices into lower rank matrices. In the next chapter the concept of non-negative constraint is discussed.

### 2.4 Conclusion

Linear and matrix algebra is the basic mathematical foundation of modern-day data analysis. Large amounts of data are usually presented as matrices. Various properties and operations of matrices make it easy to handle large data; these data can be
Chapter 2: Linear algebra and matrix factorization

a time series signal, image, list of documents etc. Optimization of matrix functions provides solutions to many practical problems which may have been difficult using other methods. Several types of optimization problems exists, not all of which are easy to solve. Some classes of optimization problems, like least squares problems are important tools for analyzing many real-world problems. Convex optimization is another class of optimization which tries to find global solutions for a problem set. When global solutions are difficult to find, an algorithm may search for local solutions, which may not be the best but practically feasible.

Matrix factorization and low-rank matrix approximation help to find multiple matrices which have lower dimensions than the original, but contain the important properties of that particular matrix. These factors are further utilized in machine learning and neural networks.
Chapter 3

Non-negative matrix factorization

3.1 Introduction

When all the elements of a matrix are non-negative, it is called a non-negative matrix. A set of \( n \)-dimensional non-negative vectors is denoted by \( \mathbb{R}^n_+ \) and set of \( n \times m \)-dimensional non-negative matrices is denoted by \( \mathbb{R}^{n\times m}_+ \). These subsets of \( \mathbb{R} \) are called the non-negative orthants.

If there is no zero row in a non-negative matrix, it is called row-allowable, and similarly when there is no zero column in a non-negative matrix, it is called column-allowable. If all the column sums of a non-negative matrix are equal to one, it is called column-stochastic and if the all row sums are equal to one, it is called row stochastic. It is called doubly stochastic when both row and column sums are equal to one.

The most important theorem for non-negative matrices is the following:

**Theorem 3.1** (Perron-Frobenius [33]): *If \( \mathbf{X} \) is a square non-negative matrix, there exist a largest modulus eigenvalue of \( \mathbf{X} \) which is non-negative and a non-negative eigenvector corresponding to it.*
This eigenvector is usually referred to as the Perron vector of the respective non-negative matrix. A similar theorem can be established for a rectangular non-negative matrix by largest singular value and corresponding singular vector.

### 3.1.1 Non-negative matrix factorization

Non-negative matrix factorization (NMF) is a class of low-rank matrix factorization which decomposes a non-negative matrix into two other low rank non-negative matrices. If \( X \in \mathbb{R}_{+}^{n \times m} \) is the matrix to be factorized and \( r < \min(n, m) \) is a reduced rank, an NMF problem consists of searching two other matrices \( A \in \mathbb{R}_{+}^{n \times r} \) and \( S \in \mathbb{R}_{+}^{r \times m} \) that approximate \( X \), i.e.

\[
X \approx AS \tag{3.1}
\]

Figure 3.1 shows a visualization of NMF. Matrix \( X \) is decomposed into a tall matrix \( A \) and a wide matrix \( S \).

![Figure 3.1: Visualization of NMF](image)

In order to measure the quality of the approximations, it is important to define
cost functions, by using proximity metrics, between the original data matrix $X$ and the resulting approximation $AS$. Two common metrics are the Euclidean distance presented in (3.2) and the generalized Kullback-Leibler divergence presented in (3.3):

\[
    J(X|A, S) = ||X - AS||_F^2 \quad (3.2)
\]

\[
    D(X|A, S) = \sum_{ij} \left( (X)_{ij} \log \left( \frac{(X)_{ij}}{(AS)_{ij}} \right) - (X)_{ij} + (AS)_{ij} \right) \quad (3.3)
\]

An optimization problem will involve minimizing any of the above cost functions to approximate $A$ and $S$, i.e.

\[
    \text{minimize}_{A, S} \quad J(X|A, S) \quad (3.4)
\]

subject to $A \succeq 0, S \succeq 0$.

where $\succeq$ denotes element-wise operation of $\geq$.

### 3.2 Principal component analysis and NMF

Principal component analysis (PCA) is the most popular subspace based matrix factorization method [34] so far. It has been used for many decades. PCA produces new variables which consist linear combination of the original data vectors. In PCA, the first component has the most variance and the last one has the least. The collection of the new variables are called the principal components (PCs). If $X = \{x_1, \ldots, x_m \in \mathbb{R}^n\}$ is a set of variables and $P$ is the set of PCs, then to find $P$ one need to solve the problem in (3.5)
where \( r \) is the number of principal components and \( I \) is the identity matrix. Important feature of \( P \) is that it has orthonormal rows. Many algorithms are developed to find PCs. The easiest and most common method is finding PCs using singular value decomposition (SVD) [35]. It decomposes the data matrix \( X \in \mathbb{R}^{n \times m} \) into three parts:

\[
X = UDV^T
\]

where \( U \in \mathbb{R}^{n \times r} \) is an orthonormal matrix and contains left singular vectors, \( D \in \mathbb{R}^{r \times r} \) is a diagonal matrix containing the singular values and \( V \in \mathbb{R}^{m \times r} \) is an orthonormal matrix which contains right singular vectors. Now PCs \( P \in \mathbb{R}^{n \times r} \) can be obtained by

\[
P = UD
\]

Therefore, (3.6) can be written as

\[
X = PV^T
\]

The matrix \( P \) is similar to the basis matrix of NMF and \( V^T \) is similar to the co-efficient matrix of the same. However, PCA has many problems while decomposing non-negative matrices into non-negative factors. It can only produce few components which are non-negative, but has many negative values in many components [36]. Therefore, PCA is not suitable for applications where both data and factor matrices are purely non-negative. The disadvantages of PCA against NMF can be summarized.
Chapter 3: Non-negative matrix factorization

as following:

- PCA involves both addition and subtraction, while NMF involves only addition. Subtraction does not make sense for many applications like in facial image classification, text or document clustering.

- PCA fails to produce purely non-negative factors for non-negative matrices.

- PCA works best for compact classes of data, while NMF works best for dispersed classes.

- Basis extraction is not physically intuitive in PCA while it is for NMF.

- NMF provides parts-based learning, while PCA cannot provide parts based information.

After the introduction of NMF, many experiments are performed to compare the performance of PCA and NMF. NMF outperforms in most of the cases specially in facial and medical image processing, text and document classifications, musical source separation etc. [25, 37, 38, 39, 40, 41]. In addition to these, other subspace based methods like PCA shows similar disadvantages compared to NMF [42, 43, 44, 45].

### 3.3 Fundamental Algorithms of NMF

Based on the optimization technique, existing NMF algorithms can be categorized into three main classes: multiplicative updates (MU), alternating least squares (ALS) and the gradient-based methods. The categorization is based on the popularity of the algorithms. The alternating least squares method is the earliest, introduced by Paatero [46] for positive matrix factorization. Lee and Seung [2] brought attention to the topic through their multiplicative update rule and introduced the term
"non-negative matrix factorization". Many researchers have been encouraged to implement NMF in numerous fields of application because of the simplicity of the MU algorithm. The projected gradient (PG) algorithm was introduced in [24] with advantages for large-scale problems. Some extensions and modified versions of these algorithms have been developed during the last few years [47].

In the next sections, the main three types of the algorithms are discussed.

### 3.3.1 Multiplication updates

The multiplicative updates (MU) rule is the most popular NMF algorithm because of its simplicity. Lee and Seung proposed MU rules for NMF problems in [2] and further extended the algorithm in [20]. To formulate the updating rules for the approximate factor matrices, they chose to fix one factor (i.e. $A$ or $S$) and minimize the other factor. The objective function presented in (3.2) can be minimized with respect to $A$ and $S$ respectively to produce the updating rules.

\[
\min_{A \geq 0} \|X - AS\|_F^2
\]  

\[
\min_{S \geq 0} \|X - AS\|_F^2
\]

The following theorem is proposed by Lee and Seung [20] for establishing multiplicative updates:

**Theorem 3.2** [20] *The Euclidean distance $\|X - AS\|$ is non-increasing under the update rules*

$$
S_{\mu} \leftarrow S_{\mu} \frac{(A^TX)_{\mu}}{(A^TAS)_{\mu}}$$

$$
A_{ia} \leftarrow A_{ia} \frac{(XS^T)_{ia}}{(ASS^T)_{ia}}
$$
Chapter 3: Non-negative matrix factorization

The Euclidean distance is invariant under these updates if and only if $A$ and $S$ are at a stationary point of the distance.

They also proposed the following theorem to minimize the objective function mentioned in (3.3):

**Theorem 3.3** [20] The divergence $D(X|A, S)$ is non-increasing under the update rules

$$S_{a\mu} \leftarrow S_{a\mu} \frac{\sum_i A_{ia} X_{i\mu}/(AS)_{i\mu}}{\sum_j A_{ja}}$$

$$A_{ia} \leftarrow A_{ia} \frac{\sum_{\mu} S_{a\mu} X_{i\mu}/(AS)_{i\mu}}{\sum_k A_{ka}}$$

The divergence is invariant under these updates if and only if $A$ and $S$ are at a stationary point of the distance.

Here $S_{a\mu}$ denotes the $a^{th}$ row and $\mu^{th}$ column element of $S$ and $A_{ia}$ denotes the $i^{th}$ row and $a^{th}$ column element of $A$. Proof of the above theorems and other related derivations are provided in [20]. The pseudo code for the MU is given in Table 3.1.

Table 3.1: MU Algorithm

<table>
<thead>
<tr>
<th>MU Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Initialize $A_{ia}^0 &gt; 0, S_{a\mu} &gt; 0, \forall i, a, \mu$</td>
</tr>
<tr>
<td>2: for $k = 0, 1, 2, \ldots$ do</td>
</tr>
<tr>
<td>3:</td>
</tr>
<tr>
<td>$A_{ia}^{k+1} \leftarrow A_{ia}^k \frac{(X(S^k)^T)<em>{ia}}{(A^k S^k(S^k)^T)</em>{ia}}$</td>
</tr>
<tr>
<td>$S_{a\mu} \leftarrow S_{a\mu} \frac{(A^T X)<em>{a\mu}}{(A^T AS)</em>{a\mu}}$</td>
</tr>
<tr>
<td>4: end for</td>
</tr>
</tbody>
</table>

Although MU method has been a popular NMF method for a long time, it lacks
optimization properties [24]. Despite Lee and Seung claim that the algorithm converges to a stationary point satisfying the Karush-Kuhn-Tucker (KKT) condition, Gonzales and Zhang [25] stated that the claim is incorrect. Again, to keep the algorithm well-defined, the denominators in the updating formulae have to be greater than zero to avoid division by zero. As a result, the solution has to be strictly positive. Some proposals have been made to add a small positive number in every iteration to avoid zero [48]; however, this creates numerical difficulties in some situations [24].

### 3.3.2 Alternating Non-negative Least Square

The alternating non-negative least square method was the first algorithm proposed to solve NMF problems in [46]. It also minimizes the Euclidean distance function of (3.2) while updating the values of $A$ and $S$ alternatively. Steps of the algorithm are:

1. Find $A^{k+1}$ such that $||X - A^{k+1}S^k||_F^2 \leq ||X - A^kS^k||_F^2$
2. Find $S^{k+1}$ such that $||X - A^{k+1}S^{k+1}||_F^2 \leq ||X - A^{k+1}S^k||_F^2$

where $A^k$ means the value of $A$ at the $k^{th}$ iteration. The pseudo code for the ALS is given in Table 3.2.

<table>
<thead>
<tr>
<th>ALS Algorithm</th>
</tr>
</thead>
</table>
| 1: Initialize $A^0 \succeq 0$, $S^0 \succeq 0$  
2: for $k = 0, 1, 2, \ldots$ do  
3: $A^{k+1} = \minimize_{A \succeq 0} ||X - A^{k+1}S^k||_F^2$  
$S^{k+1} = \minimize_{S \succeq 0} ||X - A^{k+1}S^{k+1}||_F^2$  
4: end for |

29
In the bounded constrained optimization, this method is treated as the "Block Co-ordinate Descent" method [30], where in every iteration one block of variables is minimized while others remain constant. For NMF problems the number of block variables is two (A and S). The sub-problems of the ALS algorithms need to be treated separately as individual optimization problems. Several methods can be applied for this application. Solving two sub-problems in each iteration may slow down the process compared with the MU algorithm, but ALS algorithms have better optimization properties than the MU algorithm. One may think it is quite common to achieve the convergence for ALS algorithms. Paatero [46] states that for the case of alternating non-negative least squares, the convergence is guaranteed no matter how many blocks of variables there are to optimize, although past convergence analysis for block co-ordinate descent methods requires sub-problems to have unique solutions which do not hold in this case. Fortunately for the case of two blocks, the uniqueness criterion is not mandatory [49]. Another issue is whether the optimizations have at least one stopping point. To ensure convergence, one may need to introduce some sort of upper bound. The MU algorithm may not be suitable for modification to achieve better convergence. The ALS algorithm may slow down the calculation process, but it has better optimization criteria than the multiplicative algorithm.

ALS algorithms minimize both A and S to achieve the acceptable solution for the NMF problem, in the proposed algorithm of this thesis, additional constraints and optimization functions are formulated. The explanation of these updates is provided in the next chapter.
3.3.3 Projected Gradient

Lin proposed a projected gradient (PG)-based alternating least square method in 2007 [24], which became popular later. This method solves the sub-problems of the ALS method using the projected gradient approach. To explain the method, the following standard bound constrained optimization problem is considered:

\[
\begin{align*}
\text{minimize} & \quad f(w^k) \\
\text{subject to} & \quad l_i \leq w_i \leq u_i, i = 1, 2, \ldots, n
\end{align*}
\]  

(3.11)

where \( f(w) \) is a constantly differentiable function and \( l \) and \( u \) are lower and upper bounds, and \( k \) is the index of iteration. Projected gradient methods update the current solution \( w^k \) to \( w^{k+1} \) by the following rule:

\[
w^{k+1} = P[w^k - \alpha_k \nabla f(w^k)]
\]

where \( P[w_i] = \begin{cases} 
  w_i & \text{if } l_i \leq w_i \leq u_i \\
  u_i & \text{if } w_i \geq u_i \\
  l_i & \text{if } w_i \leq l_i
\end{cases} \)

Table 3.3 outlines the PG algorithm for solving one sub-problem of the ALS method.

The author of the proposed PG-NMF algorithms claims to have faster convergence of the algorithms. Using the property of \( r << \min(n, m) \) the step sizes are reduced significantly. Despite the author’s claim, we found that the algorithm fails to separate matrices in several cases with higher dimension data sets. Moreover, this method also uses the concept of minimizing both \( A \) and \( S \) whereas our algorithm uses different cost functions to achieve better results. In the next chapter we ex-
Chapter 3: Non-negative matrix factorization

Table 3.3: PG Algorithm

<table>
<thead>
<tr>
<th>Projected Gradient Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Given $0 &lt; \beta &lt; 1, 0 &lt; \sigma &lt; 1$. Initialize any feasible $w^0$. Set $a_0 = 1$.</td>
</tr>
<tr>
<td>2: for $k = 0, 1, 2, \ldots$ do</td>
</tr>
<tr>
<td>3: Assign $a_{k+1} \leftarrow a_k$</td>
</tr>
<tr>
<td>4: If $a_{k+1}$ satisfy constraint requirement, repeatedly increase it by $a_{k+1} \leftarrow \frac{a_{k+1}}{\beta}$</td>
</tr>
<tr>
<td>5: else repeatedly decrease $a_{k+1}$ by $a_{k+1} \leftarrow a_{k+1} \cdot \beta$</td>
</tr>
<tr>
<td>6: set $w^{k+1} = P[w^k - a_k \nabla f(w^k)]$</td>
</tr>
<tr>
<td>7: end for</td>
</tr>
</tbody>
</table>

plain the need for a new cost function and why that is better than the cost function of other algorithms.

3.3.4 Other popular methods

Several other algorithms are available, most of which are based on the basic NMF algorithms explained above. Nicolas [48] has proposed some accelerated MU, Halls and PG methods and showed some improvement in the convergence time.

3.4 Initialization of decomposed matrices

The non-negative matrix factorization problem is not convex for both the factors. Hence, it is expected to have local solutions. Although existing algorithms require some initialization for these factors, starting from a point so far from the stationary points will cost many iterations to reach an optimum local solution. Therefore,
a good initial guess can save many iterations and helps to reach a better local solution. In the standard NMF algorithm \( A \) and \( S \) are initialized with random non-negative values because of its simplicity. Various efforts have focused on alternative approaches for initializing or seeding the algorithm in order to speed up or otherwise influence convergence to a desired solution. Wild et al. [50] for example, employed a spherical k-means clustering approach to initialize \( A \), but it is computationally too expensive. Boutsidis and Gallopoulos used an SVD-based initialization and show anecdotal examples of speed up in the reduction of the cost function [51]. Since there is no information about the distribution of minima, random initialization is still a good starting point. Some preprocessing may help the process, like using a local solution achieved from the MU algorithm in the ALS algorithm as an initial starting point to improve the solution. However, effective initialization remains an open problem that deserves further attention.

### 3.5 Variants and extensions of NMF

In addition to the already-mentioned traditional NMF problem, there are several variants, and extensions have been introduced from time to time by researchers. From the very beginning, it was considered that NMF has a relationship with sparsity while it provides parts-based learning, meaning that it has several zeros in the factor matrices. As a result, the sparsity of NMF has been explored in several studies [52, 53, 54, 55]. Another variant of NMF is structured NMF, where for some applications it is known that the solutions have special characteristics or structures. Such applications of structured NMF have been explored in [56, 57, 58].

Several extensions of NMF have also been proposed in many papers. Non-negative tensor factorization [11], convolutive NMF [59] and semi-NMF [60] are examples of these extensions.
Chapter 3: Non-negative matrix factorization

3.6 Applications of NMF

In the field of signal and data analysis, many practical applications deal with non-negative data. Lee and Seung [2] have shown the application of NMF in the field of image processing and text categorization. Due to the presence of purely non-negative elements in many other applications like music analysis, computational biology, financial data analysis etc., NMF has become a popular tool for low-rank matrix factorization. Many machine learning and artificial neural network algorithms use NMF as a preprocessing tool to reduce the dimensions of the training and test data sets. Some popular applications are highlighted in this section.

3.6.1 Image processing

Image data sets usually contain pixel values which are non-negative in nature. This property allows the application of NMF algorithms to images. Lee and Seung [2] demonstrated in their paper that NMF can be applied to an image dataset to extract important latent features leading to parts-based learning of images. Using this concept, NMF has application in face recognition, expression recognition and medical image analysis. Several NMF methods have been proposed since Lee and Seung [61, 62, 63, 64, 65, 66, 54]. We also apply the proposed method to image processing. Details of image processing applications of NMF have been provided in the Chapter 5 of this thesis.

3.6.2 Text and document clustering

In addition to its application to images, NMF can be applied to text data set as an unsupervised method for clustering. NMF is a natural choice for document clustering since the basis matrix $A$ provides the list of important groups or topics and the coefficient matrix $S$ can be directly used to determine the cluster membership
of each document. NMF is sometimes used for dimension reduction of text or document data sets for further application in artificial neural networks. Many studies regarding text processing using NMF can be found in the following references [67, 68, 69, 70, 71]. Details of text clustering by NMF and the application of the proposed method in this field are provided in Chapter 6 of this thesis.

### 3.6.3 Computational Biology

In recent times NMF has been utilized for the analysis of biological data. NMF is useful as an unsupervised method where there is no prior information available for analysis. A large number of studies focus on using NMF in the area of gene expression analysis and molecular pattern discovery [72, 54, 73, 74, 75, 76].

### 3.6.4 Music analysis

Sound source separation is also a successful operation of NMF. In practical scenarios multiple audio signals from different sources are mixed up very often. It is sometimes necessary to separate signals from individual sources for better analysis or manipulation for music or audio production. Usually this separation is done by utilizing prior knowledge of the source, which makes the system highly complex. Utilizing NMF, separation can be done without the help of prior knowledge. Sources are modelled as the basis matrix and the coefficient matrix contains the gains for the individual sources for specific times. Hence NMF can be useful because of its natural property of parts-based learning and the same property is used for image processing and text clustering. Several studies can be found where NMF has been used for music and audio signal analysis [77, 78, 79, 80, 81].
3.6.5 Other applications

Not specific to the above applications, NMF has been utilized for the analysis of electroencephalogram (EEG) signals [82, 83, 84], financial data analysis [85, 86], remote sensing as well as in colour and vision research [87, 88, 89, 90]. With the development of algorithms, scope for more applications is opening up and researchers are attempting to utilize the benefits of NMF in new applications.

3.7 Conclusion

NMF has attracted the attention of many researchers because of its practical applications. Although matrix factorization is a popular section of matrix algebra, factorization of non-negative matrices has not been explored in larger scale until it has been popularized by Lee and Seung [2] in 1999. Subspace based methods are used for the purpose of basis extraction, text classifications and other applications where data matrices are non-negative. NMF provided the benefit of using physically intuitive bases by factorizing the data matrix into two purely non-negative matrices. Therefore, many algorithms have developed for NMF including MU, ALS and PG based algorithms. Researchers have implemented these algorithms for variety of applications mentioned in the earlier sections. In most of the cases, NMF has outperformed subspace based methods. Therefore, in this thesis the performance of the proposed algorithm has been compared only with the NMF based algorithms. In the next chapters, the theories and mathematical explanations behind the proposed algorithm have been provided, followed by their applications in image processing, text and document classifications.
Chapter 4

Optimizing the NMF problem subject to the $\ell_1$-constraint

4.1 Introduction

In this chapter, new objective functions and constraints are introduced. In a traditional non-negative matrix factorization (NMF) problem, a non-negative matrix of dimension $n \times m$ is factorized into two reduced-rank non-negative matrices as their products. The standard NMF problem has the form

$$X = AS$$  \hspace{1cm} (4.1)

where $X \in \mathbb{R}^{n \times m}$ is the observed data matrix, $A \in \mathbb{R}^{n \times r}$ and $S \in \mathbb{R}^{r \times m}$ are the decomposed factors of $X$ and $r < \min(n,m)$ is the reduced rank. Recalling the previous chapter, existing popular NMF algorithms minimize the following objective functions to reach a feasible stationary point.

$$J(X|A, S) = ||X - AS||_F^2$$  \hspace{1cm} (4.2)

37
Having both $A$ and $S$ unknown, the above objective functions are non-convex. There is no exact solution available and existing methods search for a feasible local solution to the problem. Therefore, the solution is different for different algorithms for the same data matrix. Even the same method has different local solutions at different attempts. This thesis approaches the NMF problem using a different objective function and constraints. The aim is to explore the options available from the concepts of convex optimization and its geometrical interpretations. A new objective function has been derived to utilize the geometric properties of both NMF and convex optimization. By optimizing the proposed objective function subject to the new constraint, it is possible to find a better local solution for standard NMF problem. Empirical results also prove these claims. It is important to provide some of the definitions from convex geometry to obtain a clear understanding of the improvement of the objective function and its impact. In the next section, these definitions and visual explanations are provided.

### 4.2 Important definitions and concepts

Convex optimization is a mathematical technique with many applications. The following definitions and visualizations have significant importance in the rest of this thesis.

**Definition. Convex Set:** A set $C$ is convex if the line segment between any two points in $C$ lies in $C$ \([1]\). For example, if for any $x_1, x_2 \in C$ and any $\theta$ with $0 \leq \theta \leq 1$, we have

$$\theta x_1 + (1 - \theta) x_2 \in C$$
More simply, it can be said that a set is convex if every point in the set can be seen by every other point, along an unobstructed straight path between them. Figure 4.1 illustrates some simple examples of convex and non-convex sets in $\mathbb{R}^2$.

![Figure 4.1: At the left hexagon-shaped set is a convex set which includes its boundary shown in darker line. At the right the kidney-shaped set is a non-convex set as the line segment between two points is not contained by the set [1].](image)

If $c_1$ is a convex combination of $m$ points $x_1, x_2, \ldots, x_m$, then $c_1$ will have the form in (4.4),

$$c_1 = \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_m x_m$$  \hspace{1cm} (4.4)

where $\theta_1 + \ldots + \theta_m = 1$ and $\theta_i \geq 0, i = 1, \ldots, m$. The convex set $C$ is the set of all such combinations.

**Definition. Convex Hull:** The convex hull of a set $C$, referred to as $\text{conv} \ C$, is the set of all convex combinations of points in $C$ [1].

$$\text{conv} \ C = \{\theta_1 x_1 + \ldots + \theta_m x_m\}$$
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

where $x_i \in C, \theta_i \geq 0, i = 1, \ldots, m$ and $\theta_1 + \ldots + \theta_m = 1$

Figure 4.2 shows the illustrations for the definition of convex hull. Convex hull of any set $C$ is the smallest set that contains $C$.

![Convex Hull Illustration]

Figure 4.2: At the left the shaded pentagon is the convex hull of fifteen points shown as dots. The shaded illustration at the right is the convex hull of the kidney shaped set shown in Figure 4.1.

**Definition. Cone:** A set $C$ is called a cone or non-negative homogeneous, if for every $x \in C$ and $\theta \geq 0$ we have $\theta x \in C$.

**Definition. Convex Cone:** A set $C$ is called a convex cone if it is a cone and convex. This means that for any $x_1, x_2 \in C$ and $\theta_1, \theta_2 \geq 0$ the following relation holds

$$\theta_1 x_1 + \theta_2 x_2 \in C$$

A set $C$ is the convex cone if and only if it contains all conic combinations of its elements.

**Definition. Conic Hull:** The Conic Hull of a set $C$ is the set of all conic combinations of points in $C$. It is the smallest cone that contains $C$. Figure 4.3 illustrates the
conic hull for the convex sets shown in Figure 4.2.

![Figure 4.3: Conic hull for the convex set shown in Figure(4.2) [1].](image)

A conic hull can also be referred to as a Simplicial cone and is denoted by $\mathcal{S}$ in the rest of the thesis.

### 4.3 Geometric interpretation of NMF

By combining the definitions and their properties explained in the previous definitions, we now interpret an NMF problem from the geometric perspective. Assume that $X$ is originally a combination of $W$ and $H$ such that

$$X = WH \quad (4.5)$$

where $W \in \mathbb{R}^{n \times r}_+$ and $H \in \mathbb{R}^{r \times m}_+$. Looking at the column of $X$, one sees that each is approximated by a conic combination of $r$ non-negative basis vectors, which are the columns of $W$.

$$x_i = \sum_{j=1}^{r} w_j h_{ji} \quad (4.6)$$
here $x_i$ ($i = 1, \ldots, m$) is the column of $X$, $w_j$ is the column of $W$ and $h_{ji}$ is the element of $j^{th}$ row and $i^{th}$ column of $H$. If the columns of observed data matrix $X$ are considered to represent individual points in an $n$-dimensional co-ordinate system, as Donoho and Stodden [19] pointed out, it can be said that the simplicial cone $\mathcal{S}_W$, created by the column vectors of $W$, will contain all the data points.

$$\{x_i, i = 1, 2, \ldots, m\} \subseteq \mathcal{S}_W$$  \hspace{1cm} (4.7)

here $\subseteq$ symbolizes the subset of any set. The columns of $W$ represent the extreme rays of the cone.

Figure 4.4: Simplicial cone containing data points

Figure 4.4 is a visualization of (4.7) for $n=3$. Here the simplicial cone or convex cone created by the columns of $W$ is displayed as red lines and the blue dots represent the actual data vectors i.e. columns of $X$. Referring to the earlier explanation, while solving any NMF problem with no information about the factors for a given dataset $X$, if $X=AS$ is an approximation of the NMF problem, there exists another solution $X=\tilde{A}\tilde{S}$, where the simplicial cone created by $A$ will contain all the data vectors, as well as the simplicial cone created by $\tilde{A}$. Similarly, there will be several
other convex cones, which will contain all the data vectors of $X$. For example,

$$\{x_i, i = 1, 2, \ldots, m\} \subseteq \mathcal{S}_A$$

$$\{x_i, i = 1, 2, \ldots, m\} \subseteq \mathcal{S}_{\tilde{A}}$$

![Figure 4.5: Example of multiple convex cone available containing data points](image)

In Figure 4.5, we can observe this scenario of multiple simplicial cones where black dots are the original data points, blue lines represent the simplicial cone created by $A$ and the black lines represent the simplicial cone created by $\tilde{A}$.

Following the definitions of convex hull and convex conic hull, it can be inferred that the smallest cone that will contain all the data points will provide the optimal result in this case. The present thesis searches for that optimum simplicial cone. Several measuring parameters are available to optimize the simplicial cone. The cone can be optimized with respect to the area of a certain facet of the cone.
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

### 4.4 Non-uniqueness of solution

Unfortunately, as there is no unique solution for NMF problems, there can be several situations where these parameters fail to provide meaningful optimization. To solve an NMF problem, assume that $r = \text{rank}(X)$, thus both $A$ and $S$ are full rank. If there exists a full rank matrix $T \in \mathbb{R}_+^{r \times r}$ such that $\tilde{A} = AT \succeq 0$ and $\tilde{S} = T^{-1}S \succeq 0$ then there exists a solution $X = AS = \tilde{A}\tilde{S}$. There are several choices of $T$ possible where $T$ will be a non-negative generalized permutation matrix. For example in $\mathbb{R}^3$:

$$T = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 3 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

As multiple combinations are available for producing such $T$, more than one solution is available for a particular NMF problem. It can be said that NMF suffers from indeterminate scales.

In Figure 4.6 it has been shown that there are several simplicial cones available with the same minimum area, volume or aperture. Dotted blue, green and red lines represent the facet on the $\ell_1$-plane of a particular cone that encloses the original data vectors represented by the black circles. Blue, green and red solid circles represent the corner vector of the respective simplicial cone. In these scenarios, it is difficult to determine the best convex cone where an infinite number of solutions is possible with the same shape or size.

A certain locking situation occurs while minimizing these parameters [91]. Figure 4.7 shows such a locking situation and an unlocking procedure needs to be introduced. In addition, there can be multiple simplicial cones that have the same value as these certain measurement parameters.
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

Figure 4.6: Visualization of simplicial cone on a unitized $\ell_1$-space. We can see three different orientations of simplicial cones with the same shape and size for the same data matrix $X$ (Many other orientations are also possible with the same shape and size)

After exploring these geometric properties and difficulties, certain relations can be formed between the Frobenius norm of data and factor matrices on a unitized $\ell_1$-plane and the geometric measurement parameters like area, volume or aperture. When all the data vectors are unitized to be plotted on the $\ell_1$-plane to find $A$, minimizing the Frobenius norm of $A$ will provide the minimum simplicial cone. In addition, if the Frobenius norm of $S$ is maximized simultaneously, it provides better
convergence and accuracy in terms of error and an optimum simplicial cone. In the next section, the mathematical and geometric explanations of the relation of the Frobenius norm with geometric parameters are provided. Later, the formulation of a new objective function is also presented.

4.5 Optimization of Frobenius norm with constrained \( \ell_1 \)-norm to find the feasible solution.

While dealing with the data matrix \( X \), it is assumed that each column of this matrix represents a point in the \( n \)-dimensional space. Initially these vectors are of variable lengths. To utilize certain geometric properties, it is necessary to introduce some constraints to the length of the vectors. Such constraints unitize the vectors so that sum of their elements is equal to one. Therefore, the \( \ell_1 \)-norm of each vectors as well as the whole matrix will be equal to one. Figure 4.8 shows the effect of such transformation. Although the \( \ell_1 \)-norm will be constrained, the \( \ell_2 \)-norm, or simply the Euclidean lengths of the vectors, will still be variable.
Therefore, to achieve a new objective function, the data matrix $X$ is unitized as

$$\sum_{i=1}^{n} X_{ij} = 1 \quad \text{for} \quad j = 1, 2, \ldots, m$$

where $X_{ij}$ is the $i^{th}$ element of the $j^{th}$ column of $X$.

By imposing this constraint on the data matrix, it is further assumed that the solutions of the NMF problem $A$ and $S$ will have the same unitized form such as

$$\sum_{i=1}^{n} A_{ij} = 1 \quad \text{for} \quad j = 1, 2, \ldots, r$$
\[
\sum_{i=1}^{r} S_{ij} = 1 \quad \text{for} \quad j = 1, 2, \ldots, m
\]

where \( A_{ij} \) is the \( i \)th element of \( j \)th column of \( A \) and where \( S_{ij} \) is the \( i \)th element of \( j \)th column of \( S \).

Before proceeding, first we show the properties of the \( \ell_2 \) norm of a vector in a unitized \( \ell_1 \)-space. Assume \( u_1, u_2, u_3, u_4 \) and \( u_5 \) are unitized vectors such that

\[
\|u_1\|_1 = \|u_2\|_1 = \|u_3\|_1 = \|u_4\|_1 = \|u_5\|_1 = 1 \quad (4.8)
\]

and \( u_1 \) is normal to the \( \ell_1 \)-plane from the centre, therefore

\[
\|u_1\|_1 = \|u_1\|_2 = 1 \quad (4.9)
\]

The above relations are presented in Figure 4.9. It can be observed that when any vector \( u_i \) for any positive integer \( i \) goes away from the normal vector, the \( \ell_2 \) norm of that vector increases. In the case of Figure 4.9, the relation between \( \ell_2 \) norms of the vectors will be

\[
\|u_5\|_2 > \|u_3\|_2 > \|u_2\|_2 > \|u_4\|_2 > \|u_1\|_2 \quad (4.10)
\]

As mentioned earlier, there are several solutions that satisfy (4.1). If \( \tilde{A} \) and \( \tilde{S} \) are also the factors of \( X \), then

\[
X = \tilde{A}\tilde{S} \quad (4.11)
\]

Let \( A_i \) be the \( i \)th column vector of \( A \) and \( \tilde{A}_i \) be the \( i \)th column vector of \( \tilde{A} \) where \( i = 1, 2, 3, \ldots, n \). For simplicity, assume \( n = 2 \). Figure 4.10 shows an example of this statement where red lines represent the column vectors of \( A \) and green lines...
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

Figure 4.9: Unitized vectors in $\ell_1$-space.

Figure 4.10: Representation of column vectors of $\mathbf{X}$, $\mathbf{A}$ and $\tilde{\mathbf{A}}$ in $\mathbb{R}_+^2$.
represent the column vectors of $\tilde{A}$. Both of the matrices have created the respective convex hull which encloses the original data vectors $X_i$ where $i = 1, 2, 3, \ldots, m$. Here $m = 4$. $u$ is the normal vector to the $\ell_1$-plane from the centre which is also the geometrical centre vector of the original data matrix $X$. From the figure, it can be observed that

$$\|\tilde{A}_1\|_2 < \|A_1\|_2$$

$$\|\tilde{A}_2\|_2 < \|A_2\|_2$$

(4.12)

The above equation can be rewritten as

$$\|\tilde{A}_1\|_2 + \|\tilde{A}_2\|_2 < \|A_1\|_2 + \|A_2\|_2$$

(4.13)

Instead of using column vectors, we can re-formulate the above equation in matrix form

$$\|\tilde{A}\|_F < \|A\|_F$$

(4.14)

Without loss of generality, $\tilde{A}$ will be a better solution than $A$ if and only if $\|\tilde{A}\|_F \leq \|A\|_F$, and the statement can be proved by observation of Figure 4.10. The column vectors of $\tilde{A}$ will have less distance from the normal vector $u$ than the column vectors of $A$. Therefore, $\tilde{A}$ will enclose the column vectors of $X$ more accurately than $A$.

From the above explanation, if the Frobenius norm of $A$ is minimized until the simplicial cone $S_A$ contains all the points of dataset $X$, the algorithm can reach an optimum solution of the NMF problem. Furthermore, the weight matrix is also unknown. To find the weight matrix $S$, another optimization problem is solved as alternately.
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

While reducing the Frobenius norm of the basis matrix provides a better solution, the Frobenius norm of the coefficient matrix increases to keep the norm of the data matrix $X$ constant. Geometrically the coefficient matrix expands or spreads its column vectors, while the basis matrix squeezes inwards until it cannot enclose all the original data vectors. If $\|\bar{A}\|_F \leq \|A_1\|_F$ then $\|\bar{S}\|_F \geq \|S\|_F$.

![Diagram](image)

Figure 4.11: Effect of basis matrix over coefficient matrix. *Left* illustration shows a larger simplicial cone formed by basis matrix. *Right* illustration shows comparatively smaller simplicial cone with sparser (higher Frobenius norm) coefficient matrix.

Figure 4.11 shows the relation between basis matrix $A$ and coefficient matrix $S$. A larger Frobenius norm means sparser or more spread $S$.

Therefore, rather than only solving (4.16), we concentrate on minimizing the Frobenius norm of $A$ and maximizing the Frobenius norm of $S$. Here, maximizing a function is actually minimizing the negative of that function. Therefore, instead of optimizing (4.16), (4.15) has been optimized.
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

\[
\begin{align*}
\text{minimize} & \quad ||A||^2_F - ||S||^2_F \\
\text{subject to} & \quad X = AS \\
& \quad A \succeq 0; ||A||_1 = 1 \\
& \quad S \succeq 0; ||S||_1 = 1
\end{align*}
\] (4.15)

(4.15) can be written as

\[
\begin{align*}
\text{minimize} & \quad ||A||^2_F - ||S||^2_F + \lambda ||X - AS||^2_F \\
\text{subject to} & \quad A \succeq 0; ||a_i||_1 = 1 \quad \text{for } i = 1, 2, \ldots, r \\
& \quad S \succeq 0; ||s_j||_1 = 1 \quad \text{for } j = 1, 2, \ldots, m
\end{align*}
\] (4.16)

4.6 Algorithm

The actual objective function presented in 4.16 is a non-convex problem, as both $A$ and $S$ are unknown. To utilize the advantages of convex optimization the alternating solving method has been adopted. Hence the problem is formulated into two steps defined in (4.17) and (4.18).

Step 1:

\[
\begin{align*}
\text{minimize} & \quad ||A||^2_F - ||S||^2_F + \lambda ||X - AS||^2_F \\
\text{subject to} & \quad A \succeq 0; ||a_i||_1 = 1 \quad \text{for } i = 1, 2, \ldots, r
\end{align*}
\] (4.17)
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

Step 2:

$$\min_S \|A\|_F^2 - \|S\|_F^2 + \lambda \|X - AS\|_F^2$$

subject to

$$S \succeq 0; \|s_j\|_1 = 1 \quad \text{for } j = 1, 2, \ldots, m$$

In this alternating solution approach, the sub-problems become convex optimization problems. There are several convex optimization techniques available for solving the sub-problems, including the gradient-based method, the interior point method and the least squares method. Some recent methods are available which improve the optimization. Initially the software packages, CVX and Sedumi for convex optimization were utilized to optimize the sub-problems in the proposed algorithm. The proposed algorithm will be referred to as $\text{MinMaxNMF}$ for rest of the thesis.

A pseudo code for the algorithm is presented in Table 4.1

<table>
<thead>
<tr>
<th>New Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Table 4.1: Summary of our Algorithm</strong></td>
</tr>
</tbody>
</table>
| 1: Initialize $S^0 \succeq 0$  
2: for $k = 0, 1, 2, \ldots$ do  
3: $A^{k+1} = \arg \min_{A^{k+1}} \|A^{k+1}\|_F^2 - \|S^k\|_F^2 + \lambda \|X - A^{k+1}S^k\|_F^2$  
subject to  
$A^{k+1} \succeq 0; \|a_{i}^{k+1}\|_1 = 1 \quad \text{for } i = 1, 2, \ldots, r$  
$S^{k+1} = \arg \min_{S^{k+1}} \|S^{k+1}\|_F^2 - \|S^{k+1}\|_F^2 + \lambda \|X - A^{k+1}S^{k+1}\|_F^2$  
subject to  
$S^{k+1} \succeq 0; \|s_j^{k+1}\|_1 = 1 \quad \text{for } j = 1, 2, \ldots, r$  
4: if $\|X - A^{k+1}S^{k+1}\|_F^2 \leq Tol$ or $k > maxiter$ then Return  
5: end for |
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

4.7 Stopping Criteria

For any iterative optimization algorithm, it is important to define stopping conditions to terminate the algorithm. As NMF algorithms are usually converge to only local minima, the stopping criteria are a very important factor. Many versions of multiplicative update rules and alternating least squares algorithms need to be stopped by the user, by specifying the iterations number or time limit. Some researchers check the differences between recent iterations to decide the termination. Such stopping conditions do not provide information about whether the solution is close to a stationary point or not. Standard conditions should be included in NMF algorithms to check whether the solution has reached the stationary point and the iteration or time limits. In alternating optimization methods, stopping conditions for the sub-problems are also needed.

In this thesis, the proposed algorithm checks several parameters to define the stopping conditions. Following traditional methods, it also includes the iteration and time limit to terminate the algorithm, which gives more user control over the solution. Minimum error and similar solution conditions are introduced to evaluate whether the algorithm should be terminated.

4.7.1 Minimum error condition

The cost function of MinMaxNMF algorithm is defined in (4.16) where it minimizes the Euclidean error in terms of the Frobenius norm along with the norms of $A$ and $S$. While alternatively optimizing the problems of (4.17) and (4.18), MinMaxNMF evaluates the error defined in equation (4.19) in every iteration.

$$e_k = \|X - A_k S_k\|_F^2$$  \hspace{1cm} (4.19)
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

where $e_k$ means the $k^{th}$ error function and $A_k, S_k$ are the $k^{th}$ solution. In ideal conditions, when $e_k$ reaches zero, the algorithm would terminate. In practical scenarios, it will take many iterations and time to reach zero error. Sometimes it can cause infinite iteration as usually there is no unique solution for NMF problems. Hence we introduce a minimum error value $e_{min}$ to terminate the algorithm such that the algorithm will terminate when

$$e_k \leq e_{min} \tag{4.20}$$

Similar error criteria are also imposed for solving the sub problems as well.

4.7.2 Similar solution condition

If the solution reaches a stationary point, $A_k$ and $S_k$ will not change, although the algorithm may continue its iteration while the $e_k \geq e_{min}$. For this situation, we introduce another criterion to terminate the iteration. While solving the sub-problem in (4.17), if $A_{k+1} = A_k$ then the internal iteration will stop. Same event occur for the sub-problem in (4.18) when $S_{k+1} = S_k$

Now for global problems, the outer iteration will stop when both

$$A_{k+1} = A_k \text{ and } S_{k+1} = S_k$$

4.8 Experimental results

In this section several empirical results are presented, produced by the proposed MinMaxNMF algorithm. Before their application to the practical problems such as image processing and text classification, to prove the accuracy of the MinMaxNMF algorithm and compare the performance, we apply the algorithms to synthetic data. The final error, simplicial cone, convergence performance and variation in results
are analyzed for different important parameters, like the value of \( r \). Results from multiplicative update rules (MU), hierarchical alternating least squares (HALS) and projected gradient (PG) algorithms are compared with those of MinMaxNMF. For comparison, the following error function is introduced which is mentioned in (4.21), as initially individual algorithms have their own cost functions.

\[
Error = \| X - AS \|_F^2
\]  

(4.21)

To visualize the convex hull created by the solutions from the algorithms, for the first few simulations we use a comparatively small data set with \( n = 3, m = 32, r = 3 \). The data matrix \( X \) is then generated randomly with all non-negative values. Normalization of the data was done to keep the \( \ell_1 \)-norm to 1. We run the simulations with maximum iteration of 100 for all the algorithms. Table 4.2 shows the comparison among the algorithm in terms of error. It can be seen that the final error of the MU algorithm is much higher than other algorithms. The HALS and PG algorithms show very little difference, but better than the MU algorithm. MinMaxNMF produces results with least error in this case.

Table 4.2: Error comparison between NMF algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>( 4.8 \times 10^{-2} )</td>
</tr>
<tr>
<td>HALS</td>
<td>( 8.34 \times 10^{-4} )</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>( 5.1 \times 10^{-4} )</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>( 3.8 \times 10^{-6} )</td>
</tr>
</tbody>
</table>

Figure 4.12 demonstrates the convex hull created by each algorithm. Blue dots represent data vectors projected on a plane in the \( \ell_1 \)-space. Green, red, yellow and black respectively represent the conic hull created by the basis matrix solved using MU, HALS, PG and MinMaxNMF algorithms. It is observed that the conic hull cre-
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

Figure 4.12: Convex conic hulls created by the basis matrices

ated by the MU algorithm fails to enclose all the data points of $\mathbf{X}$. Other algorithms show nearly similar convex hulls as to that of the MinMaxNMF algorithm.

In the next experiment with the same set-up we show how the algorithms converge to their final error. While producing the optimum result, the number of iterations is also a major parameter that determines the performance of any algorithm. Figure 4.13 demonstrates how the error is decreasing in every iteration for each algorithm. It shows that the decrease in error for the MinMaxNMF algorithm is faster and better. It reduces its error very quickly in the first few iterations and reduction in error is less in the later iterations. The HALS and PG algorithms show a similar drop in error, but the error in every iteration is always higher than that in the MinMaxNMF algorithm. The convergence performance of the MU algorithm is slowest
Chapter 4: Optimizing NMF problem subject to the $\ell_1$-constraint

of all algorithms.

![Error vs Iterations for MU, HALS, PG, and MinMaxNMF](image)

Figure 4.13: Decrease of error in every iteration for MU, HALS, PG and MinMaxNMF algorithms.

For the rest of the simulations, we vary the value of factorization rank, $r$, to evaluate performance according to the change. We set the values of $n = 100$, $m = 200$ while producing the data matrix with random non-negative values. The matrix is unitized as described in previous sections. We vary $r = 5, 10, 20, 50$ and observe the final errors for 100 iterations. Along with the MinMaxNMF algorithm, the results from MU, HALS, PG algorithms are listed in Table 4.3. For $r = 5$, $r = 10$ and $r = 20$ the errors are close to each other, while MinMaxNMF has the least error. When $r = 50$, the PG algorithm fails to perform well and has a much higher error than all other algorithms. MinMaxNMF performs better in this case also.
Table 4.3: Error comparison with variable $r$

<table>
<thead>
<tr>
<th>$r$</th>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Multiplicative Update (MU)</td>
<td>0.6064</td>
</tr>
<tr>
<td>5</td>
<td>HALS</td>
<td>0.6052</td>
</tr>
<tr>
<td>5</td>
<td>Projected Gradient (PG)</td>
<td>0.6072</td>
</tr>
<tr>
<td>5</td>
<td>MinMaxNMF</td>
<td>0.6051</td>
</tr>
<tr>
<td>10</td>
<td>Multiplicative Update (MU)</td>
<td>0.5139</td>
</tr>
<tr>
<td>10</td>
<td>HALS</td>
<td>0.5139</td>
</tr>
<tr>
<td>10</td>
<td>Projected Gradient (PG)</td>
<td>0.5142</td>
</tr>
<tr>
<td>10</td>
<td>MinMaxNMF</td>
<td>0.5039</td>
</tr>
<tr>
<td>20</td>
<td>Multiplicative Update (MU)</td>
<td>0.4298</td>
</tr>
<tr>
<td>20</td>
<td>HALS</td>
<td>0.4205</td>
</tr>
<tr>
<td>20</td>
<td>Projected Gradient (PG)</td>
<td>1.6524</td>
</tr>
<tr>
<td>20</td>
<td>MinMaxNMF</td>
<td>0.4201</td>
</tr>
<tr>
<td>50</td>
<td>Multiplicative Update (MU)</td>
<td>0.806</td>
</tr>
<tr>
<td>50</td>
<td>HALS</td>
<td>0.3489</td>
</tr>
<tr>
<td>50</td>
<td>Projected Gradient (PG)</td>
<td>1.6365</td>
</tr>
<tr>
<td>50</td>
<td>MinMaxNMF</td>
<td>0.3484</td>
</tr>
</tbody>
</table>

In Figure 4.14, the decrease of error in every iteration is observed for MinMaxNMF algorithm. These experimental results show that the MinMaxNMF algorithm outperforms other methods. Results for the synthetic data inspired us to apply this algorithm to practical applications.
4.9 Conclusion

This chapter has explored the NMF problem more deeply and proposed a novel objective function for it. Geometric theories and concepts behind the convex optimization and NMF problem have led to the introduction of a new cost function. Although there is no unique solution available for any NMF problem we have discussed, it is possible to find a solution which is a local minimum. In many cases, these local minima provide useful results. The proposed algorithm utilizes the benefit of the new cost function while optimizing it. It minimizes the Frobenius norm of the basis.
matrix and maximizes the Frobenius norm of the coefficient matrix alternately in every iteration. As the cost function itself is non-convex for both the factors, it is convex when one factor is fixed. The final error and the number of iterations to converge are the important parameters for the evaluation of the performance of an algorithm. A number of simulations on synthetic data provide satisfactory results. The MinMaxNMF algorithm showed least error and fast convergence, compared with multiplicative update, hierarchical alternating least squares and projected gradient algorithms.
Chapter 5

Application to parts-based learning of images

5.1 Introduction

In the field of image processing two types of concepts are available. One is learning the images as a whole, and the other is to learn the images from distinguishable parts or features from the whole image. Although there is substantial psychological and physiological evidence of parts-based representation in the brain, it is a challenging job for any computer to learn how to utilize the parts-based image representation [2]. Modern research has focused on parts-based learning of images, where the intuitive concept that the whole is constituted from the combination of the parts, is utilized.

Image and signal processing, especially face recognition, facial expression recognition, pattern recognition, medical imaging and musical and speech data representation are examples of applications where parts-based learning provides important information. Previously developed methods like principal component analysis
(PCA) and independent component analysis (ICA) consider the image or signal as a whole \cite{92, 42}. These methods and their extensions have certain advantages, but they also have limitations. Most of the above applications naturally hold the property of non-negativity in the data, and the previously-mentioned methods do not consider this particular constraint. Considering this fact, non-negative matrix factorization (NMF)-based algorithms have advantages over other algorithms. Again, NMF algorithms provide parts-based information by their nature without any further processing.

In this chapter, we apply our novel NMF method (MinMaxNMF) for parts-based imaging with applications to face detection, expression recognition and medical imaging and compare the results with other traditional and NMF-based algorithms. In the later sections we provide information regarding the image data, the usefulness of NMF for parts-based learning and experimental results supporting our claim that our algorithm performs better than other methods.

\section{Previous methods for image processing}

Subspace-based methods have demonstrated their success in the field of visual recognition and detection like face recognition, face detection, object detection and tracking. PCA, ICA, Fisher Linear Discriminant Analysis (FLDA) and NMF are examples of subspace-based methods \cite{93, 94}. These methods learn to represent a face as a linear combination of basis images, but in different ways \cite{95}. The basis images of PCA are orthogonal and have a statistical interpretation as the directions of largest variance. ICA is a linear nonorthogonal transform that provides a representation in which unknown linear mixtures of multidimensional random variables are made as statistically independent as possible \cite{96}. FLDA searches to find a linear
transformation that can maximize the between-class scatter and minimize within-
class scatter [97]. NMF provides the two non-negative matrices and learns images
as parts [2].

5.3 NMF for parts-based learning

As explained in the previous chapters, NMF decomposes a non-negative matrix $X$
into two different non-negative matrices, $A$ and $S$, respectively as presented in (5.1):

$$X = AS$$  \hspace{1cm} (5.1)

where $X \in \mathbb{R}^{n \times m}$ is the observed data matrix, $A \in \mathbb{R}^{n \times r}$ and $S \in \mathbb{R}^{r \times m}$. $A$ can be
termed the basis matrix and $S$ the coefficient matrix. In the field of image process-
ing, a set of images can be considered as the data matrix $X$ whose columns are the
vectorized images. When we decompose this matrix into two matrices using NMF,
basis matrix $A$ contains the parts of the images, to be particular different facial fea-
tures or objects in an image. The co-efficient matrix $S$ contains the numerical values
representing the importance of those features in each image. This concept can be
visualized using Figure 5.1.

In contrast to PCA, as subtraction does not occur in NMF, the non-negative con-
straint is compatible with the intuitive concept of combining parts to form a whole
image, for example of a face. That is why NMF learns a parts-based representation
[98].

The dependencies between image pixels and the co-efficient or encoding vari-
able are represented in Figure 5.2. The top nodes represent co-efficients $s_1, \ldots, s_m$
(columns of $S$), and the bottom nodes represent the images $x_1 \ldots x_m$ (columns of $X$).
The non negative value $a_{ij}$ characterizes the extent the influence that the $j^{th}$ coefficient variable $s_j$ has on the $i^{th}$ image pixel $x_i$. Because of the non negativity of $a_{ij}$, the image pixels in the same part of the face image will be co-activated when a part is present and NMF learns by adapting $a_{ij}$ to generate an optimal approximation.

In the next few sections we provide details about the application of our algorithm to facial recognition and medical imaging for parts-based learning and discuss its advantages. We provide information regarding the data sets and performance measures and compare the results with those using traditional and NMF-based algorithms.

### 5.4 MinMaxNMF for parts-based imaging

The proposed algorithm MinMaxNMF is applied to several image processing applications. Face recognition using basis extraction and positron emission tomography (PET) image are the two examples presented in this chapter. The results have been
5.4.1 Facial Recognition

This section provides the results from feature extraction and parts-based learning experiments on CBCL [99] and AT&T [100] databases to evaluate the performance of the proposed MinMaxNMF algorithm. The performance of the proposed MinMaxNMF algorithm is compared with some of the NMF based methods, including MU, HALS and PG. Popular subspace based facial recognition algorithms are used to compare the recognition performance [93].

MIT CBCL Face database

The proposed MinMaxNMF algorithm was tested for the well known facial reconstruction example from Lee and Seung [2]. The CBCL facial data set was used for simulation, and the database was of 2429 gray-scale facial images, each consisting
of $19 \times 19$ pixels. Figure 5.3 represents some of the face images from the CBCL face database. These images are vectorized to form an $n \times m$ data matrix $X$, where $n = 361$ and $m = 2429$. For compatibility, the data matrix is unitized with individual column summing to one.

![Figure 5.3: Sample images from CBCL face database](image)

This data matrix was used for NMF using MU, HALS, PG and the proposed Min-MaxNMF algorithm. All the methods found approximate factorizations of the form $X \approx AS$, but with different types of $A$ and $S$. As presented in Figure 5.4, showing the $7 \times 7$ montages, each method has learned a set of $r = 49$ basis images. From the results and reconstructions, it is clear that our algorithm outperforms other algorithms. The final error for our algorithm is also less than others, as summarized in Table 5.1.

**Table 5.1: Final error of NMF algorithms for CBCL face database**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>0.0734</td>
</tr>
<tr>
<td>HALS</td>
<td>0.0566</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>0.0571</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>0.0478</td>
</tr>
</tbody>
</table>
Figure 5.4: Basis extract from CBCL face database by (a) MU, (b) HALS, (c) PG and (d) MinMaxNMF algorithm

Figure 5.5 shows the convergence of the algorithms for this simulation. It shows that the MU algorithm converges very slow as it was noticed while experimenting with synthetic data. Other algorithms show similar convergence, however, MinMaxNMF produces least error. In Figure 5.9, the reconstructed images are presented. Those images are very similar to the original images (showed in Figure 5.3).

**AT&T face database**

The AT&T face database (formerly known as the Cambridge ORL face database) consists of 400 images of 40 individuals. There are 10 images of each individual. The
Images were taken at different times with variable light, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). The faces are in an upright position in frontal view with tolerance for some tilting and rotation of the faces up to 20°. The original images have resolution of $112 \times 92$ with 256 gray levels.

For our experiment, all the images were manually cropped to $56 \times 46$ for efficiency and compatibility of all algorithms. Figure 5.7 shows some example images from the AT&T face database. The images were vectorized to form the data matrix $X$ where each column represents one image. We further normalized the data matrix...
Figure 5.6: Samples from the reconstructed images of CBCL face database for (a) MU, (b) PG, (c) HALS and (d) MinMaxNMF algorithms.

to make the column sum equal to one. All the experimental methods were then run to factorize the data set into $A$ and $S$, where $A$ represents the basis of the images. Figure 5.8 shows the basis extracted by different algorithm for $r = 20$. The images show the basis extracted by the MinMaxNMF is additive and more spatially localized than the other algorithms. Table 5.2 shows the final error for different algorithms.

Figure 5.9 shows the convergence of the algorithms for this simulation. MU converges slower, while other algorithms converge faster and have similar convergence.

All the algorithms were then compared for face recognition on the AT&T face
database. For face recognition, it is required to derive the co-efficients for the test images. If $x_{test} \in \mathbb{R}_+^n$ represents a test image in vectorized form, then the coefficient vector $s_{test} \in \mathbb{R}_+^r$ can be derived from the following equation [42]:

$$s_{test} = A^\dagger x_{test}$$

(5.2)

where $A^\dagger$ is the pseudo inverse of the basis matrix $A$. This test co-efficient vector is compared with the training co-efficient vectors generated by the NMF algorithms for training dataset. The accuracy of face detection is measured from the ratio of
Figure 5.8: Basis extracted from AT&T face database by (a) MU, (b) HALS, (c) PG and (d) MinMaxNMF algorithm

the number of correctly detected faces and the number of total test faces. The rates of accuracy in facial recognition for these algorithms are presented in Table 5.3.
Figure 5.9: Decrease of error in every iteration for MU, HALS, PG and MinMaxNMF algorithms while extracting basis of AT&T face database.

Table 5.3: Face detection accuracy using basis extracted by NMF algorithms for AT&T face database

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>55</td>
</tr>
<tr>
<td>HALS</td>
<td>72</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>69</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>78</td>
</tr>
</tbody>
</table>
5.4.2 Medical Imaging

Different physiological information can be retrieved from PET[101]. PET is a imaging technique for measuring concentrations of positron-emitting radio-isotopes within the tissue of living objects. It provides information from regional concentrations, which helps to model the radioactivity in a particular organ [102]. In addition to this, time activity curve (TAC) can be obtained from PET by applying NMF which provide information about myocardial blood flow (MBF). In this section, a PET raw image of mouse brain is used to detect various regions of brain activity and TAC. Figure 5.10 shows the TAC extracted by MinMaxNMF, of four sources that indicates the MBF. In Figure 5.11 and 5.12, the spatial images are presented for these four sources, which are extracted by MinMaxNMF as basis matrix.

Many information can be observed from these basis extracted by the NMF from PET images. Medical experts can interpret them for diagnosis of diseases.
Figure 5.10: TAC for four sources, extracted from raw PET image using MinMaxNMF algorithm.

### 5.5 Conclusion

In computer vision and image processing, parts-based learning has great importance. From facial recognition to object detection, parts-based learning provides detailed information regarding an image in terms of its important features and the respective co-efficients. Traditional facial recognition, expression recognition and object detection software uses the techniques of machine learning, where the computer learns to recognize specific parts or objects after training with thousands of images of the same object. This process requires a huge database and is comparatively time consuming.
Chapter 5: Application to parts-based learning of images

Non-negative matrix factorization enables unsupervised learning. Using this method, the computer does not need to know about thousands of images to extract the important features from an image. Rather, it classifies different parts of images, for example the ears, nose, eyes in a human face, automatically while factorizing the data. This inherent capability of NMF enables vast application in computer vision and image processing.

The proposed MinMaxNMF algorithm provides a new cost function to minimize where the Frobenius norm of the basis matrix that contains the features of the images is minimized and the Frobenius norm of the co-efficient matrix is maximized alternately in every iteration. The details of the algorithm were explained in Chapter four. In this chapter the proposed algorithm is applied to the field of image processing and parts-based learning. The results are discussed and compared with other NMF algorithms.

At first the MIT CBCL face database was used for simulation, this database is used in the pioneer article by Lee and Seung [2] to provide the proof that NMF has the capability of parts-based learning. We used the same dataset to compare our algorithm with popular NMF algorithms. After observing the results it can be stated that the performance of MinMaxNMF shows significant improvements in terms of classification of features, final error and number of iterations taken to converge to minimum error.

The AT&T face database was also used for simulation purposes to demonstrate the versatility of the proposed algorithm. Facial recognition has been run after basis extraction, and again the proposed algorithm outperformed the popular NMF algorithms.

The MinMaxNMF algorithm was also applied to the analysis of medical images,
and in this chapter we present an example of PET brain image analysis to extract features from the raw image to identify certain problems in the brain. The proposed algorithm has shown promising result in medical image analysis compared to other NMF algorithms.

As a novel technique the proposed NMF algorithm has shown its capability of parts-based learning of images. This chapter has summarized the results and comparisons to provide the proof of performance.
Figure 5.11: Spatial images of four different sources extracted as basis from raw PET image using MinMaxNMF algorithm (source 1 and 2)
Chapter 5: Application to parts-based learning of images

Figure 5.12: Spatial images of four different sources extracted as basis from raw PET image using MinMaxNMF algorithm (source 3 and 4)
Chapter 6

Text and document clustering using \( \ell_1 \)-constrained NMF

6.1 Introduction

Text clustering and data mining refer to representing text or document according to the trends, patterns or similarities within them. It is often necessary to classify a given set of texts or documents into groups or clusters based on similarities of content. It is very difficult to organize large volume of data manually. Computerized clustering and grouping of such data greatly reduce time consumption. When the categories or topics are predefined for classification, it is considered as a supervised process. There are several methods in use which provide satisfactory results in the case of supervised processes. However, when little or no information is provided, it is very difficult to classify the data into groups. This process is called unsupervised learning, where the structure or organization of the data provided is the only valid information.
6.2 Popular methods for clustering

Many supervised and unsupervised methods are developed for text and document clustering, as it has been studied widely for a long time. Traditional techniques usually focused on quantitative data \([103, 104, 105, 106]\), where the properties of data are numeric \([107]\). There are also many studies on categorical data \([108, 109, 110]\). Several implementations of clustering algorithms can be found in \([111]\). Clustering algorithms can be divided into several types depending on their techniques. For example, agglomerative clustering algorithms, partitioning algorithms and EM-algorithms. Clustering using feature extraction has become popular recently and few methods are being popular such as Latent Semantic Indexing (LSI) and Probabilistic Latent Semantic Analysis (PLSA) \([107]\). Recently, non-negative matrix factorization (NMF) has attracted the researchers for text and document clustering based on feature extraction. NMF is also considered as unsupervised clustering method.

6.3 Clustering using NMF

NMF provides unsupervised document clustering, where a set of documents is decomposed into two matrices containing the information on important categories in the basis matrix and the weight values of that category for a particular document can be found in the co-efficient matrix. Recalling the previous chapters, we can formulate NMF as (6.1)

\[
X = AS
\]  

If a document is viewed as a combination of basis vectors, then it can be categorized belonging to the topic represented by its principal vector. Therefore, NMF can be used to organize text collection into partitional structures or clusters directly.
Chapter 6: Text and document clustering using $\ell_1$-constrained NMF

derived from the non-negative factors. Figure 6.1 represents the use of NMF for document clustering.

![Figure 6.1: Visualization of document clustering using NMF](image)

6.4 $\ell_1$-constrained NMF applied to text classification

We simulated several text datasets to observe the efficiency of our algorithm in this field. For example we ran on a medical abstract dataset, an email dataset and the Reuters dataset. In this section, we provided a brief overview of the datasets. Clustering results and performances are then presented and compared with different algorithms.

The Medical Literature Analysis and Retrieval System (MEDLARS) is a digitalized biomedical abstract retrieval system. This dataset was introduced and enriched by the National Library of Medicine (NLM) of United States in 1964. The current database has over 24 million records, and smaller versions of the dataset are available for non-commercial use. We used a collection of 1159 medical keywords, all of which are indexed based on the importance of appearance. The large data set $X$
Table 6.1: Final error of NMF algorithms for MEDLARS text database

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>2.2050</td>
</tr>
<tr>
<td>HALS</td>
<td>1.9338</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>2.0172</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>1.8471</td>
</tr>
</tbody>
</table>

consists 124 medical abstracts with 1159 keywords. An element of a column of $X$ is positive when the key word is present in that particular abstract, zero otherwise. As a result, we have a very sparse matrix $X$. This was further unitized to make every column of $X$ sum to one to make the dataset suitable for our algorithm.

We then factorize $X$ into two different non-negative matrices, $A$ and $S$, using several NMF algorithms, including MU, HALS, PG and MinMaxNMF. The columns of $A$ categorize the words which are similar, and the columns of $S$ represent the co-efficients of appearances of these keywords in a document.

Figure 6.2 represents the decrease of error function in every iteration for the algorithms. Table 6.1 summarizes the final error for each algorithms, while Table 6.2 shows some classification of the keywords.

Next we ran NMF algorithms for a dataset consisting of e-mails classified into three classes: conference, job, and spam. There were 200 emails for each class totalling 18302. The data matrix $X$ was constructed with 600 columns representing all the emails, with 18302 rows representing the frequency of appearance of the words. This dataset was unitized to be compatible with our algorithm. We used $r=3$ to see the classification and summarize the results in Table 6.3. Table 6.4 shows the
Table 6.2: Top ten words in each category in MEDLARS dataset, clustered by Min-MaxNMF when $r = 10$. Here $a_i$ ($i = 1, \ldots, 10$) represents each column of $A$. 

<table>
<thead>
<tr>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$a_3$</th>
<th>$a_4$</th>
<th>$a_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>'oxygen'</td>
<td>'cells'</td>
<td>'blood'</td>
<td>'maternal'</td>
<td>'acids'</td>
</tr>
<tr>
<td>'tension'</td>
<td>'normal'</td>
<td>'response'</td>
<td>'glucose'</td>
<td>'fatty'</td>
</tr>
<tr>
<td>'cerebral'</td>
<td>'cell'</td>
<td>'rats'</td>
<td>'level'</td>
<td>'acid'</td>
</tr>
<tr>
<td>'blood'</td>
<td>'group'</td>
<td>'brain'</td>
<td>'fetal'</td>
<td>'newborn'</td>
</tr>
<tr>
<td>'pressure'</td>
<td>'culture'</td>
<td>'hypoxia'</td>
<td>'levels'</td>
<td>'fat'</td>
</tr>
<tr>
<td>'arterial'</td>
<td>'strain'</td>
<td>'ventilation'</td>
<td>'ffa'</td>
<td>'pregnancy'</td>
</tr>
<tr>
<td>'cisternal'</td>
<td>'tissue'</td>
<td>'increased'</td>
<td>'plasma'</td>
<td>'free'</td>
</tr>
<tr>
<td>'tissue'</td>
<td>'study'</td>
<td>'rise'</td>
<td>'delivery'</td>
<td>'women'</td>
</tr>
<tr>
<td>'fluid'</td>
<td>'lung'</td>
<td>'results'</td>
<td>'correlation'</td>
<td>'glucose'</td>
</tr>
<tr>
<td>'tissues'</td>
<td>'epithelial'</td>
<td>'ph'</td>
<td>'free'</td>
<td>'pregnant'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$a_6$</th>
<th>$a_7$</th>
<th>$a_8$</th>
<th>$a_9$</th>
<th>$a_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>'human'</td>
<td>'crystallin'</td>
<td>'lens'</td>
<td>'alveolar'</td>
<td>'cancer'</td>
</tr>
<tr>
<td>'cell'</td>
<td>'protein'</td>
<td>'species'</td>
<td>'electron'</td>
<td>'lung'</td>
</tr>
<tr>
<td>'tissue'</td>
<td>'lens'</td>
<td>'rabbit'</td>
<td>'disease'</td>
<td>'major'</td>
</tr>
<tr>
<td>'cultures'</td>
<td>'proteins'</td>
<td>'activity'</td>
<td>'pneumonia'</td>
<td>'lungs'</td>
</tr>
<tr>
<td>'virus'</td>
<td>'fractions'</td>
<td>'bovine'</td>
<td>'pulmonary'</td>
<td>'previous'</td>
</tr>
<tr>
<td>'tissues'</td>
<td>'fraction'</td>
<td>'epithelium'</td>
<td>'epithelial'</td>
<td>'growth'</td>
</tr>
<tr>
<td>'culture'</td>
<td>'soluble'</td>
<td>'lenses'</td>
<td>'bodies'</td>
<td>'patients'</td>
</tr>
<tr>
<td>'lung'</td>
<td>'insoluble'</td>
<td>'specific'</td>
<td>'lungs'</td>
<td>'found'</td>
</tr>
<tr>
<td>'growth'</td>
<td>'molecular'</td>
<td>'dehydro'</td>
<td>'interstitial'</td>
<td>'contrast'</td>
</tr>
<tr>
<td>'tumor'</td>
<td>'acid'</td>
<td>'showed'</td>
<td>'lining'</td>
<td>'presented'</td>
</tr>
</tbody>
</table>
Figure 6.2: Decrease of error in every iteration for NMF algorithms while clustering MEDLARS data.

6.4.1 Document clustering

Reuters-21578 is currently the most widely used benchmark document collection of business newswire posts. The original database contains over 20,000 documents grouped under 118 different topics. This database is available online\(^1\). The subset reuters10 is a collection of 9248 documents with 10 categories from the "ModApte

\(^1\)www.daviddlewis.com
Chapter 6: Text and document clustering using $\ell_1$-constrained NMF

Table 6.3: Top ten words in each category in email database clustered by Min-MaxNMF when $r = 3$. Here $\mathbf{a}_i$ ($i = 1, \ldots, 3$) represents each column of $\mathbf{A}$.

<table>
<thead>
<tr>
<th>$\mathbf{a}_1$</th>
<th>$\mathbf{a}_2$</th>
<th>$\mathbf{a}_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>'systems'</td>
<td>'people'</td>
<td>'university'</td>
</tr>
<tr>
<td>'neural'</td>
<td>'report'</td>
<td>'poster'</td>
</tr>
<tr>
<td>'research'</td>
<td>'money'</td>
<td>'institute'</td>
</tr>
<tr>
<td>'usa'</td>
<td>'internet'</td>
<td>'model'</td>
</tr>
<tr>
<td>'university'</td>
<td>'order'</td>
<td>'neural'</td>
</tr>
<tr>
<td>'cognitive'</td>
<td>'information'</td>
<td>'learning'</td>
</tr>
<tr>
<td>'papers'</td>
<td>'business'</td>
<td>'technology'</td>
</tr>
<tr>
<td>'computer'</td>
<td>'free'</td>
<td>'research'</td>
</tr>
<tr>
<td>'conference'</td>
<td>'program'</td>
<td>'models'</td>
</tr>
<tr>
<td>'information'</td>
<td>'time'</td>
<td>'california'</td>
</tr>
</tbody>
</table>

Table 6.4: Final error of NMF algorithms for email text database

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>1.1057</td>
</tr>
<tr>
<td>HALS</td>
<td>0.9080</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>0.9567</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>0.7481</td>
</tr>
</tbody>
</table>

Split”. It has greatest number of positive training examples.

Document clustering is a similar process to text classification. Each document is assigned a positive value when it is similar to a particular topic. Columns of $\mathbf{X}$ contain the frequencies for each documents. We factorized the $\mathbf{X}$ into $\mathbf{A}$ and $\mathbf{S}$ with $r = 10$ and matched the result with pre-classified data. The accuracy of factorization is summarized in Table 6.5.
Chapter 6: Text and document clustering using $\ell_1$-constrained NMF

Table 6.5: Final error of NMF algorithms for reuters10 database

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplicative Update (MU)</td>
<td>$4.83 \times 10^{-2}$</td>
</tr>
<tr>
<td>HALS</td>
<td>$8.3408 \times 10^{-2}$</td>
</tr>
<tr>
<td>Projected Gradient (PG)</td>
<td>$1.03 \times 10^{-2}$</td>
</tr>
<tr>
<td>MinMaxNMF</td>
<td>$3.842 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

Figure 6.3: variation of error with respect to number of categories for reuters10 database in MinMaxNMF algorithm.

6.5 Conclusion

Text and document classification has become one of the most popular fields of research in the past decades. With the advancement of computational devices and
algorithms, this process has become much easier than the manual classification. A number of classification algorithms have been developed in recent times to increase the speed and accuracy of classification. Artificial neural networks and machine learning algorithms are utilized mostly for classification and clustering where thousands of data are used for training purposes. When new documents or texts are provided as input, the algorithms match the previously learned pattern with the new input and categorize the document or text. This supervised process is much more complex and time consuming as the usual datasets are very large.

On the other hand, NMF is a very recent idea, where a non-negative data set can be factorized into a category matrix (basis matrix) and a co-efficient matrix. As NMF provides parts based learning, it categorizes the input dataset without any prior training or knowledge. The process is also known as unsupervised learning. In this chapter we presented the application of the MinMaxNMF algorithm proposed in Chapter four. We provided the experimental results for several datasets and compared the performances with other algorithms.

It is difficult to find a suitable match for NMF algorithms with other algorithms. For this reason we mainly compared our algorithm with other NMF algorithms. The performance results and clustering results indicate that the MinMaxNMF algorithm outperforms other algorithms. While the final error is less, it does not always guarantee the best categorization, and there is also no perfect categorization. Therefore, the categorization or classification accuracy is not the ideal approach for comparing performance. However, we also compared the accuracy with respect to the original classifications provided by the database owners.

Since the data matrix of a text database is usually sparse, our algorithm per-
forms better in this scenario, as our algorithm generates a sparse co-efficient matrix while generating a dense category matrix. In conclusion, NMF algorithms provide a new method of text and document classification, and the proposed MinMaxNMF algorithm outperforms many of the previous algorithms. With further improvements, this algorithm can also outperform many supervised classification methods.
Chapter 7

Conclusions and Recommendations

This chapter summarizes the main contributions of the thesis and provides suggestions for pursuing future research on this topic.

7.1 Summary of main contributions

The focus of this thesis is the development of a novel method for non-negative matrix factorization. Matrix factorization is a popular subject of linear algebra with a wide spectrum of applications. Non-negative matrix factorization (NMF) is a very recent concept, but it has attracted attention due to its application in several practical fields including image processing, text and document classification, dimensionality reduction, music analysis, computational biology and many more.

This research aims to improve the factorizing techniques. To develop a novel algorithm, previously developed matrix factorization and NMF methods were studied. Singular value decomposition (SVD), principal component analysis (PCA) and independent component analysis (ICA) are some of the popular factorization techniques, but they do not utilize the non-negativity constraints over data. On the
other hand, the multiplicative update rule, non-negative alternating least squares, projected gradient algorithm for non-negative factorization are popular NMF algorithms. These algorithms solve similar cost functions, and the basic difference is the process of optimization. This thesis concentrates on developing a new objective function that will produce improved factorization compared to other methods. The geometric concepts and theories behind the NMF were studied to formulate a new cost function. Based on the geometric properties of convex optimization, the convex cone containing the dataset needs to be minimized to obtain a good solution. While minimizing the convex cone, the other factor, the co-efficient matrix, became sparser. We utilized this property to produce the cost function to be optimized. In a unitized $\ell_1$-space, the Frobenius norm of the basis matrix has been minimized to obtain the minimum convex cone that contains all the data vectors. The Frobenius norm of the co-efficient matrix is forced to be maximum by optimization at the same time. Although the problem itself is not convex, the alternating optimization technique was adopted, where the basis matrix is solved while the co-efficient matrix is fixed and vice versa. This makes sub-problems convex.

The advantages of the proposed algorithm are:

- In the proposed objective function, the algorithm minimizes the Frobenius norm of basis matrix. This process provides more compact convex cone or simplicial cone that encloses all the data points within it. To find the optimum simplicial cone, this method reduces the Euclidean distance of each column vector of the basis matrix, respect to a unitized $\ell_1$ space, where the $\ell_1$ norm of each column equals to one. By doing this, the extreme rays of the cone come closer to the data points. It is further reduced to reach a stationary point, until the cone cannot be smaller subject to the restriction that it contains all the data points.
Chapter 7: Conclusions and Recommendations

- The proposed algorithm maximizes the Frobenius norm of co-efficient matrix simultaneously with the previous process, which provides the sparse co-efficient matrix. When the Frobenius norm is maximized on a unitized $\ell_1$ space, the column vectors of the co-efficient matrix spreads from the center points.

- By optimizing the basis matrix and co-efficient matrix simultaneously as described above, it improves the convergence. It reduces the iterations to reach a stationary point or local solution. Hence this algorithm provides faster solution while the minimum error is also reduced.

- This method is very effective, especially where the co-efficient matrix is naturally sparse. For example, in facial image processing, text or document classification the co-efficient matrix is usually sparse. Therefore, proposed Min-MaxNMF algorithm performs better for these applications.

Of several popular applications, the proposed algorithm has been applied to parts-based image learning and text or document clustering. These two applications have received a good deal of attention during the last decades because of the popularity of computer vision, global information and data boosting. Parts-based image learning provides a set of features extracted from the image data. NMF has the capability of feature extraction by its nature. Our algorithm has been applied on MIT CBCL and AT&T face databases for facial recognition. These have been the popular databases for many years and previously developed algorithms have also utilized these databases for experiments. By applying the proposed method to these databases, the performances are compared with other algorithms. Chapter five summarized the results and outcomes of these experiments along with explanations.

Later in this thesis, the proposed constrained NMF algorithm was applied to text and document classification. Datasets from popular databases including Medical
Chapter 7: Conclusions and Recommendations

Literature Analysis and Retrieval System (MEDLARS), Email and Reuters-21578 are used for text and document clustering in Chapter six. Results from the simulations show that the proposed algorithm outperforms the popular NMF algorithms. Classification errors are listed and explained to prove the claims.

This thesis approached to solve the NMF problem by introducing a new constraint and formulating a new objective function for optimization unlike other algorithms, where the focus were only to search for better methods of optimization. With empirical results, the performances of popular algorithms were compared with the proposed algorithm and the proposed algorithm outperforms them. As a new technique, the work of this thesis has the scope for further extensions in future. Some of the recommendations are indicated in the next section.
7.2 Recommendations for future work

Some recommendations for future work are listed below:

- **Improved optimization technique for $\ell_1$-constrained NMF.**
  
  In this thesis, a new objective function is proposed where the data matrix and its factors are constrained to have their $\ell_1$-norm equal to one. This objective function is optimized using traditional optimization software to find the local solutions which proves the effectiveness of the improvement in the objective function. Different optimization techniques can be applied to solve the proposed objective function which may provide better and faster convergence. Future works may concentrate on improving the optimization technique to reduce iterations and time consumption.

- **Searching for global solutions.**
  
  So far, NMF problems are usually solved to find local solutions. In this thesis, the traditional approach to find local solutions was followed. Some explanations for this approach were provided throughout the thesis. However, many studies [19, ?, 112] indicated that, there are possibilities of finding global solutions by introducing some additional constraints, though it is not proven yet. Further studies are encouraged to search the possibility to find uniqueness conditions for an NMF problem.

- **Application to blind source separation.**
  
  In addition to the theoretical improvements, the proposed algorithm has the possibility of being applied to non-negative blind source separation (NBSS). Many previous studies show impressive results for NBSS problems using NMF algorithm. However, main obstacle to this is the unavailability of a global or unique solution.
• Combination with machine learning and artificial neural network.

Previously few NMF techniques were combined with machine learning (ML) and artificial neural network (ANN) algorithms for pre-processing purpose. The proposed method in this thesis can also be combined with ML and ANN methods to increase the pre-processing performance.
References


References


References


References


References


References


[70] D. Wang, T. Li, and C. Ding, “Weighted feature subset non-negative matrix factorization and its applications to document understanding,” in IEEE 10th
References


References


References


[96] K. J. Karande, “Multiscale wavelet based edge detection and independent component analysis (ica) for face recognition,” in *2012 International Con-
References


References


The End