

HYPERSPECTRAL UNMIXING AND BAND WEIGHTING FOR MULTIPLE ENDMEMBER SETS

A Thesis presented to
the Faculty of the Graduate School
at the University of Missouri

In Partial Fulfillment
of the Requirements for the Degree
Master of Science

by
PIYUSH PRADEEP KHOPKAR
Dr. Alina Zare, Thesis Supervisor
May 2014

© Copyright by Piyush Pradeep Khopkar 2014
All Rights Reserved

The undersigned, appointed by the Dean of the Graduate School, have examined
the thesis entitled:

**HYPERSPECTRAL UNMIXING AND BAND WEIGHTING
FOR MULTIPLE ENDMEMBER SETS**

presented by Piyush Pradeep Khopkar,
a candidate for the degree of Master of Science and hereby certify that, in their
opinion, it is worthy of acceptance.

Dr. Alina Zare

Dr. James Keller

Dr. Mihail Popescu

To Aai-Baba and Ayush

ACKNOWLEDGMENTS

I would like to take this opportunity to express my gratitude and sincere thanks for everybody who made my graduation at University of Missouri possible. I owe deepest gratitude to my parents Anjali and Pradeep Khopkar, brother Ayush Khopkar and my family. Without their motivation, faith, and continuous support this would not have been possible.

I would like to thank my adviser Dr. Alina Zare for her valuable guidance, constant support, and encouragement throughout my research. I would like to sincerely thank members of my thesis committee; Dr. James Keller and Dr. Mihail Popescu for providing valuable suggestions to further improve my research work.

I am very grateful to my lab mates, especially, Josephine Dula, Xiaoxiao Du, and Changzhe Jiao for the countless number of times they provided insight and help during my studies. I would like to thank Nakini Tushar Kanta Das, for his help, and valuable suggestions throughout my studies and research. I am also thankful to Late Dr. Aditi Avhad, Akshay Wankhede, Anup Mishra, Gautam Gurnani, Harish Bezawada, Kaustubh Gadre, Nikhil Dhadake, Naren Kumar, Piyush Sharma, Shraddha Sankhe, and Tejas Sevekari for the good times spent in the graduate school. My sincere appreciation to ASNE-School of Journalism and International Center for providing me the financial assistance through the most part of my graduate studies.

Last but not the least, I am thankful to Neha Modak and Gitam Shikkenawis for their encouragement and support during ups and downs throughout my studies.

TABLE OF CONTENTS

| | |
|---|-------------|
| LIST OF TABLES | vii |
| LIST OF FIGURES | viii |
| ABSTRACT | x |
| 1 Introduction | 1 |
| 1.1 Hyperspectral Image Analysis | 1 |
| 1.2 Hyperspectral Unmixing | 2 |
| 1.2.1 Linear Mixing Model | 4 |
| 1.2.2 Non-Linear Mixing Model | 5 |
| 1.3 Hyperspectral Band Weighting and Selection | 6 |
| 1.4 Piece-Wise Convex Unmixing | 6 |
| 1.5 Problem Statement | 7 |
| 2 Literature Review | 9 |
| 2.1 Review: Hyperspectral Unmixing and Endmember Estimation Methods | 9 |
| 2.1.1 Pure Pixel-Based Endmember Extraction Methods | 10 |
| 2.1.2 Minimum Volume-Based Spectral Unmixing Methods | 13 |
| 2.1.3 Sparse Endmember Estimation Methods | 15 |
| 2.2 Review: Hyperspectral Dimensionality Reduction Methods | 18 |
| 2.2.1 Transformation-Based Methods | 19 |
| 2.2.2 Band Selection Methods | 27 |

| | | |
|----------|---|------------|
| 2.3 | Review: Piece-Wise Convex Unmixing | 38 |
| 2.3.1 | Dirchlet Process-Based Methods | 38 |
| 2.3.2 | Spectral and Spatial Information-Based Piece-wise Convex Methods | 39 |
| 2.3.3 | Other Methods | 40 |
| 2.3.4 | PCOMMEND | 41 |
| 3 | Methodology | 44 |
| 3.1 | Review: Simultaneous Clustering and Attribute Discrimination (SCAD) | 45 |
| 3.2 | Weight Decay Approaches | 47 |
| 3.2.1 | SUBSUME-SCAD-WD | 51 |
| 3.2.2 | SUBSUME-BST-WD | 54 |
| 3.3 | Sparsity Promoting Approaches | 58 |
| 3.3.1 | SUBSUME-SCAD-SP | 61 |
| 3.3.2 | SUBSUME-BST-SP | 65 |
| 4 | Experimental Results and Analysis | 70 |
| 4.1 | Simulated Hyperspectral Data | 71 |
| 4.1.1 | ASTER Data Set | 71 |
| 4.1.2 | Results and Analysis- Simulated Hyperspectral Data | 75 |
| 4.2 | Real Hyperspectral Data | 85 |
| 4.2.1 | AVIRIS Indian Pines Data | 87 |
| 4.2.2 | ROSIS Pavia University Data | 98 |
| 4.2.3 | MUUFL Gulf Port Data | 114 |
| 5 | Conclusion and Future Work | 121 |

| | | |
|-----|-----------------------|-----|
| 5.1 | Conclusion | 121 |
| 5.2 | Future Work | 122 |

APPENDIX

| | | |
|----------|---|------------|
| A | Derivations: Weight Decay Approaches | 124 |
| A.1 | SUBSUME-SCAD-WD | 124 |
| A.1.1 | Membership | 125 |
| A.1.2 | Weights | 127 |
| A.1.3 | Endmembers | 129 |
| A.1.4 | Proportions | 131 |
| A.2 | SUBSUME-BST-WD | 135 |
| A.2.1 | Membership | 136 |
| A.2.2 | Weights | 136 |
| A.2.3 | Endmembers | 138 |
| A.2.4 | Proportions | 139 |
| B | Derivations: Sparsity Promoting Approaches | 140 |
| B.1 | SUBSUME-SCAD-SP | 140 |
| B.1.1 | Membership | 141 |
| B.1.2 | Weights | 142 |
| B.1.3 | Endmembers | 145 |
| B.1.4 | Proportions | 145 |
| B.2 | SUBSUME-BST-SP | 147 |
| B.2.1 | Membership | 147 |

| | | |
|-------------------------------|-----------------------|------------|
| B.2.2 | Weights | 148 |
| B.2.3 | Endmembers | 152 |
| B.2.4 | Proportions | 153 |
| BIBLIOGRAPHY | | 158 |
| VITA | | 171 |

LIST OF TABLES

| Table | Page |
|---|------|
| 4.1 Input Parameters | 70 |
| 4.2 Experiment 1: Parameter Settings | 76 |
| 4.3 Experiment 1: Mean EMD with SED SAM | 76 |
| 4.4 Experiment 1: Mean EMD with SED/D SAM/D | 77 |
| 4.5 Random Initialization E & M: Mean EMD with SED & SAM | 83 |
| 4.6 Random Initialization E & M: Mean EMD with SED/D & SAM/D | 83 |
| 4.7 Random Initialization M & P: Mean EMD with SED & SAM | 84 |
| 4.8 Random Initialization M & P: Mean EMD with SED/D & SAM/D | 84 |
| 4.9 Random Initialization E & P: Mean EMD with SED & SAM | 85 |
| 4.10 Random Initialization E & P: Mean EMD with SED/D & SAM/D | 85 |

LIST OF FIGURES

| Figure | Page |
|---|------|
| 1.1 HSI Concept | 3 |
| 1.2 Linear Mixing Model | 4 |
| 1.3 AVIRIS Indian Pines Data after PCA | 7 |
| 3.1 Laplacian Distribution | 60 |
| 4.1 True Endmembers: Simulated Data | 72 |
| 4.2 Alpha: SUBSUME-BST-WD | 79 |
| 4.3 Alpha: SUBSUME-SCAD-WD | 80 |
| 4.4 Alpha: SUBSUME-SCAD-SP | 81 |
| 4.5 Alpha: SUBSUME-BST-SP | 82 |
| 4.6 Indian Pines Data | 88 |
| 4.7 Indian Pines: PCOMMEND Results ($C=2, M=3$) | 90 |
| 4.8 Indian Pines: PCOMMEND Results ($C=2, M=5$) | 90 |
| 4.9 Indian Pines: SUBSUME-SCAD-WD Results ($C=2, M=3$) | 92 |
| 4.10 Indian Pines: SUBSUME-SCAD-WD Results ($C=2, M=5$) | 92 |
| 4.11 Indian Pines: SUBSUME-SCAD-SP Results ($C=2, M=3$) | 94 |
| 4.12 Indian Pines: SUBSUME-SCAD-SP Results ($C=2, M=5$) | 94 |

| | |
|--|-----|
| 4.13 Indian Pines: SUBSUME-BST-WD Results ($C=2, M=3$) | 95 |
| 4.14 Indian Pines: SUBSUME-BST-WD Results ($C=2, M=5$) | 95 |
| 4.15 Indian Pines: SUBSUME-BST-SP Results ($C=2, M=3$) | 97 |
| 4.16 Indian Pines: SUBSUME-BST-SP Results ($C=2, M=5$) | 97 |
| 4.17 Pavia University Data Subset- 1 | 99 |
| 4.18 Pavia-1: PCOMMEND Results | 100 |
| 4.19 Pavia-1: SUBSUME-SCAD-WD Results | 101 |
| 4.20 Pavia-1: SUBSUME-SCAD-SP Results | 103 |
| 4.21 Pavia-1: SUBSUME-BST-WD Results | 104 |
| 4.22 Pavia-1: SUBSUME-BST-SP Results | 106 |
| 4.23 Pavia University Data Subset- 2 | 107 |
| 4.24 Pavia-2: PCOMMEND Results | 108 |
| 4.25 Pavia-2: SUBSUME-SCAD-WD Results | 109 |
| 4.26 Pavia-2: SUBSUME-SCAD-SP Results | 110 |
| 4.27 Pavia-2: SUBSUME-BST-WD Results | 112 |
| 4.28 Pavia-2: SUBSUME-BST-SP Results | 113 |
| 4.29 Gulf Port Data | 114 |
| 4.30 Gulf Port: PCOMMEND Results | 115 |
| 4.31 Gulf Port: SUBSUME-SCAD-WD Results | 116 |
| 4.32 Gulf Port: SUBSUME-SCAD-SP Results | 117 |
| 4.33 Gulf Port: SUBSUME-BST-WD Results | 119 |
| 4.34 Gulf Port: SUBSUME-BST-SP Results | 120 |

ABSTRACT

Imaging spectrometers measure the response from materials across the electromagnetic spectrum. Often, in remote sensing applications, the imaging spectrometers have low spectral resolution resulting in most measurements being mixed spectra from a scene. In these cases, pixels are assumed to be mixtures of pure spectra known as endmembers. Given the prevalence of mixed spectra, a common analysis technique is spectral unmixing. When the responses from materials are measured across hundreds or thousands of narrow contiguous wavelengths, then it gives rise to the concept of *Hyperspectral Imaging (HSI)*.

The high dimensionality of the hyperspectral data results in highly correlated information in neighboring spectral bands. Also, irrelevant bands may degrade the performance of spectral unmixing methods. As a result, band weighting can aid hyperspectral unmixing and endmember estimation methods. ‘*Band Weighting*’ refers to a method for assigning weights to each band in hyperspectral data that indicate the relative degree of importance during unmixing.

Many hyperspectral scenes possess multiple sets of endmembers. For example, consider a scene consisting of two distinct ground regions (eg. urban and rural areas) composed of distinct materials or endmembers. The image spectra from each region will then only be mixtures of the endmembers associated with the corresponding region.

The proposed methods perform endmember estimation for multiple sets of endmembers, estimates proportion values, and assigns partition specific band weights. Band weights are estimated using weight decay and sparsity promoting techniques.

The proposed approach is compared with state of the art method present in the literature. Results are shown on simulated and real hyperspectral data sets.

Chapter 1

Introduction

1.1 Hyperspectral Image Analysis

Spectral imaging is one of the domains of spectroscopy and photography. It refers to collecting and processing spectral information across a range of the electromagnetic spectrum. For a layperson, spectral imaging can be related to conventional photography by the fact that in conventional photography an image is captured on three wavelengths: Red, Green, and Blue (i.e. every pixel in an image has its reflectance value measured on these three wavelengths); whereas, in spectral imaging every pixel has its reflectance value measured on hundreds or thousands of narrow wavelengths bands. This makes spectral imaging helpful in the extraction of additional information from the scene which can not be done with the human eye. *Multispectral* and *hyperspectral* are two categories in spectral imaging. Due to coarser spectral resolution or a larger width between neighboring spectral bands, multispectral images

have fewer number of bands when compared to hyperspectral images. On the other hand, hyperspectral imagery has a higher spectral resolution. Hyperspectral images are captured by imaging spectrometers such as NASA JPL's AVIRIS (Airborne Visible InfraRed Imaging Spectrometer) [1], HYDICE (Hyperspectral Digital Imagery Collection Experiment) [2], and others.

In Hyperspectral Imaging (HSI), hyperspectral sensors collect radiance or reflectance values from a scene across different wavelengths of the electromagnetic spectrum. These radiance values are organized into planes forming a three dimensional data cube [3]. Each element of this cube refers to the radiance or reflectance value measured at one particular wavelength at one pixel location. Each plane corresponds to radiance values measured at particular spectral band for all pixels in the scene. Fig. 1.1 illustrates the concept of Hyperspectral Imaging. This data cube was acquired by the OMEGA instrument on the Mars Express Orbiter observing the South Polar Cap of Mars in 2004 [4; 5].

Due to higher spectral resolution, hyperspectral imagery may allow for target identification. This possibility broadens the application of HSI to variety of fields such as food safety and quality [6; 7], biometric [8], archeology [9; 10], space exploration [11; 12], preservation of historical documents [13], and many more.

1.2 Hyperspectral Unmixing

Hyperspectral image analysis consists of *endmember* estimation and *proportion* estimation. Endmember estimation aims at estimating the pure spectral signature of a material. It is assumed that a pixel in a hyperspectral image is a combination of

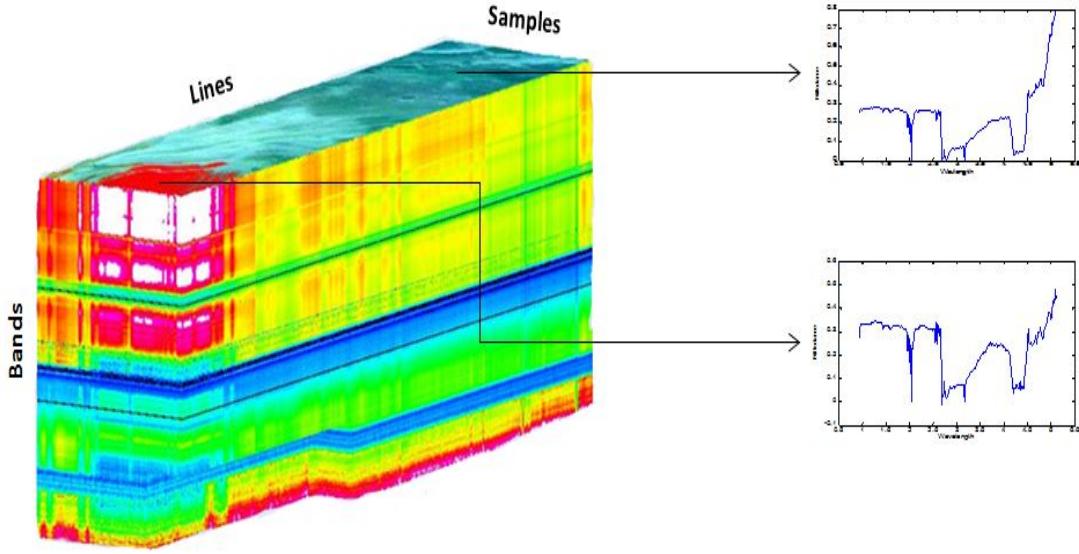


Figure 1.1: Hyperspectral Data cube and illustration of Hyperspectral Imaging. Image cube visualized by PCI Geomatica 2012.

the spectral signatures of the different material present in the pixel's field of view. Estimation of the percentage of a material that present in a pixel is referred to as proportion estimation [3]. The above process of extracting endmembers and their corresponding proportion is called Hyperspectral Unmixing.

The method of unmixing depends on the mixing model. The mixing model specifies the way in which endmembers and proportions are combined in a pixel. Mixing models can be categorized into two categories: *Linear* and *Non-Linear* mixing models [14; 3].

1.2.1 Linear Mixing Model

According to Singer, et al. [15], the Linear Mixing Model (*LMM*) is applicable when the mixing scale is macroscopic and incident light undergoes interaction with only one material in the scene. This phenomenon can be explained with a ‘check board’ mixture situation as shown in Fig. 1.2, where m_1 , m_2 , and m_3 are the three different materials present in a pixel in α_1 , α_2 , and α_3 proportions respectively.

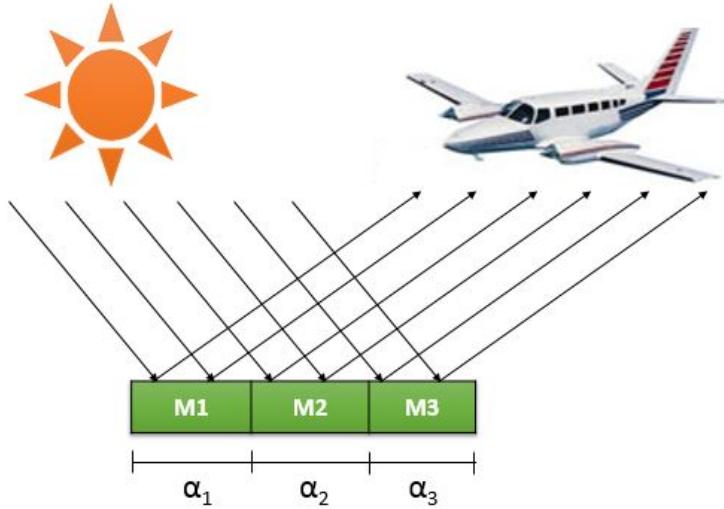


Figure 1.2: Linear Mixing Model

In the LMM, light from the different materials is mixed within the detector [3]. Assuming that there is no scattering of light between the distinct materials [3], then the reflectance spectra measured by a sensor is the linear combination of the spectral signature of each endmember weighted by its corresponding proportion value [14]. Mathematically, the LMM model can be written as shown in (1.1),

$$\mathbf{x}_i = \sum_{k=1}^M p_{ik} \mathbf{e}_k + \boldsymbol{\epsilon}_i \quad i = 1, \dots, N \quad (1.1)$$

where N is the number of pixels in the image, M is the number of endmembers, ϵ_i is an error term, p_{ik} is the proportion of endmember k in pixel i , and \mathbf{e}_k is the k^{th} endmember. The proportions of this model satisfy the constraints in (1.2),

$$\text{Non-negativity: } p_{ik} \geq 0 \quad \forall k = 1, \dots, M$$

$$\text{Sum-to-one: } \sum_{k=1}^M p_{ik} = 1 \quad (1.2)$$

The LMM model given by (1.1) is also known as the *Convex Geometry Model*. This model assumes that every pixel is a convex combination of endmembers present in the scene and that the endmembers lie at the vertices of a simplex [14; 3]. A k -simplex is defined as a k -dimensional polytope which is the convex hull of its $(k+1)$ vertices. With this, a 2-simplex would be a triangle, a 3-simplex is a tetrahedron etc.

Under linear mixing model there are generally four different types of hyperspectral unmixing approaches: geometrical, statistical, sparse regression, and spatial-spectral contextual information based approaches [3]. These methods are reviewed in Chapter 2 in the Literature Review.

1.2.2 Non-Linear Mixing Model

Uneven scattering of light between multiple materials can give rise to Non-Linear Mixing [3]. The non-linear interactions can be between multiple layers of materials or at an intimate level. In multilayer situations, the radiated light after being reflected from one material undergoes additional reflection from other materials before it hits the detector. On the other hand, in intimate interactions, the surface materials are

intimately mixed which cause the interaction of light at a molecular level.

Since the proposed method is based on linear mixing model, algorithms based on the linear mixing model are discussed in this work.

1.3 Hyperspectral Band Weighting and Selection

Hyperspectral imaging involves acquiring reflectance or radiance values at many different wavelengths called bands [16]. Often the information present in neighboring bands is highly correlated. Although higher spectral resolution allows for more precise identification of materials, hyperspectral imagery often deals with the *curse of dimensionality* [17; 16]. It is often beneficial to reduce the data dimensionality before analysis. This problem can be solved by data reduction techniques [18].

1.4 Piece-Wise Convex Unmixing

The vast majority of spectral unmixing methods search for a single set of endmembers to describe a hyperspectral scene and, as such, they assume the input hyperspectral data lie in a single convex region. Therefore, these approaches cannot be appropriately applied to non-convex hyperspectral data. Many hyperspectral scenes are non-convex. For example, consider a scene consisting of two distinct ground regions (eg., urban and rural areas) composed of distinct materials or endmembers. Fig. 1.3 shows the AVIRIS Indian Pines Hyperspectral Data set [19] after applying Principal Component Analysis (PCA) [20] for dimensionality reduction to three dimensions. It can be observed that the Indian Pines Data is non-convex. Thus, an approach that can

perform piece-wise convex unmixing is desirable.

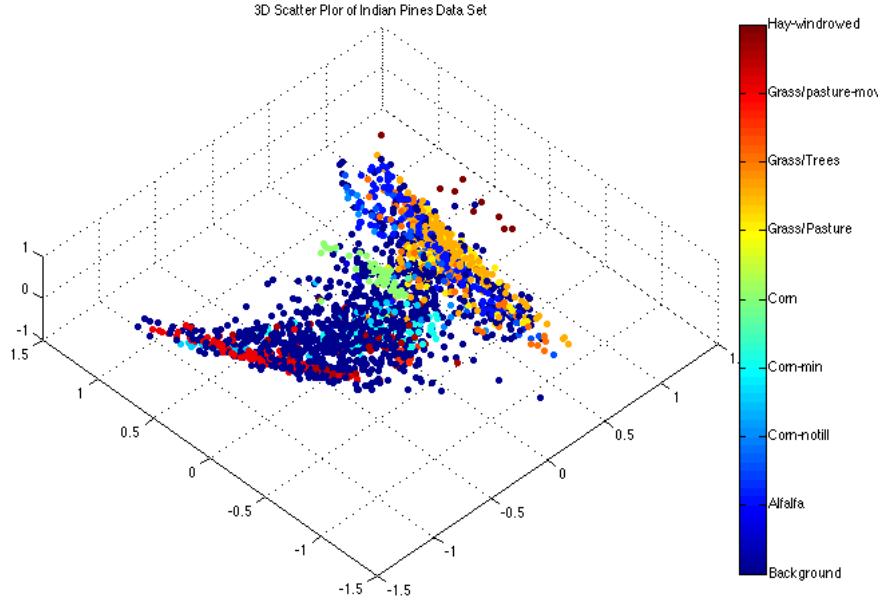


Figure 1.3: The AVIRIS Indian Pines Hyperspectral Data after applying PCA for dimensionality reduction to 3 dimensions.

In piece-wise convex unmixing the image spectra from each region will only be mixtures of the endmembers associated with the corresponding convex region.

1.5 Problem Statement

Most of spectral unmixing algorithms assume that the hyperspectral data lie in a single set convex region formed by a single set of endmembers. However, some hyperspectral data scenes are better represented by overlapping convex regions (i.e. by multiple sets of endmembers).

Furthermore, analysis of hyperspectral data cubes are plagued by the *Curse of*

dimensionality. This is often dealt with in two ways is solved in two ways: projecting the hyperspectral data on some lower dimensional subspace which often results in the loss of physical meaning of each band of the data or Reducing the number of bands through band selection.

This work develops and investigates methods in hyperspectral unmixing for non-convex data sets and simultaneous dimensionality reduction of hyperspectral data.

Chapter 2

Literature Review

This chapter provides a review of existing hyperspectral unmixing and endmember estimation methods, hyperspectral dimensionality reduction methods, and piece-wise convex unmixing methods.

2.1 Review: Hyperspectral Unmixing and End-member Estimation Methods

As mentioned in section 1.2, hyperspectral image analysis often consists of endmember estimation and proportion estimation. Estimating the pure spectral signature of each material in a hyperspectral image is referred to as endmember estimation. Estimating the proportion values for each pixel is referred to as proportion estimation.

This section reviews hyperspectral unmixing and endmember estimation methods. This section is subdivided into three parts: Pure pixel-based endmember extraction methods, minimum volume-based spectral unmixing methods, and sparse endmember

estimation methods.

2.1.1 Pure Pixel-Based Endmember Extraction Methods

Pure pixel-based endmember extraction algorithms are based on the assumption that there exists at least one pure pixel for each endmember in the scene. These algorithms rely on the fact that endmembers from an input hyperspectral scene can be found without performing spectral unmixing [21]. Some of representative algorithms following the pure pixel assumption are Pixel Purity Index (PPI) [22], N-FINDR [23], Iterative Error Analysis (IEA) [24], and Vertex Component Analysis (VCA) [25].

2.1.1.1 Pixel Purity Index (PPI)

The Pixel Purity Index (PPI) [22] is a hyperspectral endmember extraction algorithm for determining pure pixels in input hyperspectral data. This algorithm projects every spectral vectors onto *skewers*. These *skewers* are randomly generated vectors. The algorithm is initialized by assigning zero as the initial value of purity index for each pixel. Following each projection onto the randomly generated *skewers*, points that are projected onto the extreme ends of the skewers are identified and their corresponding purity indices are incremented. After all projections, pixels with the highest purity indices are identified as the purest ones and, thus, the endmembers. This algorithm does not estimate the number of endmembers for an input image.

There are major limitations with PPI. Since the *skewers* are generated randomly, there is no guarantee that the pixels identified using PPI are actually the true end-members [26]. Moreover, PPI uses the Maximum Noise Fraction (MNF) [27] for reducing the dimensionality of the data as a pre-processing step, but it does not have

a method to determine the number of dimensions to retain. In addition PPI is computationally costly [28]. To overcome these problems, the Fast Iterative PPI (FIPPI) has been proposed. FIPPI is an iterative method and results in final set of sample vectors as its desired set of endmembers after the convergence of the algorithm. FIPPI employs the concepts introduced in Automatic Target Generation Process (ATGP) [29] to generate an initial set of endmembers. FIPPI uses a similar indexing scheme as in the original PPI, for the estimation of PPI for each pixel.

2.1.1.2 Vertex Component Analysis (VCA)

Vertex Component Analysis (VCA) [25] also assumes the presence of pure pixels for each endmember in input hyperspectral data. This method takes an advantage of two assumptions: 1) that endmembers are the vertices of a simplex; and 2) that the affine transformation of a simplex is also a simplex. Following the assumption of linear mixing model, that an input hyperspectral data lie inside a convex cone, the VCA algorithm models the input hyperspectral data using a convex cone and projects this convex cone onto a properly selected hyperplane. Following the projection of the data onto a selected hyperplane, VCA projects all the image pixels onto random direction on the plane. The pixel having the largest projection is selected as the first endmember. The remaining endmembers are identified by iteratively projecting the data onto a direction orthogonal to subspace spanned by the previous endmembers. The pixel that corresponds to the extreme of the projection is selected as the new endmember. This algorithms iterates until the desired number of endmembers are extracted.

2.1.1.3 N-FINDR

N-Findr [30] aims at finding the set of pixels from an image that maximizes their enclosed volume. These pixels are eventually termed as potential endmembers. As a pre-processing step at the beginning, the dimensionality of the data is reduced using MNF (Maximum Noise Fraction) [27]. Endmembers are initialized by randomly picking pixels from the data. In each iteration, an endmember is replaced by each of the other pixels in the image in turn. The volume of each set is calculated using (2.1). The current replacement of an endmember in the image is maintained if the calculated volume is found to be greater than the volume calculated previously

$$\mathbf{E} = \begin{bmatrix} 1 & 1 & \cdot & 1 \\ \mathbf{e}_1 & \mathbf{e}_2 & \cdot & \mathbf{e}_M \end{bmatrix}$$

where \mathbf{e}_i is the endmember column vectors, M is the number of endmembers used to calculate the simplex volume. The volume of the endmembers are estimated by computing (2.1).

$$V(\mathbf{E}) = \frac{1}{(M-1)} \text{abs}(|\mathbf{E}|) \quad (2.1)$$

The method continues until there is no replacement of any endmember in an iteration [21]. Apart from assuming the presence of pure pixels in the scene, this method also requires the knowledge of number of endmembers present in the input hyperspectral scene [21].

2.1.2 Minimum Volume-Based Spectral Unmixing Methods

Minimum volume-based algorithms try to estimate an endmember matrix such that the enclosed volume is minimized subject to the constraints given by (2.2)

$$\text{Non-negativity: } p_{ik} \geq 0 \quad \forall k = 1, \dots, M$$

$$\text{Sum-to-one: } \sum_{k=1}^M p_{ik} = 1 \quad (2.2)$$

Minimum volume-based spectral unmixing methods are optimization-based methods that do not assume the presence of pure pixels in the scene. This section reviews some minimum volume-based algorithms.

2.1.2.1 Iterated Constraint Endmembers (ICE)

Iterated Constraint Endmembers (ICE) [31] is an optimization-based method that attempts to minimize the objective function given by (2.5) in order to estimate endmembers and proportions. The ICE algorithm performs a minimization of the Residual Sum of Squares (RSS) term based on the linear mixing model. The RSS term shown in (2.3) is the measure of discrepancy between the input data and the estimated endmembers and proportions.

$$RSS = \sum_{i=1}^N \left(X_i - \sum_{k=1}^M p_{ik} E_k \right)^T \left(X_i - \sum_{k=1}^M p_{ik} E_k \right) \quad (2.3)$$

where N is the number of data points, M is the number of endmembers, p_{ik} is proportion of the k^{th} endmember in i^{th} data point, and E_k is the k^{th} endmember.

In order to constrain the size of the simplex, a Sum of Squared Distance (SSD) term given by (2.4) is added to the ICE objective function.

$$SSD = \sum_{k=1}^{M-1} \sum_{l=k+1}^M (E_k - E_l)^T (E_k - E_l) \quad (2.4)$$

The final ICE objective function is given by (2.5)

$$RSS_{reg} = (1 - \mu) \frac{RSS}{N} + \mu V \quad (2.5)$$

where V is the sum of variance over the bands of the endmember vertices and μ is a regularization parameter that trades off between the two terms.

$$V = \frac{SSD}{M} (M - 1) \quad (2.6)$$

The regularization parameter μ has a large influence on the endmember estimation. The value of μ is constrained between $[0, 1]$. When $\mu = 1$ all the endmembers collapse to a single point near the mean of the data. When $\mu = 0$ the problem is ill-defined [31].

The ICE objective function needs to be minimized for both endmembers and proportions for each pixel in the data set. The ICE algorithm follows an iterative procedure for minimizing the objective function. For the first iteration, endmembers are randomly chosen from the image. Proportion estimation is done by performing a separate minimization of all the N terms, each of which is a quadratic programming problem [32]. After solving for proportions using quadratic programming, endmembers are estimated using (2.7)

$$\mathbf{e}_d = \left\{ \mathbf{P}^T \mathbf{P} + \lambda \left(\mathbf{I}_M - \frac{\mathbf{1}\mathbf{1}^T}{M} \right) \right\}^{-1} \mathbf{P}^T \mathbf{x}_d \quad (2.7)$$

where \mathbf{P} is the $N \times M$ proportion matrix, \mathbf{e}_d is the vector of the endmember values in the d^{th} band, \mathbf{x}_d is the vector of all the pixel values in the d^{th} band, \mathbf{I}_M is the $M \times M$ identity matrix, $\mathbf{1}$ is the M -vector of ones, and λ is given in (2.8),

$$\lambda = \frac{N\mu}{\{(M-1)(1-\mu)\}} \quad (2.8)$$

2.1.3 Sparse Endmember Estimation Methods

Estimating the number of endmembers for hyperspectral data is often a challenge. There are state of the art hyperspectral unmixing methods proposed in the literature but most of these algorithms require knowledge of number of endmembers present in the scene [33]. Hyperspectral unmixing algorithms which do not address the issue of estimating the number of endmembers from the scene may result in estimating unnecessary endmembers. This gave rise to the need of having sparse solutions for the endmember estimation problem.

This section will review some of the sparsity promotion based hyperspectral endmember estimation methods.

2.1.3.1 Sparsity Promoting Iterative Constraint Endmembers (SPICE)

The Sparsity Promoting Iterative Constraint Endmembers (SPICE) proposed by Zare et al. [33] is an extension of the Iterated Constrained Endmembers (ICE). In this method, a sparsity promoting term is added to the existing ICE objective function

which helps in pruning of unnecessary endmembers. The Sparsity Promoting Term (SPT) is given by (2.9)

$$SPT = \sum_{k=1}^M \gamma_k \sum_{i=1}^N |p_{ik}| = \sum_{k=1}^M \gamma_k \sum_{i=1}^N p_{ik} \quad (2.9)$$

where γ_k is

$$\gamma_k = \frac{\Gamma}{\sum_{i=1}^N p_{ik}^{(\text{old})}} \quad (2.10)$$

where Γ is a constant and it determines the degree to which the proportions are driven to zero. The sparsity term on the proportions forces the proportions values for unnecessary endmembers to be driven to zero. Endmembers whose corresponding proportions are zero can be discarded, resulting in sparse set of endmembers. The SPICE objective function can be written by adding the SPT term to the ICE objective function

$$RSS_{reg} = (1 - \mu) \frac{RSS}{N} + \mu V + SPT \quad (2.11)$$

Both proportions and endmembers are estimated using the same iterative method adopted in ICE. Given the endmembers estimates, proportions are solved using the least square optimization using quadratic programming approach [32] for each of the N pixels of the data sets.

2.1.3.2 Other Sparse Endmember Extraction Methods

A sparse endmember extraction method, Sparse Endmember Extraction and Demixing, is proposed by Ehler et al. [34]. This method results in sparse set of endmembers. For each \mathbf{x}_i in X (where $X = \bigcup_{i=1}^N x_i \subset R^d$) this algorithm tries to represent \mathbf{x}_i in terms of the remaining elements of X . This method assigns weights to the input data points and the data points with larger weights are selected as the potential endmembers. In order to achieve a sparse endmembers set of, a sparsity promoting prior is applied to the weights in the optimization problem given in the (2.12)

$$w_{i,\cdot} = \arg \min_{\tilde{w}} \|x_i - \sum_{j=1}^n \tilde{w}_j x_j\|_{\ell_2}^2 + \tau_i \|\tilde{w}\|_{\ell_0} \quad (2.12)$$

subject to the following constraints in (2.13)

$$\tilde{w}_j \geq 0, \quad \tilde{w}_i = 0 \quad (2.13)$$

where w_{ij} are the weights associated with the corresponding input pixels. The first term in the above equation is the reconstruction error term and the second term is the sparse penalty term. The $\tau_i \geq 0$ balances the reconstruction error term and the penalty term. Higher value of τ will place more importance on the l_0 norm term which results in more sparse solution.

Data points whose corresponding weights are zero, are later pruned resulting in a sparse representation of the data. The remaining data points are then ordered on the basis of the relation \preceq_o and \preceq_2 , which is given by $x_i \preceq_0 x_j$ if

$$\|w_{\cdot,i}\|_{l_0} \leq \|w_{\cdot,j}\|_{l_0}$$

and $x_i \preceq_2 x_j$ if

$$\frac{\|w_{\cdot,i}\|_{l_2}}{\|w_{\cdot,i}\|_{l_0}} \leq \frac{\|w_{\cdot,j}\|_{l_2}}{\|w_{\cdot,j}\|_{l_0}}$$

These two relations would create two additional indices, r_i^0 and r_i^2 for each ordered list. The first index, r_i^0 , ranks the vector in X according to the number of times a particular vector is represented by the remaining elements of \mathbf{x}_i . On the other hand, the second index, r_i^2 , is a measure of the importance of \mathbf{x}_i in the representation. It assigns higher rank to the vector \mathbf{x}_i if it plays a significant role in the representation. Following this, a final ordered list is created and the final endmember set is given by the highest rankings. A proportion matrix α_i is estimated using (2.14)

$$\alpha_{i,\cdot} = \arg \min_{\tilde{\alpha}} \|x_i - \sum_{j=1}^s \tilde{\alpha}_j e_j\|_{\ell_2}^2 + \beta_i \|\tilde{\alpha}\|_{\ell_1} \quad (2.14)$$

subject to the constraints

$$\tilde{\alpha}_j \geq 0, \quad \sum_{j=1}^s \alpha_{i,j} \approx 1$$

2.2 Review: Hyperspectral Dimensionality Reduction Methods

Hyperspectral imaging instruments collect data up to thousands of narrow wavelengths. Dimensionality Reduction (DR) is often used as a preprocessing step in the spectral unmixing of the hyperspectral data. DR in hyperspectral imaging is commonly achieved in two ways: transformation or band selection. Using a transfor-

mation, dimensionality is reduced by transforming the data into some subspace of the features that preserve most of the desired information. Commonly used transformation techniques are: Principal Component Analysis (PCA) [20], Independent Component Analysis (ICA) [35], Linear Discrimination Analysis (LDA) [36], and others.

On the other hand, feature selection (FS) methods select those features which can be efficiently used for analysis and discard those features which do not contribute in analysis. Thus in this case, a subset of features are selected which maintains or improve analysis.

The following section reviews state-of-the-art transformation based and band selection methods.

2.2.1 Transformation-Based Methods

2.2.1.1 Projection-Based Methods

As stated earlier, Principal Component Analysis (PCA) is a widely used linear transformation method applied for dimensionality reduction. PCA projects data from higher dimensional subspace to a lower dimensional subspace while maximizing data variance in the reduced dimensional space. Conventionally PCA tries to maximize the global covariance of the data ignoring the local covariance between classes. Local variance is sometimes necessary in order to effectively discriminate between several classes, moreover, PCA does not provide information about number of dimensions to retain.

In order to solve the problem incurred by PCA, a method for dimensionality re-

duction of hyperspectral image was introduced by Harsanyi et al. [37]. This method reduces dimensionality of the data by nullifying an undesired or irrelevant spectral signature so as to retain only the signature of interest. Following the linear mixing model given in (1.2), this method assumes that endmembers are given as $\mathbf{E} = (\mathbf{z}_1, \dots, \mathbf{z}_i, \dots, \mathbf{z}_{M-1}, \mathbf{d})$, where the \mathbf{E} is a $D \times M$ matrix and the $D \times 1$ column vector z_i is the spectral signature of i^{th} distinct material in the scene. Moreover, this method assumes that the last columns of \mathbf{E} is the desired signature of interest and the remaining columns are undesired signature. An operator based on orthogonal subspace projection is developed, given in (2.15), to eliminate the undesired signatures resulting in the desired signature with a random noise and reduces each pixel across d dimension into a scalar which represents a desired signature.

$$\mathbf{q}^T = \mathbf{d}^T \mathbf{P} \quad (2.15)$$

The operator shown in (2.15) is modified for the k signature of interest, which is given in (2.16),

$$\mathbf{Q} = (\mathbf{q}_1, \dots, \mathbf{q}_i, \dots, \mathbf{q}_k)^T \quad (2.16)$$

Here the hyperspectral image cube is reduced to k images which represents each signature of interest. Also each q_i is given in (2.17)

$$\mathbf{q}_i^T = \mathbf{d}_i^T \mathbf{P}_i \quad (2.17)$$

Similarly, the method proposed by Venkataraman et al. [38] utilizes the fact that high dimensional data can be effectively projected to a lower dimensional space with-

out loss of significant information. This lower dimensional space is then used for feature extraction. In this method, the input hyperspectral data is first divided into adjacent disjoint groups from which features can be extracted to perform classification. The grouping of bands is done based on the estimation of performance matrix, which is the measure of correlation between bands and maximum separability between classes. Bhattacharyya Distance (BD), Jeffery Matusita distance (JM), and area under the Receiver Operating Characteristic curve (ROC) are used as a measure of class separation. The number of groups represents the reduced dimensionality of the input hyperspectral data. After grouping, features are extracted from each group by using both supervised (LDA) and unsupervised (PCA) feature extraction methods. The algorithm concludes by classification of extracted features by parametric classifier such as Maximum Likelihood (ML) and non-parametric classifier such as k-Nearest Neighbor. Random Projection (RP) proposed by [39] is an another projection-based method. In random projection, the original D dimensional data is projected to a new D' dimensional subspace. This D' dimensional subspace is formed by a projection matrix T (with unit length columns) of dimension $D' \times D$. Where the projection matrix is determined using (2.18),

$$X_{k \times o}^T = T_{k \times b} \cdot X_{b \times o} \quad (2.18)$$

where b is the original dimension of the data being projected onto k dimensional subspace ($k < < b$), $k \times b$ is the projection matrix, o is the number of observations. From (2.18) it is clear that the RP is a linear mapping from the high dimensional space to the lower dimensional space. Since the newly projected subspace by RP still forms a simplex, endmembers can be extracted from vertices of the projected simplex

in a subspace.

Many hyperspectral dimensionality reduction methods leverage class labels. However this is sometimes not available. This limitation was addressed by a method proposed by Chen et al [40]. Instead of using the class labels as prior information, the algorithm proposed here uses a pairwise constraint which specifies whether a pair of samples belongs to same class or not. Samples belonging to same class are referred as ‘must-link constraint’ and the one belonging to different class are referred as ‘cannot-link constraint’. Using the input data, the ‘must-link’ constraints, and ‘cannot-link’ constraints, the proposed method estimates a new projection vector $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_d]$ (where d is the number of bands in the data) in low dimensional space while preserving the input data, must-link constraints, and cannot-link constraints. Based on above information, an objective function for semi-supervised dimensionality reduction (SSDR) framework is presented. The general form of the objective function is given in (2.19),

$$\mathbf{J} = \mathbf{J}_{\mathbf{D}} + \beta \cdot \mathbf{J}_{\mathbf{R}} \quad (2.19)$$

where $\mathbf{J}_{\mathbf{D}}$ is the discrimination term, $\mathbf{J}_{\mathbf{R}}$ is the regularization term, and β is the trade-off between the discrimination term and the regularization term. The discrimination term specifies the distance between samples in ‘cannot-link’ constraint and ‘must-link’ constraint. This term forces the distance to between the samples involved by the ‘cannot-link’ constraint as large as possible and the distance between the samples involved by ‘must-link’ constraint to be small as possible.

On the other hand, the regularization term is added in the objective function to use the data when few or none of the constraints (cannot-link and must-link constraints)

are available. The paper presented two types of regularization terms: $SSDR_{pca}$ and $SSDR_{sp}$. $SSDR_{pca}$ is based on Principal Component Analysis (PCA). This term preserves the global variance of the original data along with the pairwise constraints (both ‘must-link’ and ‘cannot-link’ constraints). The other term is based on sparsity. The regularization term SSD_{sp} is motivated by Sparsity Preserving Projection (SSP) [41]. Combining the regularization term and the discrimination results in the final set of projection vectors $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_d$.

In another approach presented by Zhang et al. [42], a clonal selection technique based on an Artificial Immune System (AIS) is proposed for dimensionality reduction of hyperspectral data. Both feature selection and feature weighting based on the proposed clonal selection method is presented. In the initialization step, N elements from the original space are selected to form the initial antibody population given by (2.20),

$$AB = ab_1, ab_2, \dots, ab_N \quad (2.20)$$

where N is the number of antibody population and each antibody is given as (2.21),

$$ab_i = \left(ab_i^1, ab_i^2, \dots, ab_i^{N_b} \right) \quad (i = 1, 2, \dots, N) \quad (2.21)$$

where N_b is the number of bands in the data. The generated antibody population is transformed from original feature space to binary space. This is done by assigning either ‘0’ or ‘1’ to each element of the antibody population. Here ‘0’ represents exclusion of n^{th} feature and ‘1’ represents the inclusion of n^{th} feature in the newly created binary space. Affinity of each antibody ab_i is calculated using Jeffries-Matusia (JM) distance [43]. Antibodies with the highest affinity are selected to form the new

set of high affinity antibodies and they are cloned to generate a new set. This set represents the set of better fitting antibodies. In order to replace the lowest affinity antibodies in the AB , mutation is performed on the cloned set followed by an affinity calculation. Antibodies with the highest affinity replaces the lowest affinity antibodies in the initial set AB . This cycle repeats until a maximum number of iterations are met. The final selected subset is converted from binary space to feature space by selecting the features having value 1 in the binary space. In an feature weighing approach each antibody is assigned a weight. The algorithm is iterated until the maximum number of iteration is reached. For getting final selection of bands, features are transformed from binary space to feature subset space.

2.2.1.2 Wavelet Transform-Based Methods

In the literature, it has been shown that the wavelet transform is an excellent tool for feature extraction [44]. Projection of a signal on the basis of a wavelet function helps in the separation of fine scale and large scale information from a hyperspectral signature. Discrete wavelet transform for dimensionality reduction by feature extraction is presented by Bruce et al. [45]. Mathematically, the DWT can be defined as the inner product of a finite-length input sequence and a mother wavelet and it is given in (2.22),

$$Wf(j, k) = \sum_{d=0}^{D-1} f(d) \cdot \psi_{j,k}^*(n) \quad (2.22)$$

where $Wf(j, k)$ is a DWT coefficient, $f(d)$ is a input hyperspectral signal with d spectral bands, and $\psi_{j,k}(n)$ is given in (2.23),

$$\psi_{j,k}(d) = \frac{1}{\sqrt{s_0^j}} \left(\frac{d - s_0^j \cdot k}{s_0^j} \right) \quad (2.23)$$

where s_0^j and $s_0^j \cdot k$ are the scale and translation parameters for a ‘mother wavelet’. In the proposed method, the Harr wavelet was selected as the mother wavelet, the Harr wavelet is similar to a step function given as in (2.24),

$$\psi_n = \begin{cases} 1, & 0 \leq n \leq 1/2 \\ -1, & 1/2 \leq n \leq 1 \\ 0, & \text{otherwise.} \end{cases} \quad (2.24)$$

Following the selection of ‘mother wavelet’, the Discrete Wavelet Transform (DWT) is applied on the input hyperspectral signature which decomposes the signal to separate fine-scale (detail) from the large-scale (detail) of the signals. The extracted features are later evaluated for their effectiveness and ability to discriminate between different ground labels.

Moreover, Gormus et al. [46] addresses the problem of dimensionality reduction using a wavelet transformation as an application of the Empirical Model Decomposition. Similar to the wavelet transform, emperical mode decomposition is also a signal processing method. A signal can be decomposed into several functions termed as Intrensic Mode Function (IMF) [47] by utilizing the emperical mode decomposition. Empirical model decomposition-2D has been presented in previous literature [47]. In this approach, the dimensionality is reduced from a bigger set to a smaller subset with inclusion of both spectral and spatial information from the scene. In the first step, 2D-emprical model decomposition is applied to each band of hyperspectral image in

the spatial domain resulting in the and K IMF's (where K is the number of IMF's) given in (2.25),

$$IMF_{1,k}(i,j) \quad (2.25)$$

where 1 is the 1st band of the hyperspectral image, K is the number of IMF's, and (i,j) is the spatial location a pixel. 1D-DWT (Discrete Wave Transform) is applied on the spectral domain i.e. on each pixel of the each IMF features obtained from previous step generating an another set of k IMF features given by $WIMF_k, (k=1 : K)$. A new wavelet based set of IMF features is generated by,

$$WIMF (k = 1 \cdots K) = \sum_{k=1}^K WIMF_{m,i,j} \quad (2.26)$$

R features are selected from this final set. In order to estimate the overall accuracy of the R features selected, SVM is used.

2.2.1.3 Other Methods

Nonparametric Weighted Feature Extraction for Classification (NWFE) proposed by Kuo et al. [48] is an another approach for feature extraction from hyperspectral data. The goal of this method is to find a linear transformation which maximizes the between-class scatter and minimizes within-class scatter. By assigning weights to the input hyperspectral data, a weighted mean is computed. Nonparametric between class and within class scatter matrix is estimated which assigns higher weights to the samples close to the class boundary and vice-versa. The estimated weighted mean along with distance between each sample determines the closeness of each data

point to the class boundary. The eigenvectors corresponding to largest f eigenvalues obtained from (2.27) are the final set of extracted features.

$$(S_w^{NW})^{-1} (S_b^{NW}) \quad (2.27)$$

where S_w^{NW} is the nonparametric weighted within-class scatter matrix and S_b^{NW} is the nonparametric weighted between-class scatter matrix.

In another approach by Li et al. [49], NWFE is extended by extracting features using a support vector machine. Here kernel-based feature extraction is presented and class boundaries are determined by a soft-margin SVM the Support Vector Domain Description (SVDD); followed by the application of NWFE in a Hilbert Space. Weights are estimated by estimating the distance between $\phi(x_k^i)$ (which is the feature mapping from original space to Hilbert Space of a sample i) and the boundary between class i and j (considering two classes).

Although all these projection-based dimensionality reduction methods reduce the dimensionality of the hyperspectral data, they do not retain the physical meaning of the data. By projecting data from higher dimensional space to a lower dimensional space, data loses the physically meaningful bands each of which correspond to a wavelength from the original data [50]

2.2.2 Band Selection Methods

Band selection methods determine the subset of bands that contains the needed information. Optimizing an objective function and employing a search strategy are two major approaches for band selection. Many methods have been proposed in literature

for band selection [51; 52; 53].

This section will discuss some of the band selection methods based on distance measures, independent component analysis, and mutual information based methods.

2.2.2.1 Distance Measure-Based Band Selection Methods

Approaches such as distance measure-based methods (Bhattacharya distance, Jeffreys-Matusita distance), information-theoretic approaches (divergence, mutual information measure) and eigen-analysis (Principal Component Analysis (PCA)) have been used for band selection. In [54], a joint method for band prioritization and band-decorrelation is presented. Band prioritization is achieved by two eigen-analysis based PCA criteria: (i) Maximum Variance PCA (MVPCA) and (ii) Maximum-SNR PCA (MSNRPCA) and two eigen-analysis based classification criterion: (i) Minimum-Missclassification Canonical Analysis (MMCA) and (ii) Orthogonal Subspace Projection (OSP). Bands are prioritized and selected on the basis of information content for classification. Since the band prioritization proposed here does not take spectral correlation into account this may result in two bands with similar properties, a divergence-based band-decorrelation approach is presented that measures the correlation between two band images. Based on a specified threshold, one band out of two with similar priority bands is eliminated.

Following this, Keshava [55] proposed a method for band selection that measures the distance between spectra of two materials at corresponding spectral bands. Separability of the measured spectral bands is determined by Spectral Angle Mapper (SAM). The objective here is to maximize the SAM distance between two spectra of differing materials. Similarly in another approach by Keshava [56] Spectral Angle

Mapper (SAM) and Euclidean Minimum Distance are used as tools for measuring the distance between two spectra. The Band Add-On (BAO) method is presented, which iteratively selects bands and tries to increase the angular separation between two spectra. Bands with a maximum spectral angle is selected as an initial pair, followed by the calculation of cosine of angles (β) between remaining pairs of bands. Bands with lowest angle are appended in the set of selected bands. BAO is extended to distinguish between two classes by material identification. Three methods: Average Distance Method, Minimum Distance Method, and Binary Discrimination with Similar Targets are proposed for material identification. In all these methods, material identification is performed by comparing an unknown spectra with some reference spectra. A major limitation with these methods is that, instead of considering the ability of a band to separate distinct endmembers, these methods perform band selection on the basis of distance between two bands [50]. Moreover, these distances need to be computed for all possible subset of bands, which makes these methods computationally expensive.

2.2.2.2 Independent Component Analysis (ICA)-Based Methods

ICA has received a vast attention in the literature for the band selection of hyperspectral images. Using ICA, the observed signal $\mathbf{X} = (x_1, x_2, \dots, x_n)^T$ and the source signal given by $\mathbf{S} = (s_1, s_2, \dots, s_m)^T$ are represented by the ICA unmixing model by (2.28)

$$S_{m \times p} = W_{m \times n} X_{n \times p} \quad (2.28)$$

where m is the number of source signals, n is the number of observed signals, and W is the unmixing matrix or weight matrix given by (2.29)

$$\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m)^T \quad (2.29)$$

Exploiting the ICA, Du et al. [57] proposed a method for band selection for hyperspectral images. The proposed method assigns weights to each band according to a band's contribution to the ICA transformation. Following this, an average absolute weight coefficient for all the bands is determined using (2.30)

$$\bar{w}_j = \frac{1}{M} \sum_{i=1}^M |w_{ij}| \quad (2.30)$$

where $j = (1, 2, \dots, n)$ and M is the number of materials present in the scene. The bands with higher absolute weight coefficient contributes more to the ICA transformation and, hence, contains more spectral information and, thus, these bands are kept during band selection.

2.2.2.3 Mutual Information-Based Methods

Feature selection using a suboptimal search strategy has been developed in [58]. A solution of the feature selection problem is represented as a binary string where 1 represents a selected feature and 0 represents a discarded feature. Two algorithms are proposed: Steepest-Ascent Search Strategy and Fast Algorithm for Constraint Search. The goal of the proposed method is to reduce the set of features. In order to achieve this a criterion function J is presented which is the maximization of a separability index. The criterion function is calculated using a pre-classified reference

set of pattern which tries to maximize the criterion function which further leads to the selection of relevant features and, hence, dimensionality reduction. Results indicate that the proposed method provides better solution for feature selection than the Sequential Forward Selection (SFS) and Sequential Forward Floating Selection (SFFS). However, high computational cost is the main limitation of this method.

Guo et al. [59] present another method for band selection and classification using Mutual Information (MI). Mutual information is a measure of statistical dependence between two random variables and, hence, bands are selected by measuring the statistical dependence between two bands of a hyperspectral image. Instead of using class-label ground truth, the method proposed here uses a reference map. This method is useful when ground truth is not available. The reference map is generated from the spectral signature of frequently encountered material using a standard spectral library. MI is estimated between the d^{th} band and the estimated reference map. This is followed by ranking the bands in decreasing order of their MI values. Bands with MI values greater than some threshold are taken as the final set of selected bands. In order to avoid redundant information within multiple bands correlation between bands is also taken into account along with MI measure for the selection of bands.

MI has been also exploited in clustering based hyperspectral band selection using information measures [60]. Hierarchical clustering based on Ward’s linkage [61] has been used. Using mutual information and divergence-based criteria, clusters of bands that share same information are formed and a representative band is selected from each group of bands. Mutual information-based dissimilarity criteria try to identify the subset of bands that are independent. For divergence-based dissimilarity criteria, Kullback-Leibler divergence is used. This criterion measures the dissimilarity

between two bands. After measuring the dissimilarity between the bands, the bands are grouped into clusters in order to minimize the intra-cluster variance and maximize the inter-cluster variance. A weight is assigned to each band in each cluster on the basis of dissimilarity measures. A higher weight value represents that a band has on average a small distance from the other bands in the cluster and, thus, a high-correlation with respect to the other bands. So, bands having high correlation with other bands in the cluster can represent the cluster very well. Bands with the highest average correlation coefficient with respect to other bands are selected as the representative band. At the end of the algorithm k bands representing each cluster is selected as the final bands.

2.2.2.4 Other Methods

An approach for hyperspectral band selection based on higher order moments was proposed by Du [62]. Since most of the time in hyperspectral unmixing class information is not available, target detection in hyperspectral imagery is carried out by looking for deviation from the background distribution. In the proposed approach, band selection is performed on the basis of two higher order moments: skewness and kurtosis. The algorithm begins by estimating either the skewness or kurtosis value for each band of the image. A large value for skewness or kurtosis represent higher information content in the corresponding band. A set Q of the final selected bands is initialized by the first band and divergence is calculated between the d^{th} and $(d-1)^{th}$ band. Kullback-Leibler (K-L)divergence is the measure of similarity or dissimilarity between two probability distributions. If the K-L divergence is less then a predefined threshold, the bands with smaller values of the higher order moments are discarded

and final set of selected bands is populated accordingly resulting in the selection of P bands. Then number of bands to be selected P , is determined by Neyman-Pearson detection theory-based eigen-threshold method proposed by Harsayni et al. [63].

In an another approach the Minimum Estimated Abundance Covariance (MEAC) proposed by Yang et al. [64] exploits the Sequential Forward Selection (SFS) method. In the initialization step, a band is randomly selected (say A_1) and other bands are projected onto to the orthogonal subspace of initially selected band. The band which is the most dissimilar to the A_1 will form the first best band pair (A_1, A_2). The algorithm proceeds by estimating the third best band (A_3) by projecting the remaining bands on the the orthogonal subspace of A_2 and so on. The algorithm concludes when the most dissimilar band found in n^{th} iteration is same as in $(n - 1)^{th}$ iteration.

These all methods has the drawback of not considering unmixing while performing band selection and so it is not certain whether the selected bands can achieve best performance in terms of distinct endmember estimation.

2.2.2.5 Hyperspectral Band Selection and Endmember Detection using Sparsity Promoting Priors (B-SPICE)

The use of sparsity promoting priors for band selection has been exploited by Zare et al. [50]. They developed a Hyperspectral Band Selection and Endmember Detection algorithm using sparsity promoting priors (BSPICE) for simultaneous band selection and endmember estimation. This method extends SPICE [33] by adding spectral band weights and a sparsity promoting term for bands to the SPICE objective function. The algorithm tries to estimate endmembers, the number of endmembers present in a hyperspectral scene, proportion values, and band weights simultaneously. The

BSPICE objective function is shown in (2.31)

$$BSPICE = \eta \frac{RSS_B}{N} + \beta SSD_B + SPT + BST \quad (2.31)$$

where

$$RSS_B = \sum_{n=1}^N \left(\mathbf{WX}_n - \sum_{m=1}^M p_{nm} \mathbf{WE}_m \right)^T \left(\mathbf{WX}_n - \sum_{m=1}^M p_{nm} \mathbf{WE}_m \right), \quad (2.32)$$

$$SSD_B = \sum_{m=1}^{M-1} \sum_{l=m+1}^M (\mathbf{WE}_m - \mathbf{WE}_l)^T (\mathbf{WE}_m - \mathbf{WE}_l), \quad (2.33)$$

$$SPT = \sum_{m=1}^M \gamma_m \sum_{n=1}^N p_{nm}, \quad (2.34)$$

$$\mathbf{W} = diag(w_1, \dots, w_d), \quad (2.35)$$

$$\gamma_m = \Gamma / \sum_{n=1}^N p_{nm}, \quad (2.36)$$

$$BST = \sum_{j=1}^d \lambda_j |w_j| = \sum_{j=1}^d \lambda_j w_j, \quad (2.37)$$

$$\lambda_j = \frac{\Lambda \left(\left(\sum_{m=1}^M \frac{1}{N} \sum_{n=1}^N p_{nm} (x_{nj} - e_{mj})^2 \right) + 1 \right)}{\left(\sum_{m=1}^M (e_{mj} - \mu_{0j})^2 \right) + 1}, \quad (2.38)$$

N is the number of pixels, M is the number of endmembers, d is the number of bands, w_d is the weight of d^{th} band, η and β are the constant coefficients parameters for the RSS and SSD terms respectively, Γ is a constant controlling the degree of sparsity among the endmembers, Λ is a tunable parameter controlling the degree of sparsity among the band weights, μ_0 is the global data mean, x_{nj} is the n^{th} pixel in j^{th} band, and e_{mj} is the m^{th} endmember in j^{th} band. The band weights are subject to the constraints given by

$$w_j \geq 0, \quad j = 1, 2, \dots, d \quad (2.39)$$

$$\sum_{j=1}^d w_j = d \quad (2.40)$$

The λ_j value is motivated by within-class to between-class scatter ratios. For each endmember, there exist the data class which has higher proportion value for that endmember. The numerator in λ_j is the error between those data points and their corresponding estimated endmembers weighted by associated proportion values. The denominator is the difference between estimated endmembers and global data mean. The objective here is to estimate endmembers which are close to the data with high proportion values but are far from other endmembers.

Algorithm 1 BSPICE [50]

```
1: iter = 1;
2: BandUpdate = 0;
3: n = iteration frequency of band updates
4: StartBandUpdate = the ireration to start updating band weights
5: while ( $ObjValue - PrevObjValue)^2 > ChgThresh$  do
6:   Update proportion values
7:   Update Endmember values
8:   BandUpdate = 0;
9:   if ( $iter > StartBandUpdate$ ) and ( $mod_n(iter) == 0$ ) and ( $BandUpdate == 0$ ) then
10:    Update Band Weights
11:    Prune Weights
12:    if  $norm(PrevBand - CurrentBandWt) < BandChgThresh$  then
13:      BandUpdate = 1;
14:    end if
15:   else
16:     Update Objective Function Value, ObjValue
17:     iter = iter + 1;
18:   end if
19: end while
```

Minimization of the objective function is performed iteratively. The minimization iterates between solving for proportions, endmembers, and band weights. Endmembers are solved by (2.41).

$$\mathbf{e}_j = \left\{ \mathbf{P}^T \mathbf{P} + \lambda \left(\mathbf{I}_M - \frac{\mathbf{1}\mathbf{1}^T}{M} \right) \right\}^{-1} \mathbf{P}^T \mathbf{x}_j \quad (2.41)$$

Given endmembers and band weights, N quadratic programming steps (where N is the number of data points) are employed for solving proportion values with respect to following constraints.

$$p_{nm} \geq 0, \quad m = 1, \dots, M \quad \text{and} \quad \sum_{m=1}^M p_{nm} = 1 \quad (2.42)$$

Similarly, a single quadratic programming step is employed for solving band weights, given endmembers and proportion values. Since the band weights and endmember values depend on each other, endmembers needs to be estimated before the start of band weighting and pruning. An optimization schedule given by a band weight threshold (threshold to start band weighting), band prune threshold (threshold to start pruning of bands) and band prune frequency (frequency of pruning bands after band prune threshold is reached) is presented. The BSPICE algorithm can be summarized in Algorithm 1.

2.3 Review: Piece-Wise Convex Unmixing

This section will review some of the state-of-the-art hyperspectral unmixing methods that performs piece-wise convex unmixing

2.3.1 Dirchlet Process-Based Methods

Most of the existing endmember detection algorithms assume that all pixels in a hyperspectral image are convex combinations of a single set of endmembers. However, some hyperspectral images can be better represented by the union of multiple convex sets. The piece-wise convex representation of the hyperspectral image was introduced in [21] and [65] with the Piecewise Convex Endmembers (PCE) method. PCE uses a Dirchlet process prior to estimate number of the convex regions needed to represent a input hyperspectral data and estimates the proportions for each set of endmembers. Endmembers and proportions are estimated by maximizing a likelihood function using a stochastic Expectation Maximization (EM) algorithm [66]. In statistics, likelihood describes the parameter, given the outcome of certain event. PCE has three major limitations. First, this method is computationally expensive. Second, in PCE, the convex boundaries are crisp [67], which requires each data point to belong to only one convex region. Third, PCE may not converge to a global optima. In order to overcome these limitations, two different methods were proposed.

The first and the third limitation were addressed by the Sampling Piecewise Convex Unmixing and Endmember Extraction (S-PCUE) method, proposed by Zare et al. [66]. S-PCUE uses a Metropolis-within-Gibbs sampling method to sample desired parameters. Apart from spectral unmixing for multiple sets of endmembers,

this algorithm considers spectral variability in endmembers. This method is more computationally efficient than the PCE and, if sampled long enough, should converge to a global optimum.

The second limitation is overcome in the Piecewise Convex Multiple-Model End-member Detection and Spectral Unmixing (PCOMMEND) [68] algorithm. Here, a fuzzy clustering approach is employed instead of a crisp clustering approach. Multiple sets of endmembers and their corresponding proportion maps are found using an iterative fuzzy clustering method motivated by the Fuzzy C-means algorithm [69] and the Fuzzy C-Shells algorithm [70]. PCOMMEND algorithm is described in section 2.3.4, as it is the basis for methods proposed here.

2.3.2 Spectral and Spatial Information-Based Piece-wise Convex Methods

Hyperspectral unmixing methods which account for both spectral and spatial information along with multiple sets of endmembers is also presented in the literature. The method given by Bchir et al. [67] uses spectral and spatial information from a hyperspectral scene to conduct unmixing. Also, instead of assuming a global convex geometry model, this method assumes hyperspectral scene to be a set of multiple regions. This method also allows a point to belong to multiple groups with different membership degrees. Similarly, in S-PCOMMEND by Zare et al. [71] and [72], PCOMMEND is extended to perform spectral unmixing using both spectral and spatial information. In S-PCOMMEND, neighboring pixels in an image are encouraged to have similar fuzzy membership values for the different convex regions. This is achieved with the inclusion of spatial information.

Spatial and spectral information is also incorporated in piece-wise convex spatial-spectral unmixing of hyperspectral imagery using probabilistic and fuzzy clustering by Zare et al. [73], by merging FLICM [74] and PFCM [75] inspired terms into PCOMMEND. Spatial information is incorporated using an approach motivated by FLICM by encouraging similar membership values to the neighboring pixels in the image in different convex regions. The Possibilistic Fuzzy Clustering (PFCM) was first proposed by Pal et al. [75]. The method proposed here uses typicality values and membership values for clustering the data. Like memberships, typicality values are often used in the literature for clustering. Typicality was introduced by [76] and it is defined as the measure of internal similarity and external dissimilarity of a data point with the other data points. Internal similarity is a measure of the similarity of a data point with other data points in the same group. On the other hand, external dissimilarity is a measure of the dissimilarity of a data point to the data points in a different group. The resultant typicality is found by aggregating internal similarity and external dissimilarity. This method (piece-wise convex spatial-spectral unmixing of hyperspectral imagery using probabilistic and fuzzy clustering) results in proportion maps for each set of endmembers weighted by estimated memberships and typicality values for each set of endmembers (i.e. for each convex region of the hyperspectral data).

2.3.3 Other Methods

Often in piece-wise convex unmixing, parameters such as the number of endmember sets and the number of endmembers per endmember set is hard to determine. This issue is addressed by Anderson et al. [77] using a spectral unmixing cluster validity

index for multiple sets of endmembers. In this method, a cluster validity index for multiple sets of endmembers is developed which aids in the estimation of parameters for piece-wise convex methods. In general, a cluster validity index is the measure of ‘goodness’ of any clustering algorithm. As stated earlier, in piece-wise convex hyperspectral unmixing algorithms a data point can belong to multiple convex regions. This approach estimates the ‘goodness’ of the outcome of such an algorithm.

Although, all these methods are effective in performing spectral unmixing of non-convex data sets, dimensionality reduction of the hyperspectral data may help unmixing. Thus, the proposed work extends PCOMMEND [68] and B-SPICE[50] to perform unmixing with the multiple endmember sets and simultaneous band weighting.

2.3.4 PCOMMEND

This section will discuss the PCOMMEND algorithm in detail. PCOMMEND estimates multiple sets of endmembers and their corresponding proportion maps.

The PCOMMEND algorithm estimates endmember and proportion values by minimizing the objective shown in (2.43),

$$J_p(\mathbf{E}, \mathbf{P}, \mathbf{U}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q (\mathbf{x}_n - \mathbf{E}_c \mathbf{p}_{cn})^T (\mathbf{x}_n - \mathbf{E}_c \mathbf{p}_{cn}) + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \right) \quad (2.43)$$

such that $p_{cnm} \geq 0$, $\sum_{m=1}^M p_{cnm} = 1$, $u_{cn} \geq 0$ and $\sum_{c=1}^C u_{cn} = 1$ $\forall n$ where \mathbf{x}_n is a $d \times 1$ column vector representing the n^{th} pixel, C is the number of endmember sets being estimated, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_c is a $d \times M$ matrix such that each column of \mathbf{E}_c , \mathbf{e}_{cm} , is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, weight u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, α is a fixed

parameter used to balance the two terms of the objective, and q is a fixed parameter which controls the degree of sharing across endmember sets.

The first term of this objective is an error term computed using the squared Euclidean distance between each pixel and its estimated with endmember and proportion values. The second term is called as Sum of Squared Distance (SSD). This term ensures the tight fitting of the endmembers across the convex region (i.e. it is used to constrain the size of each convex region by computing the squared Euclidean distance between each pair of endmembers in an endmember set). The membership values, u_{cn} , in the first term provide a fuzzy partitioning of the input scene by indicating the degree to which pixel n is in the c^{th} endmember set.

Algorithm 2 PCOMMEND [68]

- 1: Set parameters- the number of endmember sets, C , the number of endmembers per set, M , the fuzzifier q , and regularization parameters α
- 2: Initialize \mathbf{U} , \mathbf{E} , and $\{\mathbf{p}_{cn}\}_{c=1,\dots,C;n=1,\dots,N}$

REPEAT

 Update endmembers matrices $\{\mathbf{E}_c\}_{c=1,\dots,C}$ using Equation 2.44

 Update proportion values, $\{\mathbf{p}_{cn}\}_{c=1,\dots,C;n=1,\dots,N}$, using Equations 2.45 and 2.45 and renormalize

 Update the memberships using Equation 2.47

UNTIL convergence

Endmembers, proportions and membership values are estimated using alternating optimization. Following initialization of the algorithm, the endmembers, proportions, and membership values are updated iteratively by minimizing the objective function. Update equation for endmembers is solved by $\frac{\partial J}{\partial \mathbf{E}_c} = 0$ and is given as,

$$\mathbf{E}_c = \left(\sum_n u_{cn}^q \mathbf{p}_{cn} \mathbf{p}_{cn}^T + 2\alpha (M \mathbf{I}_{M \times M} - \mathbf{1}_{M \times M}) \right)^{-1} \times \left(\sum_n u_{cn}^q \mathbf{p}_{cn} \mathbf{x}_n^T \right) \quad (2.44)$$

where $\mathbf{I}_{M \times M}$ is an $M \times M$ identity matrix and $\mathbf{1}_{M \times M}$ is an $M \times M$ matrix of ones. Adding a Lagrange multiplier term to enforce the sum-to-one constraint and applying the Karush Kuhn Tucker conditions [78] to enforce the nonnegativity constraint results in the following update equation for proportions,

$$\mathbf{p}_{cn} = \max \left((\mathbf{E}_c^T \mathbf{E}_c)^{-1} \left(\mathbf{E}_c^T \mathbf{x}_n - \frac{\lambda_c}{2} \mathbf{1}_{M \times 1} \right), 0 \right) \quad (2.45)$$

where λ_c is given as,

$$\lambda_c = 2 \frac{\mathbf{1}_{1 \times M} (\mathbf{E}_c^T \mathbf{E}_c)^{-1} \mathbf{E}_c^T \mathbf{x}_n - 1}{\mathbf{1}_{1 \times M} (\mathbf{E}_c^T \mathbf{E}_c) \mathbf{1}_{M \times 1}} \quad (2.46)$$

where $\mathbf{1}_{1 \times M}$ is the $1 \times M$ matrix of ones. Similarly the update equation for the membership values is found by adding Lagrange multiplier term to the objective function to enforce the sum-to-one constraints on memberships,

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left(\frac{(\mathbf{x}_n - \mathbf{E}_c \mathbf{p}_{cn})^2}{(\mathbf{x}_n - \mathbf{E}_k \mathbf{p}_{kn})^2} \right)^{\frac{1}{q-1}}} \quad (2.47)$$

The PCOMMEND algorithm is summarized in Algorithm 2.

Chapter 3

Methodology

The focus of this research is hyperspectral unmixing of non-convex data sets and with simultaneous dimensionality reduction. Two new approaches, Simultaneous Band Weighting and Spectral Unmixing for Multiple Endmember sets - Simultaneous Clustering and Attribute Discrimination (SUBSUME-SCAD) and Simultaneous Band Weighting and Spectral Unmixing for Multiple Endmember sets - Band Sparsity Term (SUBSUME-BST) are proposed. These algorithms perform spectral unmixing with multiple endmember sets and estimate the number of bands to be retained by employing band weighting. SUBSUME-SCAD is an extension of Simultaneous Clustering and Attribute Discrimination (SCAD) [79] and PCOMMEND [68] and SUBSUME-BST is an extension of B-SPICE [50] and PCOMMEND. Both weight decay and sparsity promoting approaches of SUBSUME-SCAD and SUBSUME-BST are presented.

3.1 Review: Simultaneous Clustering and Attribute Discrimination (SCAD)

Simultaneous Clustering and Attribute Discrimination (SCAD) proposed by Frigui et al. [79] performs simultaneous clustering and cluster-specific feature weighting by minimizing the objective shown in (3.1),

$$J_{scad}(\mathbf{C}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - c_{cd})^2 + \sum_{c=1}^C \delta_c \sum_{d=1}^D v_{cd}^2 \quad (3.1)$$

where C is the number of clusters, N is the number of data points, u_{cn} is the membership values for the n^{th} data point in the c^{th} cluster, q is a fixed “fuzzifier” parameter controlling the degree of sharing across clusters, D is the number of dimensions/features of the input data, x_{nd} is the d^{th} feature of the n^{th} data point, c_{cd} is the d^{th} feature of the c^{th} cluster center, δ_c is a parameter used to balance the terms of the objective, and v_{cd} is the feature weight for the d^{th} feature in cluster c . Furthermore, the membership values, u_{cn} , and the feature weights, v_{cd} , are constrained such that

$$u_{cn} \in [0, 1] \quad \sum_{c=1}^C u_{cn} = 1 \quad \forall n = \{1, \dots, N\}, \quad (3.2)$$

$$v_{cd} \in [0, 1] \quad \sum_{d=1}^D v_{cd} = 1 \quad \forall c = \{1, \dots, C\} \quad (3.3)$$

The SCAD objective consists of two terms. The first term follows the form of the fuzzy c-means clustering algorithm objective function [69] and is used to estimate cluster centers and a fuzzy partition of the input data with the addition of weighted feature values (through the inclusion of the v_{cd} weights for each dimension). The second term in the objective function is used to promote small feature weights (and,

thus, is minimized when all feature weights are equal to $\frac{1}{D}$). The SCAD parameters are estimated using alternating optimization. Update equation for weights is found by adding Lagrange multiplier term to the objective function and is given by (3.4)

Algorithm 3 SCAD [79]

- 1: Fix the number of clusters C ;
- 2: Fix the fuzzifier $q \in [1, \infty)$;
- 3: Initialize the centers;
- 4: Initialize the fuzzy partition matrix U ;

REPEAT

Compute $(x_{nd} - C_{cd})^2$
 for $1 \leq c \leq C, 1 \leq n \leq N, 1 \leq d \leq D$
 Update weights v_{cd} by using (3.4)
 Update partition matrix u_{cn} by using (3.6)
 Update centers C_{cd} by using (3.7)
 Update delta δ_c using (3.5)

UNTIL center stabilize

$$v_{cd} = \frac{1}{d} + \frac{1}{2\delta_c} \sum_{n=1}^N (u_{cd}^q) \left[\frac{\|x_n - c_n\|^2}{n} - (x_{nd} - c_{cd})^2 \right] \quad (3.4)$$

The default value of weights is $\frac{1}{d}$, if the difference between the centers and data in (3.4) is zero. The second term in (3.4) is a bias which can be either positive or negative. The bias measures the compactness of the features, which is positive when the projected distance along the corresponding dimension is less than the average projected distance along all dimensions. The term δ_c shows that importance of second term relative to the first term. If δ_c is too small, then the first term dominates which results in the relevance of only one feature in cluster c and hence it assigns a weight of 1 to one feature, while the remaining features are assigned weights of zero. On the other hand,

if δ_c is too large, then second term will dominate and every feature will be assigned equal weight of $\frac{1}{d}$. Value of δ_c is updated in each iteration using (3.5)

$$\delta_c^{(t)} = K \frac{\sum_{n=1}^K \left(u_{cn}^{(t-1)}\right)^q \sum_{d=1}^D v_{cd}^{(t-1)}}{\sum_{d=1}^D \left(v_{cd}^{(t-1)}\right)^2} \quad (3.5)$$

where K is a constant and t is the iteration. The update equation for the membership values is determined by taking partial derivative of (3.1) with respect to u_{cn} and is given by (3.6)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left[\frac{\sum_{d=1}^D v_{cd} (x_{nd} - c_{kd})^2}{\sum_{d=1}^D v_{kd} (x_{nd} - c_{kd})^2} \right]^{\frac{1}{q-1}}} \quad (3.6)$$

Similarly, centers are updated by taking the partial derivative of (3.1) with respect to C_{cd} and is given by (3.7). The SCAD algorithm can be summarized in Algorithm 3

$$C_{cd} = \begin{cases} 0 & \text{if } v_{cd} = 0 \\ \frac{\sum_{n=1}^N (u_{cn})^q x_{nd}}{\sum_{n=1}^N (u_{cn})^q} & \text{if } v_{cd} > 0 \end{cases} \quad (3.7)$$

3.2 Weight Decay Approaches

Weight decay regularization terms are widely used the neural network literature. In order to generalize effectively beyond the training set, neural networks can be trained by using some form of regularization [80]. Also, regularization allows for smoother network mappings by adding a penalty term to the error function [81]. According to

Denker et al. [82], stabilization can be performed in two ways: *formal* and *structural*. In formal stabilization, an extra term, also known as penalty term, is added to the error function. The weight decay term is most widely used penalty term used in the neural networks literature. This penalty term can be defined as the sum of the squares of all the parameters in the network and can be written as in (3.8)

$$\text{Penalty Term} = \sum_{j=1}^N w_j^2 \quad (3.8)$$

where \mathbf{w} is the weight vector, N is the number of parameters in the network, and the sum is over all the weights and biases. In Bayesian theory, the weight decay function corresponds to a prior over weights. The regularized cost function [80] can be written as in (3.9)

$$M(\mathbf{w}) = \beta E_D(\mathbf{W}) + \alpha E_w(\mathbf{W}) \quad (3.9)$$

where E_D is error function or measure of misfit in the data, E_w is the penalty term, and $\alpha, \beta \geq 0$ are the regularization parameters balancing between E_D and E_w . The probabilistic relation corresponding to (3.9) can be determined as by taking the logarithm of the negative of (3.10) and ignoring the constants,

$$P(\mathbf{w}|\mathbf{D}) \propto P(\mathbf{D}|\mathbf{w}) P(\mathbf{w}) \quad (3.10)$$

where $P(\mathbf{w}|\mathbf{D})$ is the posterior density in weight space, $P(\mathbf{D}|\mathbf{w})$ is the likelihood of the data \mathbf{D} , and $P(\mathbf{w})$ is the prior density on weights.

The usual choice (the zero-mean Gaussian prior) is given as in (3.11)

$$E_w = \frac{1}{2} \sum_{j=1}^W w_j^2 \quad (3.11)$$

This section will introduce SUBSUME-SCAD and SUBSUME-BST with a weight decay term. The general form of the objective function for SUBSUME-weight decay approach is given by (3.12)

$$J_{SUBSUME-WD} = \sum_{c=1}^C [(1 - \alpha) RSS + \alpha SSD + \delta WD] \quad (3.12)$$

where

$$RSS = \sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \quad (3.13)$$

$$SSD = \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \quad (3.14)$$

$$WTD = \delta_c \sum_{d=1}^D v_{cd}^2 \quad (3.15)$$

where u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in endmember set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, δ_c is the fixed parameter values used to balance the terms of the objective function.

α is a regularization parameter between $[0, 1]$ used to balance the various terms of the objective function. $\alpha = 0$ will drive the SSD term of the objective function to

0 and thus results in an ill posed problem of solutions. On the other hand, $\alpha = 1$ will encourage all of the endmembers to lie near the global mean of the data.

The RSS term of this objective function is a least squares term

$$RSS = -\frac{1}{2} \sum_{n=1}^N u_{cn} \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)^T \text{diag}(\mathbf{V}_c) \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right) \quad (3.16)$$

Taking the exponential of the negative of the (3.16) would result in,

$$RSS = \ln e^{-\frac{1}{2} \sum_{n=1}^N u_{cn} \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)^T \text{diag}(\mathbf{v}_c) \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)}$$

The general form of Gaussian distribution is given by (3.17),

$$N(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}} \exp \left[\frac{(x-\mu)^2}{2\sigma^2} \right] \quad (3.17)$$

From RSS term and (3.17) it can be observed that the exponential in the RSS term follows the Gaussian distribution with mean $\sum_{m=1}^M \mathbf{E}_{cm} p_{cnm}$ and variance of 1. By taking the exponential of the weight decay term and adding to the RSS term, we can rewrite the RSS term as given in Least Square Weight Decay term,

$$LSWD = \ln e^{\left[-\frac{1}{2} \sum_{n=1}^N u_{cn} \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)^T \text{diag}(\mathbf{v}_c) \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right) + \delta \sum_{c=1}^C \sum_{d=1}^D v_{cd}^2 \right]}$$

or,

$$LSWD = \ln \left[e^{-\frac{1}{2} \left[\sum_{n=1}^N u_{cn} \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)^T \text{diag}(\mathbf{v}_c) \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right) \right]_e \delta \sum_{c=1}^C \sum_{d=1}^D v_{cd}^2} \right]$$

From the LSWD term, it is clear that the weight decay term is a Gaussian prior with mean 0 and precision δ_c .

3.2.1 SUBSUME-SCAD-WD

The SUBSUME-SCAD-WD algorithm with a weight decay term estimate parameters \mathbf{E} , \mathbf{P} , \mathbf{U} , and \mathbf{V} by minimizing the objective function given by (3.18)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{d=1}^D v_{cd}^2 \right) \quad (3.18)$$

subject to the constraints given by (3.19),

$$\begin{aligned} p_{cnm} &\geq 0; \sum_{m=1}^M p_{cnm} = 1 \\ u_{cn} &\geq 0; \sum_{c=1}^C u_{cn} = 1, \\ v_{cd} &\geq 0; \sum_{d=1}^D v_{cd} = D \end{aligned} \quad (3.19)$$

where u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets,

v_{cd} is the band weight for the d^{th} band in endmember set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, δ is the fixed parameter values used to balance the terms of the objective function.

The weight decay term encourages small band weights. Pseudo-code of SUBSUME-SCAD-WD with weight decay term is given by algorithm (4)

Algorithm 4 SUBSUME-SCAD Weight Decay Approach

```

1: Set number of clusters ( $C$ ), number of endmembers per cluster ( $M$ ), fuzzifier ( $m$ ),
   threshold for condition ( $threshold$ ),  $\alpha$ ,  $\delta$ , and maximum number of iteration
2: condition = 0;
3: iteration = 1;
4: Initialize endmembers, fuzzy membership, and weights
5: for  $i = 1$  to number of clusters do
6:   Initialize proportions
7: end for
8: for  $j = 1$  to maximum number of iteration do
9:   Update proportions  $\mathbf{P}$  using (3.21)
10:  Update endmembers  $\mathbf{E}$  using (3.23)
11:  Update weights  $\mathbf{V}$  using (3.24)
12:  Update fuzzy membership  $\mathbf{U}$  using (3.20)
13:  Estimate stopping criterion-  $Cond^j = \text{norm}(\mathbf{U}^j - \mathbf{U}^{j-1}) + \text{norm}(\mathbf{P}^j - \mathbf{P}^{j-1}) +$ 
       $\text{norm}(\mathbf{E}^j - \mathbf{E}^{j-1})$ 
14:  if  $(abs(Cond^j - Cond^{j-1})) < threshold$  then
15:    Break;
16:  end if
17:  iteration = iteration + 1;
18: end for

```

Refer section A.1 for detailed derivation of all the update equations

3.2.1.1 Update Equation: Membership

The update equation for the membership values is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint to the SUBSUME-SCAD-WD objective function (3.18)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left[\frac{\sum_{d=1}^D v_{cd}(x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}(x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right]^{\frac{1}{q-1}}} \quad (3.20)$$

3.2.1.2 Update Equation: Proportions

Minimizing the objective function, J , in (3.18) with respect to \mathbf{p}_{cn} by adding a Lagrange multiplier term to enforce the sum-to-one constraint and applying the Karush-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraints on the proportion values, results in the following update for the proportion values,

$$\mathbf{p}_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T \right) \quad (3.21)$$

where λ_c is given by,

$$\lambda_c = 2 \frac{1 - \mathbf{1}_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T}{\mathbf{1}_{1 \times M} \left(u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \mathbf{1}_{M \times 1}}. \quad (3.22)$$

3.2.1.3 Update Equation: Endmembers

As there are no constraints on endmembers, the update equation for endmembers is found by taking the partial derivative of the SUBSUME-SCAD-WD objective function

in (3.18) with respect to \mathbf{E}_{cd}

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn}^T \mathbf{p}_{cn} + \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn}^T x_{nd} \right). \quad (3.23)$$

3.2.1.4 Update Equation: Weights

The update equation for the weights is found by solving $\frac{\partial J}{\partial v_{cd}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-D constraint to the SUBSUME-SCAD-WD objective function (3.18)

$$v_{cd} = 1 + \frac{1}{2\delta} \sum_{n=1}^N u_{cn}^q \left[\frac{\|\mathbf{x}_n - \mathbf{E}_c \mathbf{p}_{cn}\|_2^2}{D} - (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \right]. \quad (3.24)$$

3.2.2 SUBSUME-BST-WD

The SUBSUME-BST-WD algorithm with a weight decay term estimate parameters \mathbf{E} , \mathbf{P} , \mathbf{U} , and \mathbf{V} by minimizing the objective function given by (3.25)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M \sum_{d=1}^D (e_{cmd} - e_{ckd})^T (\mathbf{e}_{cmd} - \mathbf{e}_{ckd}) + \mathbf{BST} \right) \quad (3.25)$$

subject to the constraints given in (3.19)

$$BST = \sum_{d=1}^D \delta_{cd} v_{cd}^2 \quad (3.26)$$

where δ_{cd} is given as in (3.27)

$$\delta_{cd} = \frac{\Delta \left(\left(\sum_{m=1}^M \frac{1}{A} \sum_{n=1}^N u_{cn} \mathbf{p}_{cn} (x_{nd} - e_{cmd})^2 \right) + 1 \right)}{\left(\sum_{m=1}^M (e_{cmd} - \mu_{0cd})^2 \right) + 1} \quad (3.27)$$

where Δ is a tunable parameter controlling the degree of sparsity among the band weights and μ_{0cd} is given by (3.28)

$$\mu_{0cd} = \frac{\sum_{n=1}^N x_{nd} u_{cn}}{\sum_{c=c} u_{cn}} \quad (3.28)$$

Also A is given by (3.29)

$$A = \sum_{n=1}^N u_{cn} \quad (3.29)$$

u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in endmember set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, and as mentioned in the previous section 3.2.1, α is a regularization term to balance the RSS and the SSD term of the objective function.

Like the previous objective function the first term in (3.25) is a weighted RSS term which encourages bands with low error to have high weights and the second term is a SSD term which ensures a tight fit of the data with the endmembers.

The numerator of the BST term is the measure of dissimilarity between the data and the estimated endmembers. The bigger the dissimilarity, smaller the weight is encouraged to be. If dissimilarity is small it means that the particular band is capable of separating the various material classes, which results in higher weight to the band. If for any band d , dissimilarity is low, then that band is being assigned higher weight.

The denominator of the BST term represents dissimilarity between endmember and global mean and encourage large weight for endmember far from the mean.

The v_{cd}^2 term in BST is a weight decay term which encourages small weights. The pseudo-code for SUBSUME-BST weight decay approach is given by algorithm (5)

Algorithm 5 SUBSUME-BST Weight Decay Approach

```

1: Set number of clusters ( $C$ ), number of endmembers per cluster ( $M$ ), fuzzifer ( $m$ ),
   threshold for condition ( $threshold$ ),  $\alpha$ , tunable parameter ( $\Delta$ ), and maximum
   number of iteration
2: condition = 0;
3: iteration = 1;
4: Initialize endmembers, fuzzy membership, and weights
5: for  $i = 1$  to number of clusters do
6:   Initialize proportions
7: end for
8: Initialize  $\delta_{cd}$  by number of bands
9: for  $j = 1$  to maximum number of iteration do
10:   Update proportions  $\mathbf{P}$  using (3.31)
11:   Update endmembers  $\mathbf{E}$  using (3.33)
12:   Update weights  $\mathbf{V}$  using (3.34)
13:   Update fuzzy membership  $\mathbf{U}$  using (3.30)
14:   Update delta  $\delta_{cd}$  using (3.27)
15:   Estimate stopping criterion-  $Cond^j = \text{norm}(\mathbf{U}^j - \mathbf{U}^{j-1}) + \text{norm}(\mathbf{P}^j - \mathbf{P}^{j-1}) +$ 
       $\text{norm}(\mathbf{E}^j - \mathbf{E}^{j-1})$ 
16:   if  $(abs(Cond^j - Cond^{j-1})) < threshold$  then
17:     Break;
18:   end if
19:   iteration = iteration + 1;
20: end for
```

Refer section A.2 for detailed derivation of above equations.

3.2.2.1 Update Equation: Membership

The update equation for the membership values is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint to the

SUBSUME-BST-WD objective function (3.25)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left(\frac{\sum_{d=1}^D v_{cd}(x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}(x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right)^{\frac{1}{q-1}}}. \quad (3.30)$$

3.2.2.2 Update Equation: Proportions

Minimizing the objective function, J , in (3.25) with respect to \mathbf{p}_{cn} by adding a Lagrange multiplier term to enforce the sum-to-one constraint and applying the Karush-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraints on the proportion values, results in the following update for the proportion values,

$$\mathbf{p}_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd} \right) \quad (3.31)$$

where,

$$\lambda_c = 2 \frac{1 - \mathbf{1}_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd}}{\mathbf{1}_{1 \times M} \left(D \mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \mathbf{1}_{M \times 1}}. \quad (3.32)$$

3.2.2.3 Update Equation: Endmembers

As there are no constraints on endmembers, update equation for endmembers is solved by taking the partial derivative of the SUBSUME-BST-WD objective function in (3.25) with respect to \mathbf{E}_{cd}

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} \mathbf{p}_{cn}^T + \alpha (MI_{M \times M} - \mathbf{1}_{M \times M}) \right)^{-1} \times \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} x_{nd}^T \right). \quad (3.33)$$

3.2.2.4 Update Equation: Weights

The update equation for the weights is found by solving $\frac{\partial J}{\partial v_{cd}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-D constraint to the SUBSUME-BST-WD objective function (3.25)

$$v_{cd} = \frac{D + \sum_{w=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nw} - \mathbf{E}_{cw} \mathbf{p}_{cn})^2}{2\delta_{cw}} \right] - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \sum_{w=1}^D \left[\frac{1}{2\delta_{cw}} \right]}{\left[\sum_{w=1}^D \frac{1}{2\delta_{cw}} \right] 2\delta_{cd}} \quad (3.34)$$

where δ_{cd} is given by (3.27).

3.3 Sparsity Promoting Approaches

The algorithms presented in section 3.2 uses a weight decay term. A major disadvantage of the Gaussian prior is that it does not encourage the estimated parameter values exactly to be zero. Instead, a Gaussian prior often retains several small-valued non-zero parameters [83]

In the context of band weighting, instead of assigning exact zero weights to irrelevant bands, Gaussian priors tends to prefer to assign several small non-zero weights to many irrelevant bands. This problem can be overcome by using a sparsity promoting prior instead.

A sparse vector is defined as a vector with many zero elements and a small number of non-zero elements. The goal is to estimate a weight vector in which irrelevant or redundant components are exactly zero [83].

Sparsity can be achieved using a zero-mean Laplacian prior instead of Gaussian

prior. As desired, the zero mean Laplacian prior promotes weights of exactly zero to the irrelevant bands and significant non-zero weights to relevant bands.

A common sparsity promoting term is shown in (3.35). This is related to the Laplacian by the negative log of the Laplacian distribution with zero mean.

$$E_W = \sum_{j=1}^W |w_j| \quad (3.35)$$

The general form of the objective function with sparsity promoting approach is given by (3.36),

$$J_{SUBSUME-SPT} = \sum_{c=1}^C [(1 - \alpha) \text{RSS} + \alpha \text{SSD} + \delta \text{SPT}] \quad (3.36)$$

$$\text{RSS} = \sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \quad (3.37)$$

$$\text{SSD} = \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \quad (3.38)$$

$$\text{SPT} = \delta_c \sum_{d=1}^D |v_{cd}| \quad (3.39)$$

Taking the exponential of the negative of the (3.36),

$$LSSP = \ln \left[e^{-\frac{1}{2} \left[\sum_{n=1}^N u_{cn} \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right)^T \text{diag}(\mathbf{v}_c^2) \left(\mathbf{X}_n - \sum_{m=1}^M \mathbf{E}_{cm} p_{cnm} \right) \right]} e^{-\sum_{c=1}^C \delta_c \sum_{d=1}^D |v_{cd}|} \right]$$

Figure (3.1) shows a Laplacian curve and the general form of Laplacian distribution

is given in (3.40),

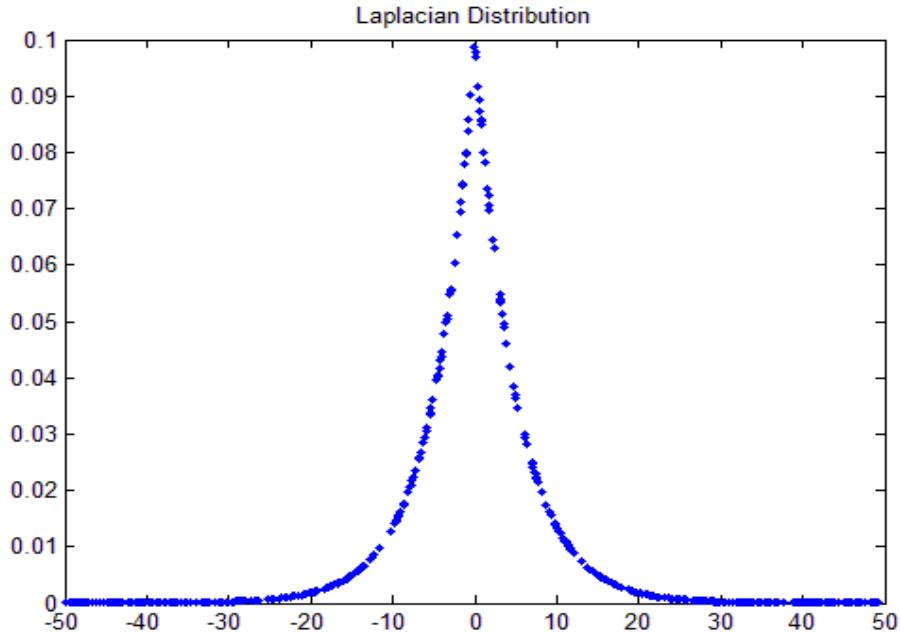


Figure 3.1: Laplacian Distribution

$$f(x|\mu, b) = \frac{1}{2} \exp\left(-\frac{|x-\mu|}{b}\right) \quad (3.40)$$

where μ is mean and b as its variance. From (3.17) and (3.40), it can be observed that the first term of LSSP follows a Gaussian distribution and the second term follows Laplacian distribution with mean 0.

This section will discuss about Sparsity Promoting approach by introducing a Laplacian prior to the objective function of the proposed method.

3.3.1 SUBSUME-SCAD-SP

This section will introduce SUBSUME-SACD-Sparsity Promoting algorithm by adding a Laplacian prior to the weights. Estimate parameters \mathbf{E} , \mathbf{P} , \mathbf{U} , and \mathbf{V} by minimizing the objective function given by (3.41)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{d=1}^D |\mathbf{v}_{cd}| \right) \quad (3.41)$$

subject to the constraints given by (3.42)),

$$p_{cnm} \geq 0; \sum_{m=1}^M p_{cnm} = 1$$

$$u_{cn} \geq 0; \sum_{c=1}^C u_{cn} = 1,$$

$$v_d \geq 0; \sum_{d=1}^D v_d = D \quad (3.42)$$

where u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in endmember set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, δ is the fixed parameter values used to balance the terms of the objective function. As mentioned in the previous section 3.2.1, α is a regularization term to balance the RSS and the SSD term of the objective function.

Algorithm 6 SUBSUME-SCAD Sparsity Promoting Approach

```

1: Set number of clusters ( $C$ ), number of endmembers per cluster ( $M$ ), fuzzifer ( $m$ ),
   threshold for condition ( $threshold$ ),  $\alpha$ ,  $\delta$ , and maximum number of iteration
2: condition = 0;
3: iteration = 1;
4: Initialize endmembers, fuzzy membership, and weights
5: for i =1 to number of clusters do
6:   Initialize proportions
7: end for
8: for j = 1 to maximum number of iteration do
9:   Update proportions  $\mathbf{P}$  using (3.44)
10:  Update endmembers  $\mathbf{E}$  using (3.46)
11:  Update weights  $\mathbf{V}$  using (3.49)
12:  Update fuzzy membership  $\mathbf{U}$  using (3.43)
13:  Estimate stopping criterion-  $Cond^j = \text{norm}(\mathbf{U}^j - \mathbf{U}^{j-1}) + \text{norm}(\mathbf{P}^j - \mathbf{P}^{j-1}) +$ 
       $\text{norm}(\mathbf{E}^j - \mathbf{E}^{j-1})$ 
14:  if ( $abs(Cond^j - Cond^{j-1}) < threshold$ ) then
15:    Break;
16:  end if
17:  iteration = iteration + 1;
18: end for

```

Refer section B.1 for detailed derivation of above equations.

3.3.1.1 Update Equation: Membership

The update equation for the membership values is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint to the SUBSUME-BST-WD objective function (3.41)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left(\frac{\sum_{d=1}^D v_{cd}(x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}(x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right)^{\frac{1}{q-1}}}. \quad (3.43)$$

3.3.1.2 Update Equation: Proportions

Minimizing the objective function, J , in (3.41) with respect to \mathbf{p}_{cn} by adding a Lagrange multiplier term to enforce the sum-to-one constraint and applying the Karush-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraints on the proportion values, results in the following update for the proportion values,

$$p_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd} \right) \quad (3.44)$$

where,

$$\lambda_c = 2 \frac{1 - \mathbf{1}_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd}}{\mathbf{1}_{1 \times M} \left(D u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \mathbf{1}_{M \times 1}}. \quad (3.45)$$

3.3.1.3 Update Equation: Endmembers

As there are no constraints on endmembers, update equation for endmembers is solved by taking the partial derivative of the SUBSUME-BST-WD objective function in (3.41) with respect to \mathbf{E}_{cd}

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} \mathbf{p}_{cn}^T + \alpha (M \mathbf{I}_{M \times M} - \mathbf{1}_{M \times M}) \right)^{-1} \times \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} x_{nd}^T \right). \quad (3.46)$$

3.3.1.4 Update Equation: Weights

Given endmembers and proportions, weights are solved using the Quadratic programming approach. The general form of quadratic programming problem is given

in (B.5),

$$\underset{x}{\min} \frac{1}{2} x^T H x + f^T x \quad (3.47)$$

subject to following constraints:

$$Ax \leq b; \quad A_{eq}x = b_{eq}; \quad lowerLimit \leq x \leq upperLimit \quad (3.48)$$

where H represents the quadratic term in the (3.47), f is the linear term in the (3.47), A is the linear coefficient in the constraint in $A.x \leq b$, b is the constant vector in the constraint in $A.x \leq b$, A_{eq} is the linear coefficient in $A_{eq}.x = b_{eq}$, b_{eq} is the constant vector in the constraint in $A_{eq}.x = b_{eq}$, and *Lower Limit* and *Upper Limit* are the lower bound and upper bound of the constraints given in (3.48).

In order to solve for the weights, the f term of the quadratic programming problem is given in (3.49),

$$f = U(X - E^T P) .* 2 + \delta_c \quad (3.49)$$

MATLAB's *quadprog* function is used to solve the quadratic programming problem mentioned above. The parameter settings for the *quadprog* function are given in (3.50),

$$f = [U(X - E^T P) .* 2 + \delta_c]^T$$

$$H = diag(ones(D, 1))$$

$$lb = zeros(D, 1)$$

$$ub = D * ones(D, 1)$$

$$Aeq = ones(1, D)$$

$$Beq = D \quad (3.50)$$

3.3.2 SUBSUME-BST-SP

This section will introduce SUBSUME-BST-Sparsity Promoting algorithm by introducing Laplacian prior instead of Gaussian prior in SUBSUME-BST-Weight Decay algorithm. SUBSUME-BST-Sparsity Promoting approach estimate \mathbf{E} , \mathbf{P} , \mathbf{U} , and \mathbf{V} by minimizing the objective function given by 3.51

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M \sum_{d=1}^D (e_{cmd} - e_{ckd})^T (e_{cmd} - e_{ckd}) + BST \right) \quad (3.51)$$

subject to the constraints given in (3.42)

$$BST = \sum_{d=1}^D \delta_{cd} |v_{cd}| \quad (3.52)$$

where δ_{cd} is given by (3.53)

$$\delta_{cd} = \frac{\Delta \left(\left(\sum_{m=1}^M \frac{1}{A} \sum_{n=1}^N u_{cn} \mathbf{p}_{cn} (x_{nd} - e_{cmd})^2 \right) + 1 \right)}{\left(\sum_{m=1}^M (e_{cmd} - \mu_{0cd})^2 \right) + 1} \quad (3.53)$$

where Δ is a constant and μ_{0cd} is given by (3.54),

$$\mu_{0cd} = \frac{\sum_{n=1}^N x_{nd} u_{cn}}{\sum_{c=c} u_{cn}} \quad (3.54)$$

Also A is given by (3.55),

$$A = \sum_{n=1}^N u_{cn} \quad (3.55)$$

u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in endmember set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is the vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, and as mentioned in the previous section 3.2.1, α is a regularization term to balance the RSS and the SSD term of the objective function.

3.3.2.1 Update Equation: Membership

The update equation for the membership values is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint to the SUBSUME-BST-WD objective function (3.51)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left[\frac{\sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}^2 (x_{nd} - \mathbf{E}_{kd} \mathbf{p}_{kn})^2} \right]^{\frac{1}{q-1}}}. \quad (3.56)$$

Algorithm 7 SUBSUME-BST Sparsity Promoting Approach

```

1: Set number of clusters ( $C$ ), number of endmembers per cluster ( $M$ ), fuzzifier ( $m$ ),
   threshold for condition ( $threshold$ ),  $\alpha$ ,  $\delta_c$ , and maximum number of iteration
2: condition = 0;
3: iteration = 1;
4: Initialize endmembers, fuzzy membership, and weights
5: for i =1 to number of clusters do
6:   Initialize proportions
7: end for
8: for j = 1 to maximum number of iteration do
9:   Update proportions  $\mathbf{P}$  using (3.57)
10:  Update endmembers  $\mathbf{E}$  using (3.59)
11:  if iteration > BandWtThresh then
12:    Update  $\delta_{cd}$ 
13:    Update weights  $\mathbf{V}$  using (3.60)
14:  end if
15:  Update weights  $\mathbf{V}$  using (3.60)
16:  Update fuzzy membership  $\mathbf{U}$  using (3.56))
17:  Estimate stopping criterion-  $Cond^j = \text{norm}(\mathbf{U}^j - \mathbf{U}^{j-1}) + \text{norm}(\mathbf{P}^j - \mathbf{P}^{j-1}) +$ 
    $\text{norm}(\mathbf{E}^j - \mathbf{E}^{j-1})$ 
18:  if ( $abs(Cond^j - Cond^{j-1}) < threshold$ ) then
19:    Break;
20:  end if
21:  iteration = iteration + 1;
22: end for

```

3.3.2.2 Update Equation: Proportions

Minimizing the objective function, J , in (3.51) with respect to \mathbf{p}_{cn} by adding a Lagrange multiplier term to enforce the sum-to-one constraint and applying the Karush-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraints on the proportion values, results in the following update for the proportion values,

$$\mathbf{p}_{cn}^T = \left(2\mathbf{u}_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(D\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T \mathbf{x}_{nd} \right) \quad (3.57)$$

where

$$\lambda_c = 2 \frac{1 - 1_{1 \times M} \left(\sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd}}{1_{1 \times M} \left(D u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}}. \quad (3.58)$$

3.3.2.3 Update Equation: Endmembers

As there are no constraints on endmembers, update equation for endmembers is solved by taking the partial derivative of the SUBSUME-BST-WD objective function in (3.51) with respect to \mathbf{E}_{cd}

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} + \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T x_{nd} \right). \quad (3.59)$$

3.3.2.4 Update Equation: Weights

Given endmembers and proportions, weights are solved using the Quadratic programming approach.

$$H = \text{diag}(U(X - E^T P)) \quad (3.60)$$

where H represents the quadratic term for the quadratic programming problem. In order to solve the quadratic programming problem, MATLAB's *quadprog* function is used. The parameter initialization settings are shown in (3.61),

$$f = [\delta_d]^T$$

$$H = diag \left(U \left(\mathbf{X} - \mathbf{E}^T \mathbf{P} \right) \right)$$

$$lb = zeros(D, 1)$$

$$ub = D * ones(D, 1)$$

$$Aeq = ones(1, D)$$

$$Beq = D \quad \quad \quad (3.61)$$

Refer section 3.3.1.4 for the description of the terms mentioned above.

Chapter 4

Experimental Results and Analysis

The proposed algorithms are tested on a simulated hyperspectral data set and three real hyperspectral data sets. The proposed methods was also compared with an existing hyperspectral unmixing algorithm for non-convex data set, PCOMMEND [68].

Table 4.1 lists all the needed input parameters in all methods taken for the experiment under consideration.

Table 4.1: Input Parameters: PCOMMEND and SUBSUME

| Sr. no. | Method | Input Parameters |
|---------|---------------------------------|----------------------------------|
| 1 | PCOMMEND | α, C, M , and q |
| 2 | SUBSUME-SCAD-Weight Decay | α, δ, M, C , and q |
| 3 | SUBSUME-BST-Weight Decay | α, Δ, M, C , and q |
| 4 | SUBSUME-SCAD-Sparsity Promoting | α, δ, M, C , and q |
| 5 | SUBSUME-BST-Sparsity Promoting | α, Δ, M, C , and q |

where C is the number of clusters or partitions, M is the number of endmembers per cluster, q is the fuzzifier which determines the degree of fuzziness among clusters,

α is the regularization parameter, Δ is the tunable parameter which measures sparsity among band weights in case of BST term, and δ is the precision parameter in case of weight decay term.

4.1 Simulated Hyperspectral Data

A simulated non-convex hyperspectral data set was generated using the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) spectral library [1]

4.1.1 ASTER Data Set

This data set has two convex regions (Partition 1 and 2) consisting each of three endmembers: Set 1: Granite, Basalt, and Aplite and Set 2: Marble, Limestone, and Shale. The data set consists of 1000 data points, 200 bands, and the endmember spectra is taken in the wavelength range from $12.644 \mu m$ to $6.156 \mu m$. Fig. 4.1 shows the true endmembers in partition 1 and partition 2. Proportions were generated using a uniform Dirichlet distribution with parameter $\alpha = [1, 1, 1]$ and zero mean Gaussian noise was added with an average SNR of 77db respectively. To check the robustness of the proposed methods, 25 different data sets are generated by using the endmembers specified above with randomly drawn proportion values.

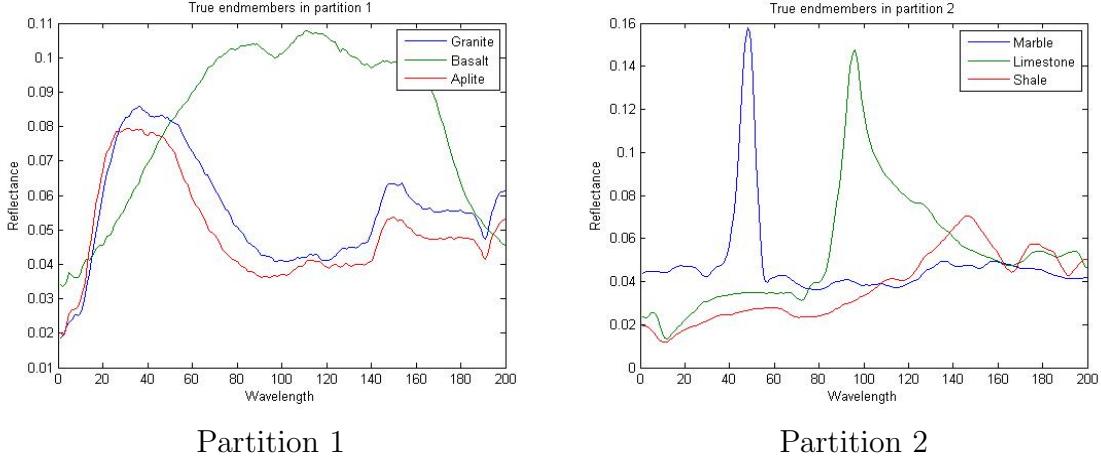


Figure 4.1: True Endmembers in Partition 1 and 2 of Simulated Hyperspectral Data

4.1.1.1 Review- Earth Movers Distance-Based Simultaneous Comparison of Hyperspectral Endmembers and Proportions

Traditionally, Earth Movers Distance (EMD) has been used to compare histograms in application of image processing and computer vision. The EMD is defined as the minimal work done to transform one histogram into the other. EMD between two histograms $h1 = (\mathbf{E}, p)$ and $h2 = (\mathbf{B}, q)$ is given as in (4.1) [84],

$$WORK(E, B, p, q, f) = \sum_{m=1}^M \sum_{n=1}^N d_{mn} f_{mn} \quad (4.1)$$

The task here is to minimize the objective function given in (4.1), by estimating the f (optimal flow), subject to the constraints given in (4.2),

$$f_{mn} \geq 0 \quad 1 \leq m \leq M, \quad 1 \leq n \leq N$$

$$\sum_{n=1}^N f_{mn} \leq p_m \quad 1 \leq m \leq M$$

$$\sum_{n=1}^N f_{mn} \leq q_n \quad 1 \leq n \leq N$$

$$\sum_{m=1}^M \sum_{n=1}^N f_{mn} = \min \left(\sum_{m=1}^M p_m, \sum_{n=1}^N q_n \right) \quad (4.2)$$

where \mathbf{E} and \mathbf{B} are the matrices with dimensions $D \times M$ and $D \times N$, e_m is the center of the m^{th} bin and the m^{th} column in \mathbf{E} . \mathbf{p} and \mathbf{q} are the vector containing the corresponding bin counts, d_{mn} is the ground distance between bin centers e_m and b_n which is the pair-wise distance between the two bin centers, p_m is the count for bin center e_m , and q_n is the count for bin center b_n .

Methods for comparing endmembers have been proposed in literature to evaluate the endmember estimation algorithm and compare endmembers during endmember estimation. Most of these methods uses: Spectral Angle Mapper (SAM) given in (4.3),

$$SAM(\mathbf{x}, \mathbf{y}) = \arccos \left(\frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\|_2^2 \|\mathbf{y}\|_2^2} \right) \quad (4.3)$$

Spectral Information Divergence (SID) given in (4.4),

$$SID(\mathbf{x}, \mathbf{y}) = D(\mathbf{x}||\mathbf{y}) + D(\mathbf{y}||\mathbf{x})$$

where

$$D(\mathbf{x}||\mathbf{y}) = \sum_{i=1}^D x_i \log \left(\frac{x_i}{y_i} \right) \quad (4.4)$$

and Squared Euclidean Distance (SED) given in (4.5),

$$SED(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2^2 \quad (4.5)$$

where \mathbf{x} and \mathbf{y} are two endmember spectra and i represents the value of i^{th} pixel in a hyperspectral image. However, common limitation with these methods arises when one has to manually map two sets of endmembers prior to computing SED, SID, and SAM. In a similar way proportion vectors needs to be sorted before computing the error between two sets of proportion vectors.

These problems are addressed by Zare et al. [84] in the Earth Movers Distance-Based Simultaneous Comparison of Hyperspectral Endmembers and Proportions. The EMD approach presented in this paper compares two sets of endmembers and proportions with varying numbers of endmembers per set. The ground distance between these sets of endmembers and proportions is calculated using SED, SAM, and SID. The objective function to be minimized is given in (4.6),

$$EMD_{tot}(\mathbf{F}^*, \mathbf{E}, \mathbf{B}, \mathbf{P}, \mathbf{Q}) = \sum_{i=1}^{N_x} EMD(\mathbf{f}_i^*, \mathbf{E}, \mathbf{B}, \mathbf{p}_i, \mathbf{q}_i) \quad (4.6)$$

subject to the constraint given in (4.7),

$$\sum_{m=1}^M \sum_{n=1}^N f_{mn} = 1 \quad (4.7)$$

where $EMD(\mathbf{f}_i^*, \mathbf{E}, \mathbf{B}, \mathbf{p}_i, \mathbf{q}_i)$, is the EMD between two proportion vectors with respect to a single data point (N_x) and is given as in (4.8),

$$EMD(\mathbf{f}_i^*, \mathbf{E}, \mathbf{B}, \mathbf{p}_i, \mathbf{q}_i) = \frac{\sum_{m=1}^M \sum_{n=1}^N f_{mn}^* d_{mn}}{\sum_{m=1}^M \sum_{n=1}^N f_{mn}^*} \quad (4.8)$$

where N_x is the total number of pixels in the hyperspectral data, \mathbf{E}, \mathbf{p} is the first endmember and proportion vector, \mathbf{B}, \mathbf{q} is the second endmember and proportion vector, \mathbf{E} and \mathbf{B} is the matrix of estimated endmember, p_i is the proportion value of the i^{th} pixel in proportion vector \mathbf{P} , q_i is the proportion value of the i^{th} pixel in proportion vector \mathbf{Q} .

4.1.2 Results and Analysis- Simulated Hyperspectral Data

This section will discuss about the experiment results and analysis on the ASTER simulated hyperspectral data. Three different experiments are performed: (1) find the best input parameter settings for this data, (2) examine the behavior of α , and (3) random initialization of endmembers, proportions, and memberships. In all the experiments with the simulated hyperspectral data, C and M was set to 2 and 3 for PCOMMEND and SUBSUME.

4.1.2.1 Experiment 1: To find the best input parameter settings for this data

In this experiment PCOMMEND and SUBSUME was run with 20 different sampled values of the parameters over 25 different data sets with randomly generated proportions. Each data set is run for 10 times with one set of sampled values of the algorithm parameter. Due to the Gaussian nature and flexibility of selecting μ and variance, multivariate normal random distribution was used for sampling the parameters: α and δ . From the experiments, Δ was found to be in the range of 1 to 10 for this simulated data set. Endmembers, proportions, and memberships are initialized randomly. For each run of the experiment, EMD with Squared Euclidean Distance (EMD-SED)

and Spectral Angle Mapper (EMD-SAM) is used to evaluate the results [84]. Table 4.2 represents the best parameter setting for PCOMMEND and SUBSUME on this data.

Table 4.2: Best parameter settings for PCOMMEND and SUBSUME methods for simulated ASTER data

| Method | α | δ | Δ | q |
|---------------------------------|------------|----------|----------|-----|
| PCOMMEND | 0.001 | NA | NA | 2 |
| SUBSUME-SCAD-Weight Decay | 0.0016 | 0.0192 | NA | 2 |
| SUBSUME-BST-Weight Decay | 2.2473e-02 | NA | 10 | 2 |
| SUBSUME-SCAD-Sparsity Promoting | 9.0141e-04 | 0.0192 | NA | 2 |
| SUBSUME-BST-Sparsity Promoting | 0.0016 | NA | 10 | 2 |

Table 4.3 presents the mean EMD-SED and mean EMD-SAM run over 25 data sets with ± 1 standard deviation. Table 4.4 presents the mean EMD-SED normalized per dimension and the mean EMD-SAM normalized per dimension run over 25 data sets with ± 1 standard deviation. The smallest error among all of the methods is highlighted with bold in each table.

Table 4.3: Mean EMD-Squared Euclidean Distance (EMD-SED) with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM) with ± 1 standard deviation for 25 data sets

| Method | EMD-SED | EMD-SAM |
|---------------------------------------|--|--|
| PCOMMEND | 21.5104 ± 14.1560 | 80.6661 ± 49.7660 |
| SUBSUME-SCAD-Weight Decay | 11.5265 ± 9.3892 | 40.7463 ± 12.6898 |
| SUBSUME-BST-Weight Decay | 11.6774 ± 8.2133 | 40.4715 ± 15.4463 |
| SUBSUME-SCAD-Sparsity Promoting | 15.2176 ± 1.6296 | 45.5874 ± 6.6680 |
| SUBSUME-BST-Sparsity Promoting | 11.3018 ± 1.8770 | 39.0064 ± 5.2972 |

Experiment 1: Results Discussion

From the quantitative results shown in Tables 4.3 and 4.4 it can be observed that SUBSUME outperforms PCOMMEND. Among the proposed SUBSUME algorithms, SUBSUME-BST-Sparsity Promoting approach outperforms other SUBSUME meth-

Table 4.4: Mean EMD-Squared Euclidean Distance (EMD-SED)/Dimension with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM)/Dimension with ± 1 standard deviation for 25 data sets

| Method | EMD-SED/D | EMD-SAM/D |
|---------------------------------------|---------------------------------------|---------------------------------------|
| PCOMMEND | 0.1076 ± 0.0958 | 0.4033 ± 0.2988 |
| SUBSUME-SCAD-Weight Decay | 0.0963 ± 0.0625 | 0.4169 ± 0.1236 |
| SUBSUME-BST-Weight Decay | 0.0959 ± 0.0709 | 0.3916 ± 0.3748 |
| SUBSUME-SCAD-Sparsity Promoting | 0.0558 ± 0.0072 | 0.3889 ± 0.0418 |
| SUBSUME-BST-Sparsity Promoting | 0.0390 ± 0.0075 | 0.2465 ± 0.0377 |

ods and it has the least EMD-SED and EMD-SAM error.

4.1.2.2 Experiment 2: Examine the behavior of Alpha

As discussed in Section 3.2.1, α in all the SUBSUME methods is a regularization term between $[0, 1]$ used to balance various terms of the objective function.

This experiment will examine the behavior of α in all SUBSUME methods. In this set of experiments, α is varied using 20 different sampled values and keeping all other parameters constant to their best found values from Experiment 1 (Table 4.2). Multivariate normal random distribution is used to sample a . For each run of the experiment the EMD with Squared Euclidean Distance (EMD-SED) and Spectral Angle Mapper (EMD-SAM) is used to evaluate the results [84]. The mean EMD-SED and EMD-SAM with and without normalization with respect to dimension over 20 runs of the experiments is plotted with respect to α for all of the proposed SUBSUME methods.

Experiment 2: Results Discussion

Fig. 4.2, 4.3, 4.4, and 4.5 shows the plot of Alpha vs. EMD-SED, Alpha vs. EMD-SAM, Alpha vs. EMD-SED/Dimension, and Alpha vs. EMD-SAM/Dimension

for SUBSUME-SCAD-Weight Decay, BST-Weight Decay, SCAD-Sparsity Promoting, and BST-Sparsity Promoting methods.

From Fig. 4.2, it can be observed that for SUBSUME-SCAD-Weight Decay approach, the value of α is inversely proportional to the EMD-SED and EMD-SAM i.e. with the increase in the value of α , both EMD-SED and EMD-SAM are decreasing and the optimum value of α is found to be between [0.00015 - 0.0002]. After this range the curve shows a constant trend between α and the EMD-SED and the EMD-SAM values. Also the error bars indicates less variation in the estimated EMD-SED and EMD-SAM.

For SUBSUME-BST-Weight Decay, it can be observed from the Fig. 4.3 that at for certain values, the estimated EMD-SED and EMD-SAM decreases with the increase in the value of α . After attaining its global minima in the range [0.0001 - 0.0005], the EMD-SED and EMD-SAM starts increasing with the increase in the value of α .

With the SUBSUME-SCAD-Sparsity Promoting approach, the plots in Fig. 4.4 shows that, SUBSUME-SCAD-Sparsity Promoting approach has the same trend of α with EMD-SED and EMD-SAM as with SUBSUME-SCAD-Weight Decay approach. It can be observed that α in this case is again inversely proportional to the EMD-SED and EMD-SAM and it attains a global minima after 0.0002

From Fig. 4.5, it can be seen that for SUBSUME-BST-Sparsity Promoting approach the estimated EMD-SED and EMD-SAM slightly decreases with the increase in value of α . Then, in the range of [0.00015 - 0.0003], the EMD-SED and EMD-SAM is constant with an increase in the value of α , followed by sudden increment of the error beyond $\alpha > 0.0003$

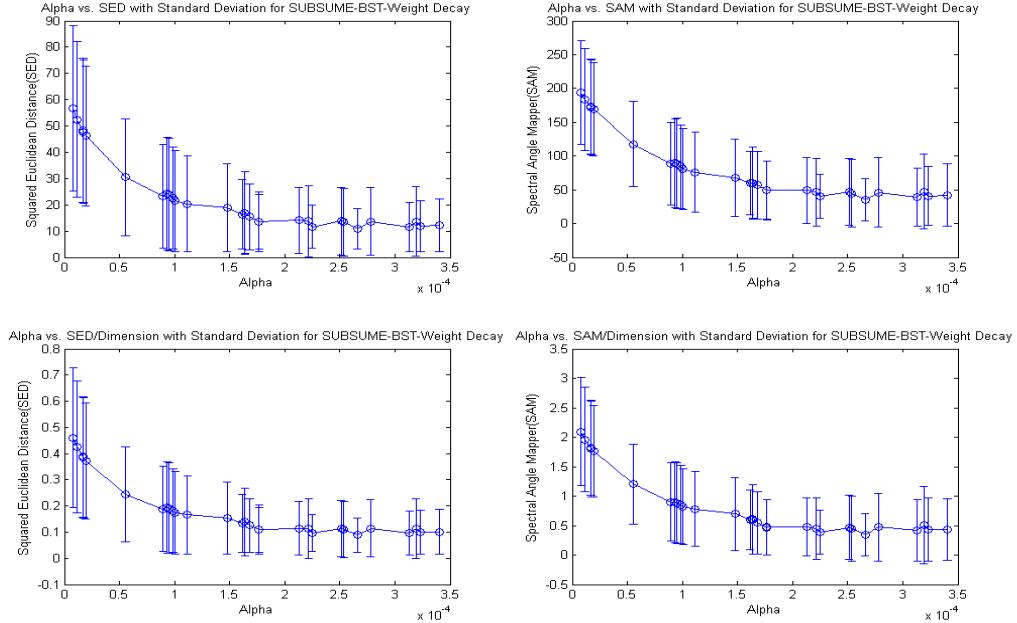


Figure 4.2: Plot of Alpha vs. EMD-SED, Alpha vs. EMD-SAM, Alpha vs. EMD-SED/Dimension, and Alpha vs. EMD-SAM/Dimension for SUBSUME-SCAD-Weight Decay

4.1.2.3 Experiment 3: Random initialization of Endmembers, Proportions, and Memberships

The goal of this experiment set is to examine the behavior of all the SUBSUME algorithms based on their initialization. This experiment set consists of 3 individual set of experiment. In the first set memberships and endmembers are initialized randomly, in the second set memberships and proportions are initialized randomly, and in the third set endmembers and proportions are randomly initialized. The input parameters in all the methods are set from the best parameter settings found from experiment 1 (Table 4.2).

In all the experiment set, mean EMD-SED and mean EMD-SAM with and with-

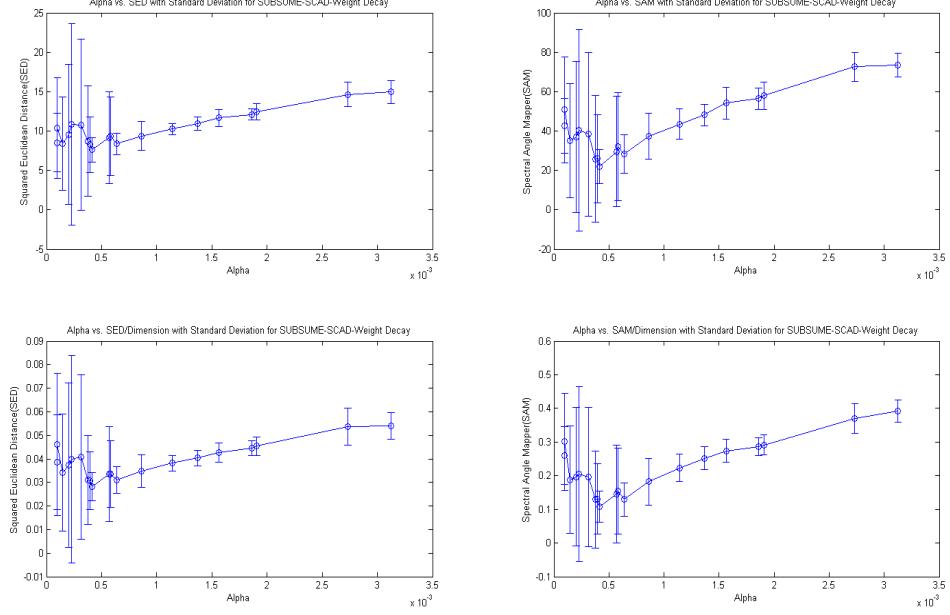


Figure 4.3: Plot of Alpha vs. EMD-SED, Alpha vs. EMD-SAM, Alpha vs. EMD-SED/Dimension, and Alpha vs. EMD-SAM/Dimension for SUBSUME-BST-Weight Decay

out normalization with respect to dimensions over 20 runs of the experiments with ± 1 standard deviation is estimated and the smallest error among all of the SUBSUME methods is highlighted with bold in each table.

Random initialization of Membership and Endmembers

In this set of experiments membership and endmembers are randomly initialized and proportions are initialized using its update equation from all the proposed SUBSUME methods.

Memberships are initialized by a single run of Fuzzy-C Means [69] and endmembers are randomly selected from the input data. For each run of the experiment EMD with

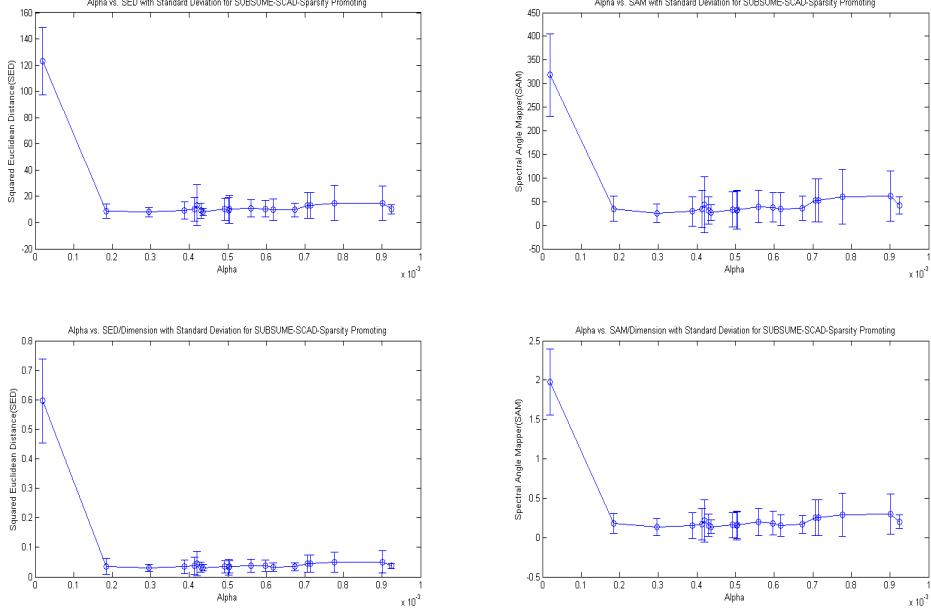


Figure 4.4: Plot of Alpha vs. EMD-SED, Alpha vs. EMD-SAM, Alpha vs. EMD-SED/Dimension, and Alpha vs. EMD-SAM/Dimension for SUBSUME-SCAD-Sparsity Promotion

Squared Euclidean Distance (EMD-SED) and Spectral Angle Mapper (EMD-SAM) is used to evaluate the results [84].

Table 4.5 presents mean EMD-SED and mean EMD-SAM over 20 runs of the experiments with ± 1 standard deviation. Table 4.6 presents mean EMD-SED normalized per dimension and mean EMD-SAM normalized per dimension over 20 runs of the experiments with ± 1 standard deviation. The smallest error among all methods is highlighted with bold in table.

Random initialization of Membership and Proportions

In this set of experiments membership and proportions are randomly initialized

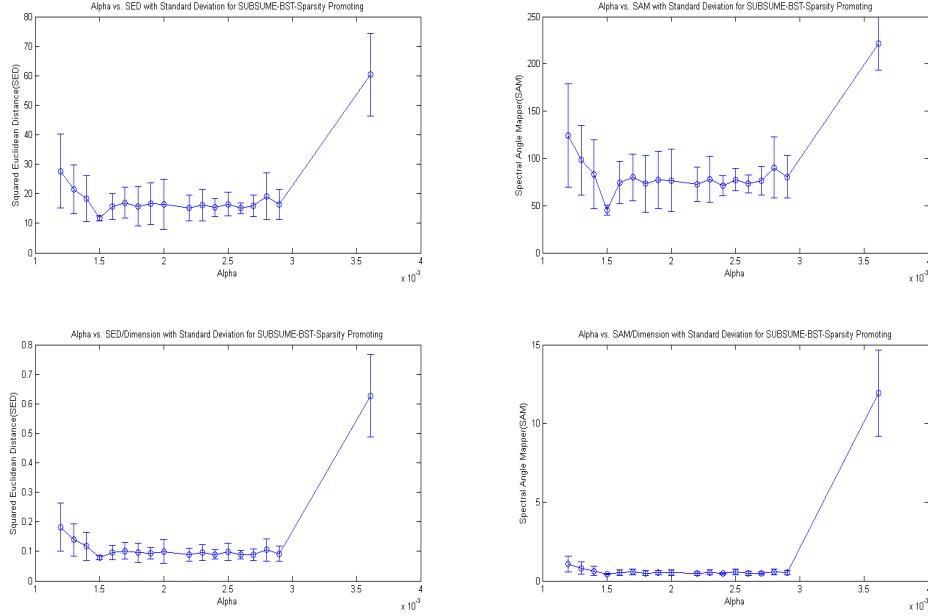


Figure 4.5: Plot of Alpha vs. EMD-SED, Alpha vs. EMD-SAM, Alpha vs. EMD-SED/Dimension, and Alpha vs. EMD-SAM/Dimension for SUBSUME-BST-Sparsity Promotion

and endmembers are initialized using its update equation from all the proposed SUBSUME methods.

Memberships are initialized by a single run of Fuzzy-C Means [69] and proportions are initialized by 1/Number of Endmembers per cluster. For each run of the experiment EMD with Squared Euclidean Distance (EMD-SED) and Spectral Angle Mapper (EMD-SAM) is evaluate the results [84].

Table 4.7 presents mean EMD-SED and mean EMD-SAM over 20 runs of the experiments with ± 1 standard deviation. Table 4.8 presents mean EMD-SED normalized per dimension and mean EMD-SAM normalized per dimension over 20 runs of the experiments with ± 1 standard deviation. The smallest error among all meth-

Table 4.5: Mean EMD-Squared Euclidean Distance (EMD-SED) with ± 1 standard deviation and Mean SAM-Spectral Angle Mapper (EMD-SAM) with ± 1 standard deviation for 20 runs with random initialization of Endmembers and Membership

| Method | EMD-SED | EMD-SAM |
|---------------------------------------|--|--|
| SUBSUME-SCAD-Weight Decay | 11.5236 ± 7.8569 | 42.2370 ± 9.2394 |
| SUBSUME-BST-Weight Decay | 11.2440 ± 5.5421 | 40.8208 ± 10.7736 |
| SUBSUME-SCAD-Sparsity Promoting | 14.8759 ± 1.4278 | 48.8798 ± 5.1002 |
| SUBSUME-BST-Sparsity Promoting | 11.6785 ± 2.6808 | 40.0012 ± 5.2972 |

Table 4.6: Mean EMD-Squared Euclidean Distance (EMD-SED)/Dimension with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM)/Dimension with ± 1 standard deviation for 20 runs with random initialization of Endmembers and Membership

| Method | EMD-SED/D | EMD-SAM/D |
|---------------------------------------|---------------------------------------|---------------------------------------|
| SUBSUME-SCAD-Weight Decay | 0.0986 ± 0.0632 | 0.5375 ± 0.1446 |
| SUBSUME-BST-Weight Decay | 0.1233 ± 0.0701 | 0.5460 ± 0.2316 |
| SUBSUME-SCAD-Sparsity Promoting | 0.0439 ± 0.0055 | 0.2754 ± 0.0268 |
| SUBSUME-BST-Sparsity Promoting | 0.0290 ± 0.0075 | 0.2645 ± 0.0425 |

ods is highlighted with bold in table.

Random initialization of Endmembers and Proportions

In this set of experiments endmembers and proportions are randomly initialized and endmembers are initialized using its update equation from all the proposed SUBSUME methods.

Endmembers are randomly selected from the input data set and proportions are initialized by 1/Number of endmembers per cluster. For each run of the experiment EMD with Squared Euclidean Distance (EMD-SED) and Spectral Angle Mapper (EMD-SAM) is used to evaluate the results [84].

Experiment Set 3: Results Discussion

Table 4.7: Mean EMD-Squared Euclidean Distance (EMD-SED) with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM) with ± 1 standard deviation for 20 runs with random initialization of Membership and Proportions

| Method | EMD-SED | EMD-SAM |
|---------------------------------------|--|--|
| SUBSUME-SCAD-Weight Decay | 11.5231 ± 8.3785 | 42.7142 ± 10.6891 |
| SUBSUME-BST-Weight Decay | 11.3684 ± 7.5905 | 40.4386 ± 12.7991 |
| SUBSUME-SCAD-Sparsity Promoting | 14.1152 ± 1.1123 | 41.7864 ± 7.5252 |
| SUBSUME-BST-Sparsity Promoting | 11.1018 ± 2.6808 | 40.0001 ± 5.6719 |

Table 4.8: Mean EMD-Squared Euclidean Distance (EMD-SED)/Dimension with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM)/Dimension with ± 1 standard deviation for 20 runs with random initialization of Membership and Proportions

| Method | EMD-SED/D | EMD-SAM/D |
|---------------------------------------|---------------------------------------|---------------------------------------|
| SUBSUME-SCAD-Weight Decay | 0.0972 ± 0.0531 | 0.4167 ± 0.1222 |
| SUBSUME-BST-Weight Decay | 0.0956 ± 0.0714 | 0.3038 ± 0.3008 |
| SUBSUME-SCAD-Sparsity Promoting | 0.0584 ± 0.0080 | 0.3882 ± 0.0329 |
| SUBSUME-BST-Sparsity Promoting | 0.0321 ± 0.0079 | 0.2152 ± 0.0212 |

From 4.5, 4.6, 4.7, 4.8, 4.9, and 4.10 it can be observed that all the SUBSUME methods are stable with respect to their initialization. Also SUBSUME-BST-Sparsity Promotion has the least error among the other SUBSUME methods.

Table 4.9 presents mean EMD-SED and mean EMD-SAM over 20 runs of the experiments with ± 1 standard deviation. Table 4.10 presents mean EMD-SED normalized per dimension and mean EMD-SAM normalized per dimension over 20 runs of the experiments with ± 1 standard deviation. The smallest error among all methods is highlighted with bold in table.

Table 4.9: Mean EMD-Squared Euclidean Distance (EMD-SED) with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM) with ± 1 standard deviation for 20 runs with random initialization of Endmembers and Proportions

| Method | EMD-SED | EMD-SAM |
|---------------------------------------|--|--|
| SUBSUME-SCAD-Weight Decay | 11.6744 ± 8.7104 | 43.3152 ± 10.4513 |
| SUBSUME-BST-Weight Decay | 11.5094 ± 7.0982 | 42.0785 ± 11.0365 |
| SUBSUME-SCAD-Sparsity Promoting | 15.2174 ± 1.6241 | 45.2525 ± 6.6684 |
| SUBSUME-BST-Sparsity Promoting | 11.3019 ± 1.7880 | 40.3605 ± 6.2966 |

Table 4.10: Mean EMD-Squared Euclidean Distance (EMD-SED)/Dimension with ± 1 standard deviation and Mean EMD-Spectral Angle Mapper (EMD-SAM)/Dimension with ± 1 standard deviation for 20 runs with random initialization of Endmembers and Proportions

| Method | EMD-SED/D | EMD-SAM/D |
|---------------------------------------|---------------------------------------|---------------------------------------|
| SUBSUME-SCAD-Weight Decay | 0.0971 ± 0.0531 | 0.4120 ± 0.1272 |
| SUBSUME-BST-Weight Decay | 0.0746 ± 0.0508 | 0.3582 ± 0.2316 |
| SUBSUME-SCAD-Sparsity Promoting | 0.0555 ± 0.0062 | 0.3421 ± 0.0418 |
| SUBSUME-BST-Sparsity Promoting | 0.0385 ± 0.0074 | 0.2315 ± 0.0365 |

4.2 Real Hyperspectral Data

Three real hyperspectral data sets are used for getting the qualitative results. These data sets are: Indian Pines Data Set [85], Pavia University Data Set [86], and MUUFL Gulf Port Data Set [87]. Similar to the simulated data experiments, performance of SUBSUME on real hyperspectral data is compared with PCOMMEND.

SUBSUME and PCOMMEND was run with varying number of clusters (C) and endmembers (M). α and δ was sampled using multivariate normal random distribution. Best results with their corresponding parameter settings for each of the method are discussed in this section. For Indian Pines results with ($C=2, M=3$) and ($C=2, M=5$), for Pavia University data set results with ($C=2, M=2$), and for MUUFL Gulf Port data set results with ($C=2, M=2$) are presented.

In all the experiments of with real data discussed here, q the fuzzifier was set to 2,

memberships are initialized using a single run of Fuzzy-C Means algorithm, proportions are initialized to 1/Number of endmembers, endmembers are randomly selected from the data, and an initial weight of 1 is assigned to each band of the input hyperspectral data, following all the respective constraints on all of the algorithms. For the stopping criterion, maximum number of iteration was set to 1000 and threshold (error between values of E , P , U , and V in the consecutive iteration) was set to 0.00001.

Product Maps

For the qualitative results with the PCOMMEND and all the SUBSUME methods, product maps instead of proportion maps are presented. Product maps are the weighted proportion maps corresponding to each partition. The product maps are obtained from the multiplication of proportion of endmember in every pixel with the membership of every pixel in every partition. Mathematically the product maps can be found using the equation given in (4.9),

$$\mathbf{wtdP}_{cn} = \mathbf{p}_{cn} \times u_{cn}^T \quad (4.9)$$

where \mathbf{wtdP}_{cn} , \mathbf{p}_{cn} , and u_{cn} are the weighted proportion, proportion, and membership values corresponding to n^{th} pixel in c^{th} cluster respectively.

Weighted Endmember Plots

For all SUBSUME methods, endmembers in each cluster obtained after unmixing are multiplied with their corresponding weights in each cluster to produce the weighted endmember plots. Mathematically the weighted endmember plots are found using the equation in given in (4.10),

$$\mathbf{wtdE}_c = \mathbf{v}_c^T \times \mathbf{E}_c \quad (4.10)$$

where \mathbf{wtdE}_c are the weighted endmembers in c^{th} cluster, \mathbf{E}_c are the endmembers estimated after unmixing, and \mathbf{v}_c is the weight of each band in c^{th} cluster.

4.2.1 AVIRIS Indian Pines Data

Qualitative analysis of SUBSUME along with its comparison with PCOMMEND is done on the AVIRIS Indian Pines data set acquired on June 1992 which is an agricultural site in northern Indiana. The image scene has 145×145 pixels taken across 200 spectral bands in the wavelength range of $0.4 - 2.5 \mu$ meters. The scene has two third of agriculture area and remaining one-third of forest and other natural potential vegetation. Apart from the vegetation, there are two major highways, a rail line as well as some other man-made structures. Being in their early stages, the soybean and the corn crop in the scene has only 5% of the crop cover [21; 88] and residue from previous crop covers the remaining part of the scene. The amount in which the residue is left from the previous crop is defined by no till, min till, and clean till labels [21; 58]. Fig. 4.6 shows the RGB image, ground truth, and spectra of the Indian Pines data set.

SUBSUME and PCOMMEND was applied on this data with varying number of clusters and endmember. Best results were found with $(C=2, M=2)$, $(C=2, M=5)$ and $q = 2$ along with other corresponding parameter settings. The following sections present the results on Indian Pines data set run on SUBSUME and PCOMMEND.

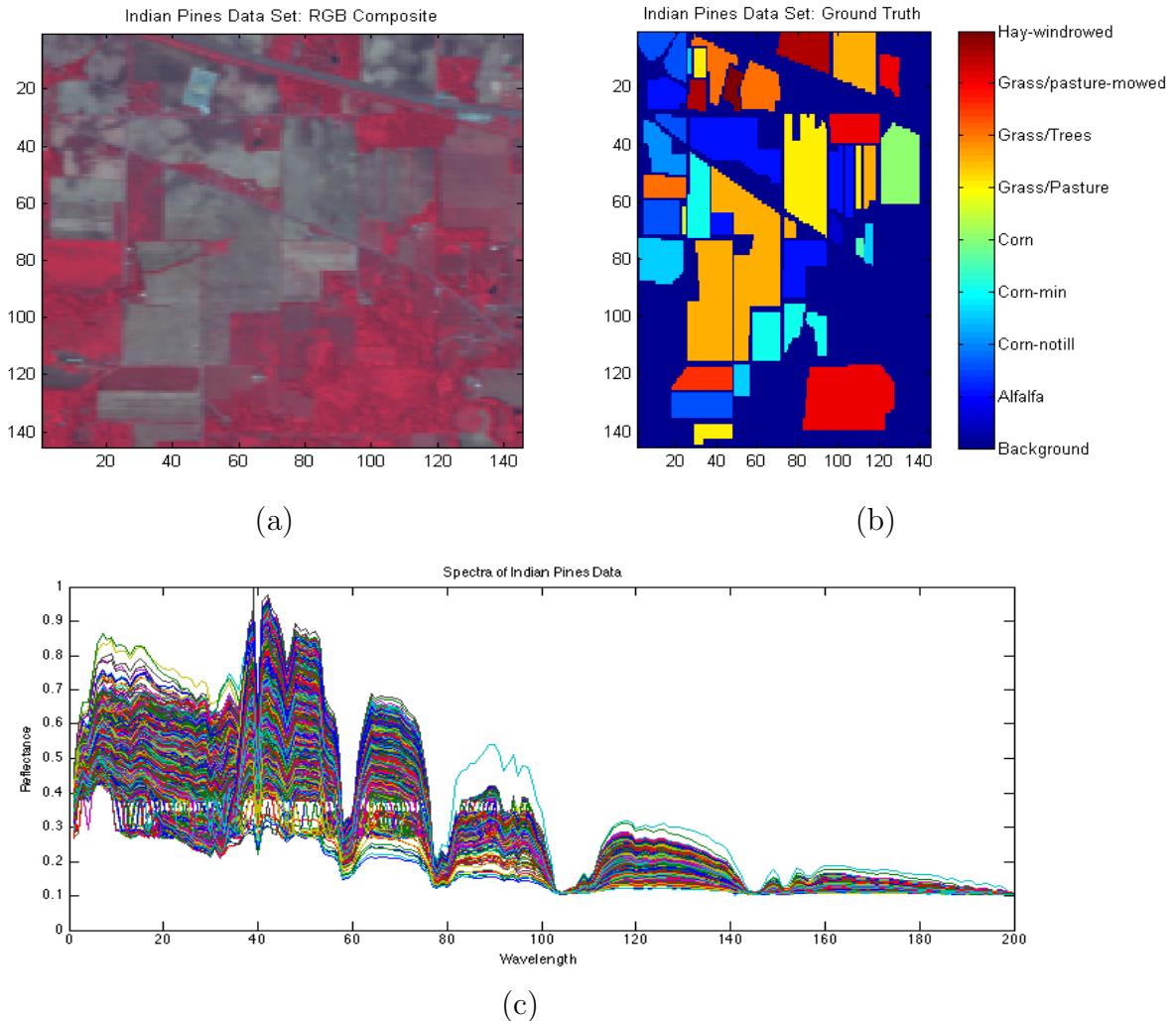


Figure 4.6: Indian Pines (a) RGB composite generated using bands- 47,24, and 14 for Red, Green, and Blue respectively (b) Ground truth showing ground labels (c) Spectra of the data

4.2.1.1 PCOMMEND Results

Two sets of experiments are performed. In the first experiment C and M were set to 2 and 3, and α was set to 0.0114. It can be observed from the Fig. 4.7(a,b) that PCOMMEND identified woods, trees, and wheat in partition 1 and soybean, corn, and hay in partition 2. Also, it can be observed that hay and woods has their product maps washed out as compared to the other product map of the endmembers.

In the second set of experiment, C and M were set to 2 and 5, and α was set to 0.0164. It can be observed from the Fig. 4.8(a,b) that most of the product maps are washed out and does not result in good set of endmembers as compared to the previous set of experiment with ($C=2, M=3$).

Fig. 4.7(c,d) and 4.8(c,d) shows the endmember spectra in partition 1 and partition 2. From this Fig. it can be observed that bands in the range of (30 – 35), (70 – 100), and (130 – 200) has very little information in these bands and, thus, they can be discarded.

4.2.1.2 SUBSUME-SCAD-Weight Decay Results

In the first set of experiments with ($C=2, M=3$) and α and δ as 0.0104 and 0.1000 respectively 4.9(a,b), corn-notill, hay, and stone-steel towers has been identified as the endmembers in partition 1 and woods, grass, and corn-notill has been identified as the endmembers in partition 2. Also, it can be observed that some of the product maps of the endmembers are washed out.

In the second set of experiment C and M were set to 2 and 5, α and δ were set to 0.1034 and 0.0067 respectively. In partition 1, hay, woods, and trees were identified and in partition 2, soybean, corn, steel-stone towers, and woods are identified. As

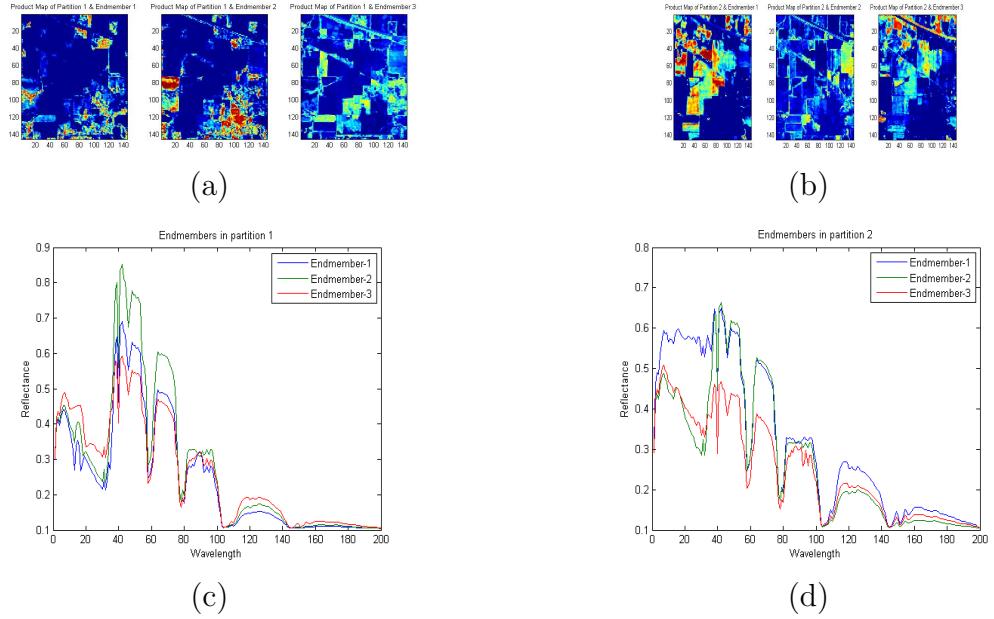


Figure 4.7: Indian Pines- PCOMMEND: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Endmembers in partition 1 (d) and partition 2

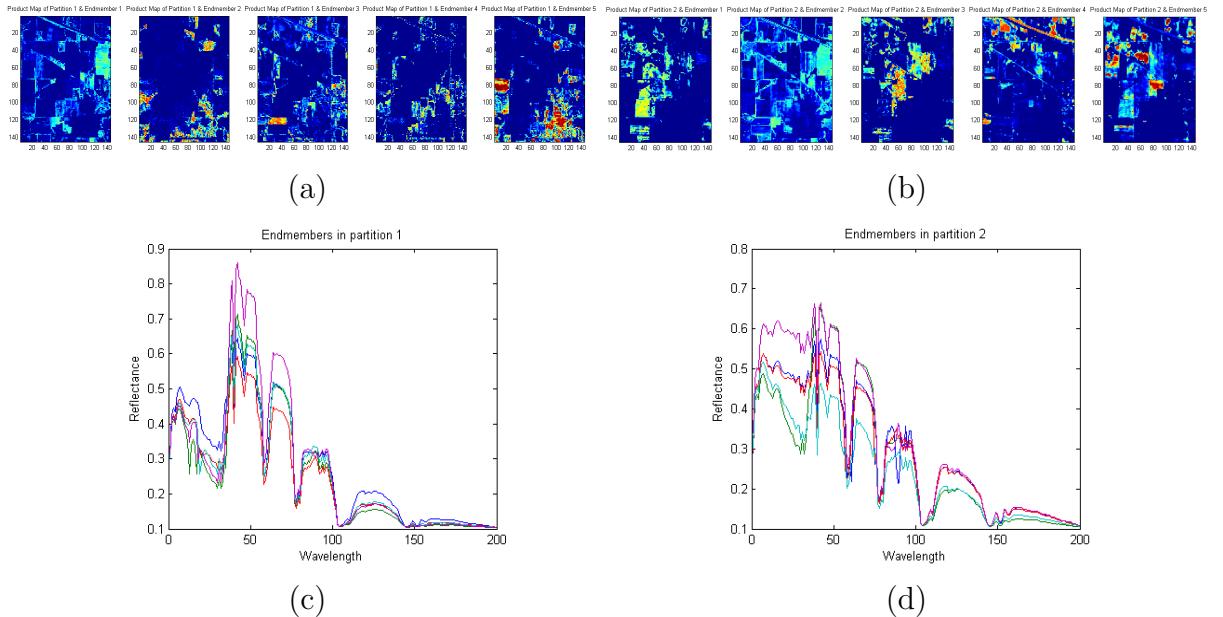


Figure 4.8: PCOMMEND: $C=2$, $M=5$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Endmembers in partition 1 (d) and partition 2

compared to PCOMMEND, SUBSUME-SCAD-Weight Decay results in high intensity of endmembers in each partition.

Fig. 4.9 (c,d) and 4.10 (c,d) shows the weighted endmembers for ($C=2, M=3$) and ($C=2, M=5$). It can be seen that this method has assigned higher weights to the irrelevant bands as compared to the relevant one. Moreover, due to the presence of Gaussian prior as a weight decay term in the objective function, several small non-zero weights instead of exact zero weights has been assigned.

4.2.1.3 SUBSUME-SCAD-Sparsity Promoting Results

In the first experiment C and M were set to 2 and 3, α and δ were set to 0.0200 and 0.0586 respectively. From Fig. 4.11 (a,b) it can be observed that woods, grass/trees, and stone-steel-towers were found in partition 1 and corn-notill, hay-windrowed, and soybean-clean in the partition 2. Also it is obvious from the product maps that the corn-notill is present in very high intensity.

In the second experiment C and M were set to 2 and 5 and α and δ to 0.0104 and 0.0100. From visual inspection of Fig. 4.12 (a,b) it is clear that stone-steel-towers, corn-notill (has washed region), grass/pasture-oats, and woods were found in partition 1, whereas hay, soybean-notill, soybean-clean; and corn-notill were found in partition 2.

As compared to the product maps obtained from PCOMMEND and SUBSUME-SCAD-Weight Decay approach, this method resulted in more effective partitioning of the endmembers in each partition. Also, hay, identified from this method (endmember 2 in partition 2) is present in very high intensity in both the results of ($C=2, M=3$) and ($C=2, M=5$).

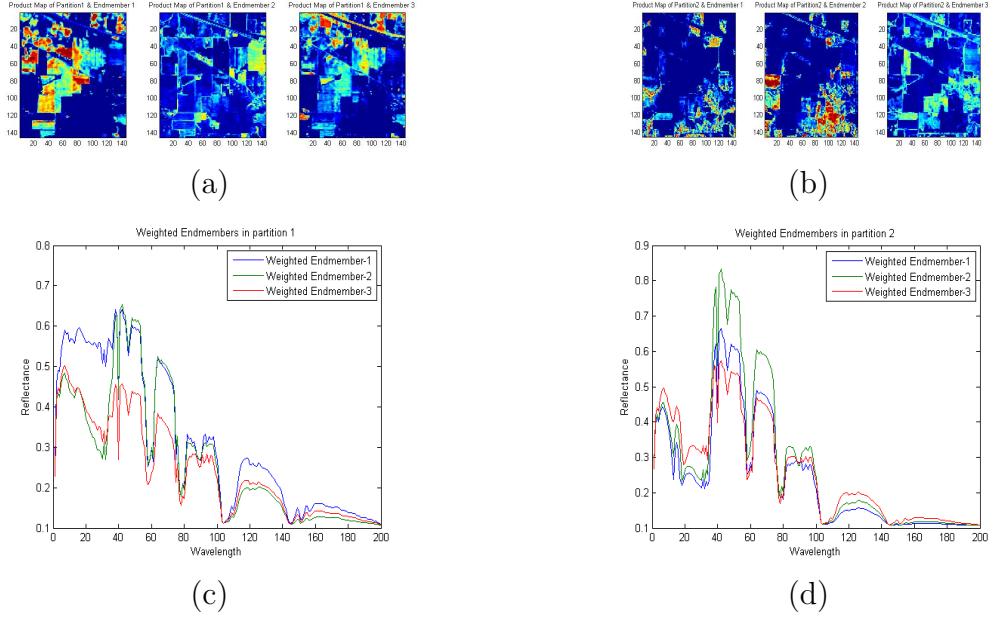


Figure 4.9: Indian Pines- SUBSUME-SCAD-Weight Decay: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmember spectra in partition 1 (d) and partition 2

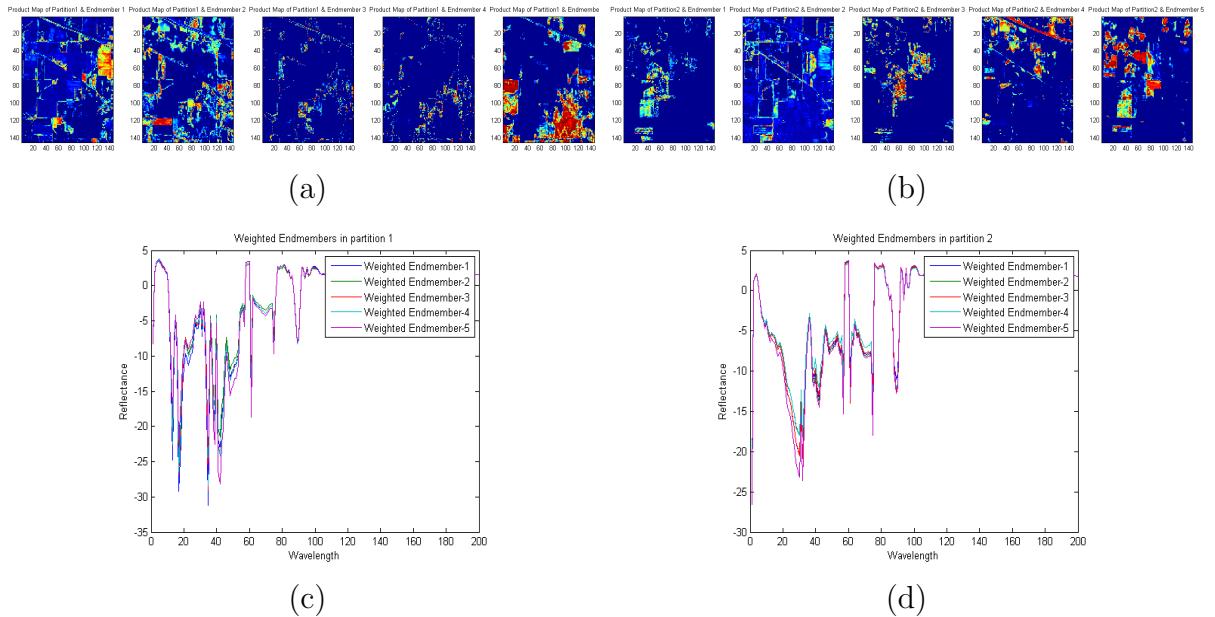


Figure 4.10: Indian Pines- SUBSUME-SCAD-Weight Decay: $C=2$ and $M=5$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmember spectra in partition 1 (d) and partition 2

From the weighted endmembers in both ($C=2, M=3$) Fig. 4.11 (c,d) and ($C=2, M=5$) Fig. 4.12 (c,d) experiments, it can be observed that limitation of having several small non-zeros weights in SUBSUME-SCAD-Weight Decay approach has been overcome by replacing the Gaussian prior with the Laplacian prior in SUBUSME-SCAD-Sparsity Promoting method. However, this method again assigned zero weights to the relevant bands instead of the irrelevant ones.

4.2.1.4 SUBSUME-BST-Weight Decay Results

In the first experiment C and M were set to 2 and 3, α and Δ were set to 0.0167 and 7000 respectively. From the product maps in Fig. 4.13 (a,b) it can be seen that this method identified soybean, hay-windrowed, and man-mande materials- stone-steel-towers in partition 1 and woods, trees-grass, and wheat-oats-trees in partition 2. Also as compared to the endmembers estimated by the PCOMMEND (Fig. (4.7)) this method results in the proportions of high intensity or more darker regions of proportion values. In the second experiment C and M were set to 2 and 5, α and Δ were set to 0.0122 and 5000 respectively.

Fig. 4.13 (a,b) and Fig. 4.14 (a,b) shows the weighted endmembers for experiment with ($C=2, M=3$) and ($C=2, M=5$). Due to the presence of Gaussian prior as the weight decay term, instead of assigning zero weights to the irrelevant bands, this method has assigned several small non-zero weights to all of the bands.

4.2.1.5 SUBSUME-BST-Sparsity Promoting Results

In the first experiment C and M were set to 2 and 3, α and Δ were set to 0.0250 and 3000, and band weighting is performed after 20 iterations. It can be observed

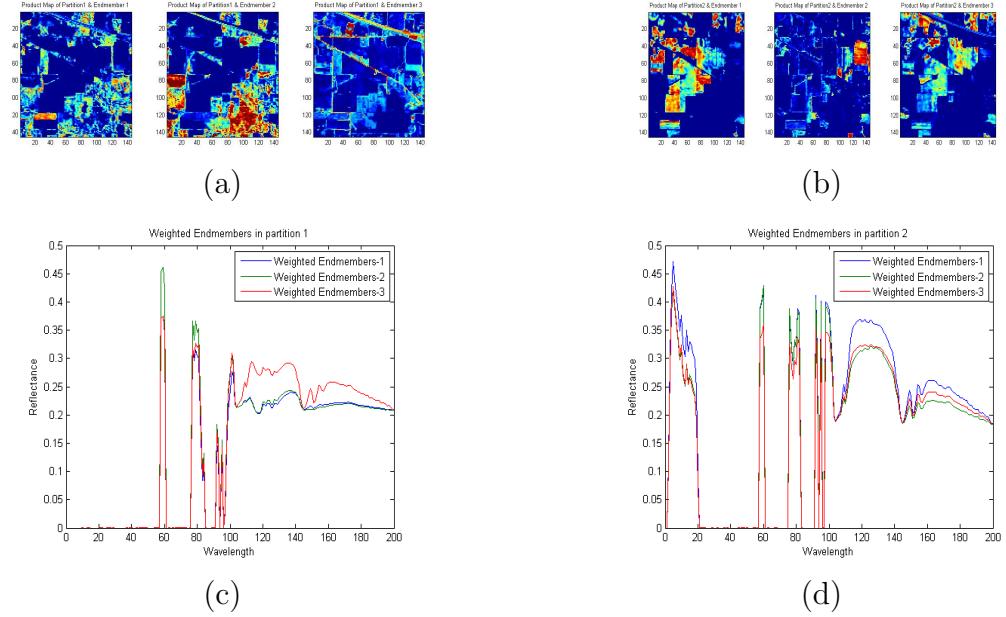


Figure 4.11: Indian Pines- SUBSUME-SCAD-Sparsity Promoting: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

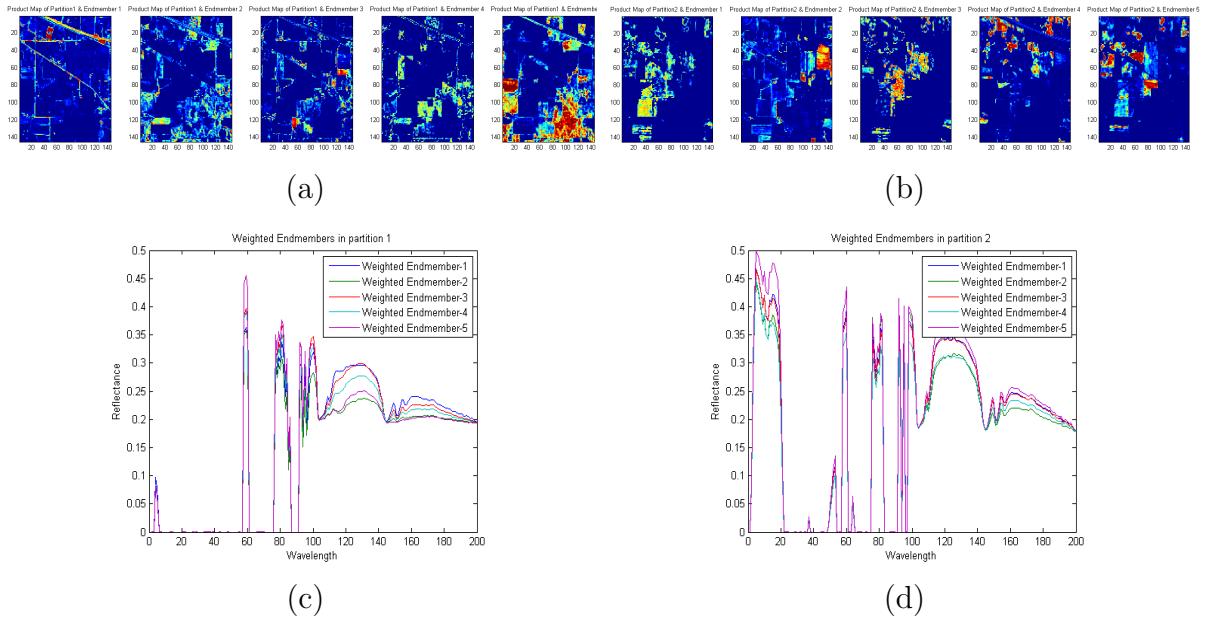


Figure 4.12: SUBSUME-SCAD-Sparsity Promoting: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

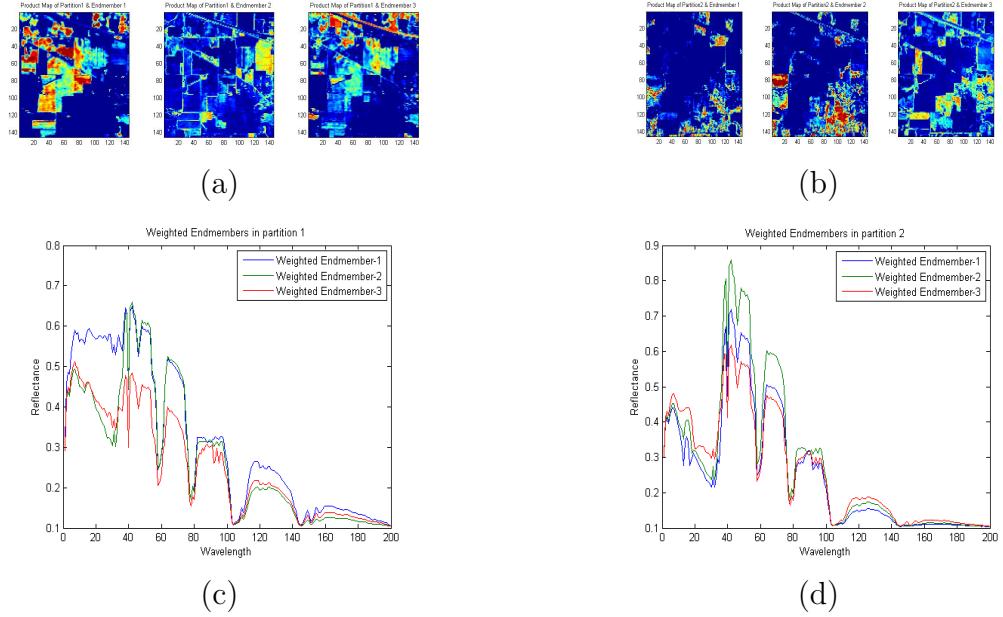


Figure 4.13: Indian Pines- SUBSUME-BST-Weight Decay: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

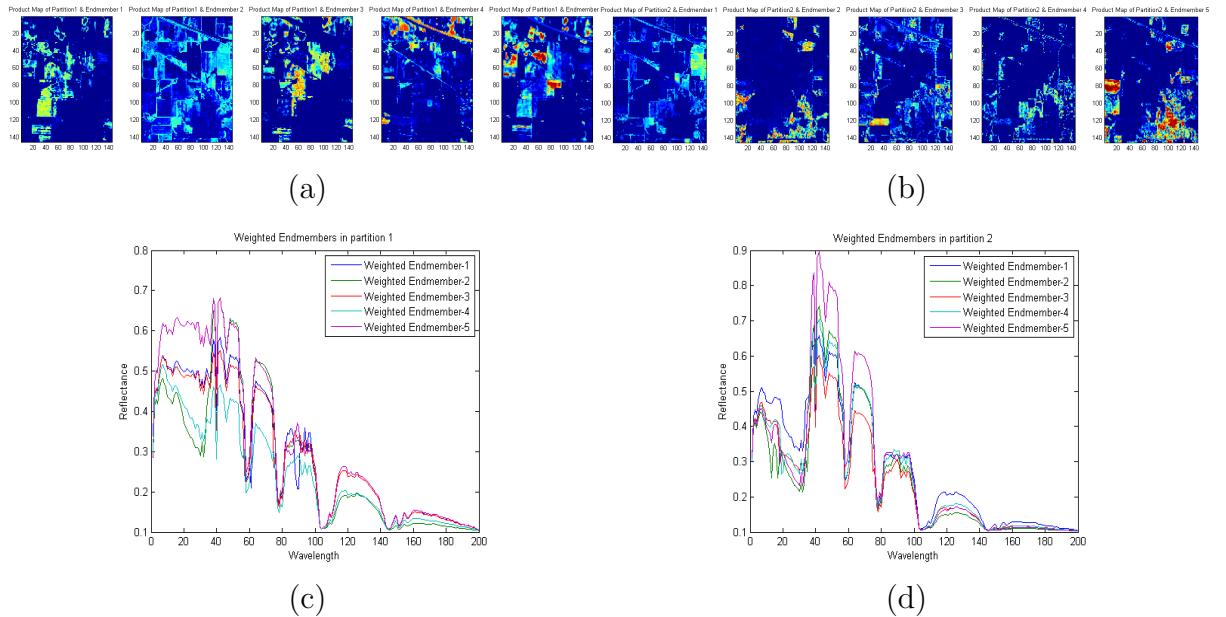


Figure 4.14: Indian Pines- SUBSUME-BST-Weight Decay: $C=2$ and $M=5$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

from the product maps in Fig. 4.15 (a,b) that these product maps has very high intensity of the estimated endmembers. In partition 1 it identified soybean-corn, hay, and stone-steel-towers and in partition 2 it identified woods, grass-trees, and grass/pasture.

In the second experiment C and M were set to 2 and 5, α and Δ were set to 0.0177 and 1000, and band weighting is performed after 20 iterations. It can be observed that the SUBSUME-BST-Sparsity Promoting approach identified corn-notill, wheat mixed with some fraction of grass/trees, grass-trees, woods with a little fraction of hay in partition 1 and in partition 2, stone-steel-towers, soybean, soybean-min and soybean-clean, soybean-notill, and hay has been identified.

Fig. 4.15(c,d) and 4.16(c,d) shows weighted endmembers for both set of experiment. It can be seen from these weighted endmember plots that the SUBSUME-BST-Sparsity Promoting approach results in sparse band weighting. In both the experiments of ($C=2, M=3$) and ($C=2, M=5$), dimensionality of the data was reduced from 200 to 85 and 62 dimensions respectively.

As compared to the PCOMMEND and other SUBSUME methods, SUBSUME-BST-Sparsity Promoting method outperformed all the methods. This method resulted in effective partitioning of the input hyperspectral scene. Moreover presence of BST term and Laplacian term in the objective function encouraged higher weights to the relevant band and exact zero weights to the irrelevant bands to the input hyperspectral data and which significantly reduced the dimensionality of the data.

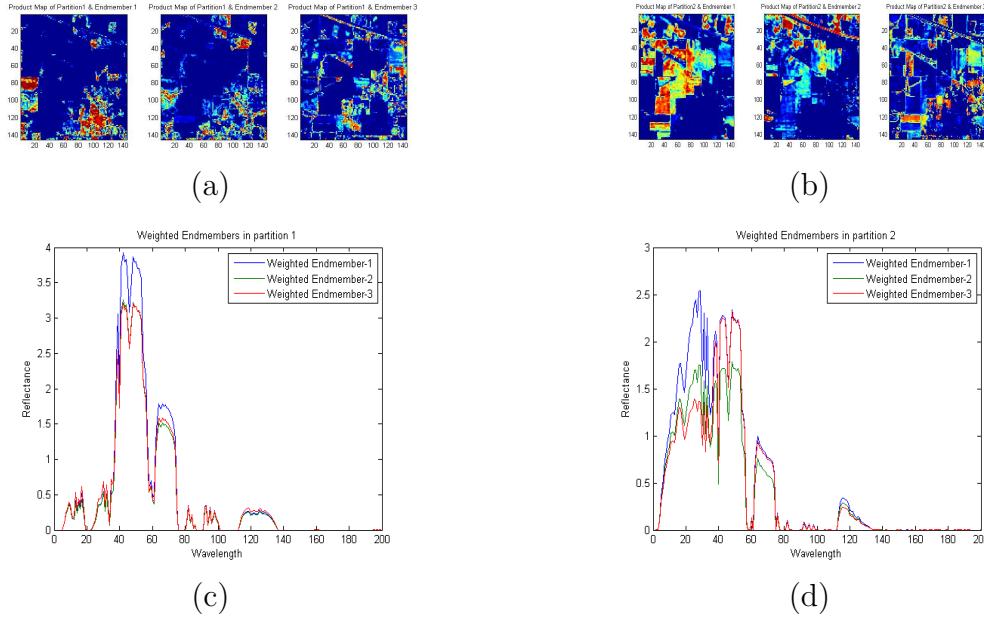


Figure 4.15: Indian Pines- SUBSUME-BST-Sparsity Promoting: $C=2$ and $M=3$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

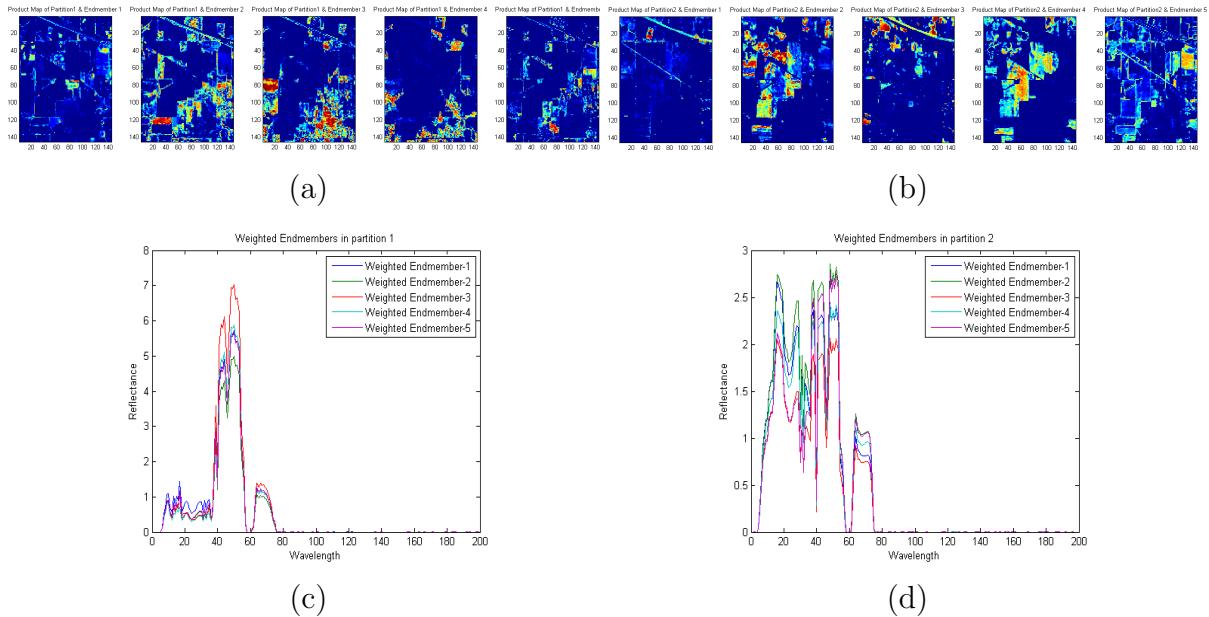


Figure 4.16: Indian Pines- SUBSUME-BST-Sparsity Promoting: $C=2$ and $M=5$ (a) Product maps of endmembers in partition 1 (b) partition 2. (c) Weighted endmembers in partition 1 (d) and partition 2

4.2.2 ROSIS Pavia University Data

The Pavia university data was acquired by ROSIS sensor on July 08, 2002. The hyperspectral image scene has 610×340 pixels across 103 spectral bands. The data was collected over the wavelength range of 0.43 to $0.85 \mu m$ and contains several materials including trees, asphalt, bricks, bitumen, tile, shadow, meadow, and soil [86]. This image contains both natural and urban regions. Two subsets of the data was created and experiments with PCOMMEND and SUBSUME were performed on these subsets.

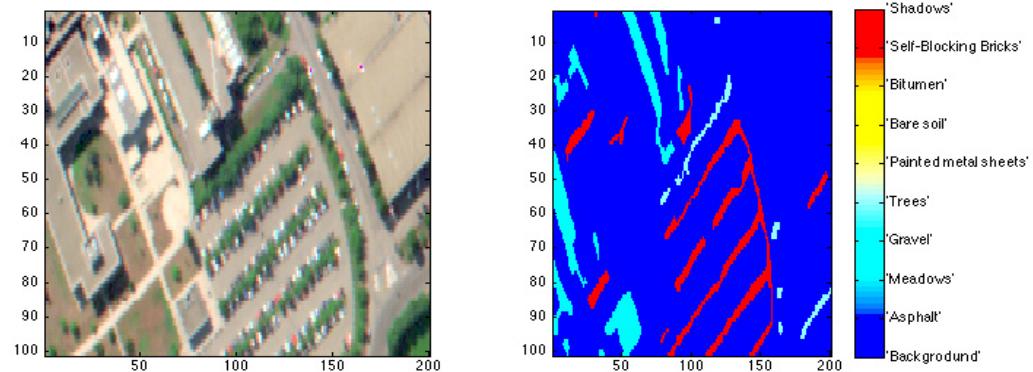
Results obtained on PCOMMEND and SUBSUME methods using Pavia data are discussed in the following sections.

4.2.2.1 Subset-1

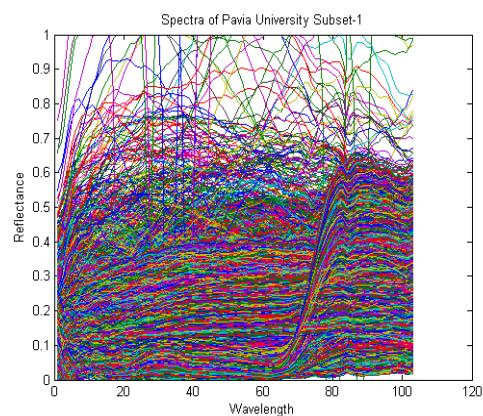
Fig. 4.17 shows the RGB composite of the subset-1 of the Pavia University data, its ground truth, and the its spectra. This subset of the image scene has 101×200 pixels taken across 103 spectral bands in the wavelength range of 0.43 to $0.85 \mu m$ meters. The RGB composite is generated using bands: 60, 55, and 13 for Red, Green, and Blue wavelengths respectively. This subset consists of shadows, trees, asphalt, and cement/sidewalk. SUBSUME and PCOMMEND was applied on this subset with varying number of clusters and endmembers. Best results were found with ($C=2, M=2$) and $q = 2$ along with other corresponding parameter settings.

4.2.2.1.1 PCOMMEND Results

In the this experiment α was set to 0.0050. It can be observed from the Fig. 4.18 that the endmembers in partition 1 are asphalt and cement/sidewalk and in partition



(a)



(b)

Figure 4.17: Pavia University Subset-1 (a) RGB composite generated using bands: 60,55, and 13 for Red, Green, and Blue respectively, Ground truth showing ground labels (b) Spectra of the data

2 are trees and shadows. It is clear that, in partition 1 and endmember 1 it has mixed bare soil with the asphalt. So, from these product maps it is clear that PCOMMEND did not result in an effective partitioning of the data.

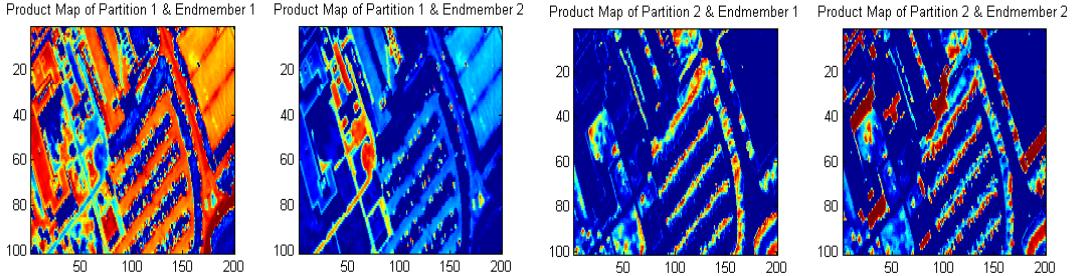


Figure 4.18: Pavia S1- PCOMMEND: Product maps of endmembers in partition 1 and partition 2

4.2.2.1.2 SUBSUME-SCAD-Weight Decay Results

In this experiment, α and δ were set to 0.0108 and 0.2125 respectively. From product maps in Fig. 4.19(a), it can be seen that it has shadows and trees with bare soil as endmember 1 and 2 in partition 1 and cement/sidewalk and asphalt as endmember 1 and 2 in partition 2 respectively. As compared to the PCOMMEND, SUBSUME-SCAD-Weight Decay resulted in effective partitioning of the data.

Fig. 4.19 (b,c) shows the weighted endmembers in partition 1 and 2. It can be seen that instead of reducing the dimensionality of the data, this method resulted in assigning several small non-zero weights to all of the bands due to the presence of Gaussian prior as a weight decay term in the objective function.

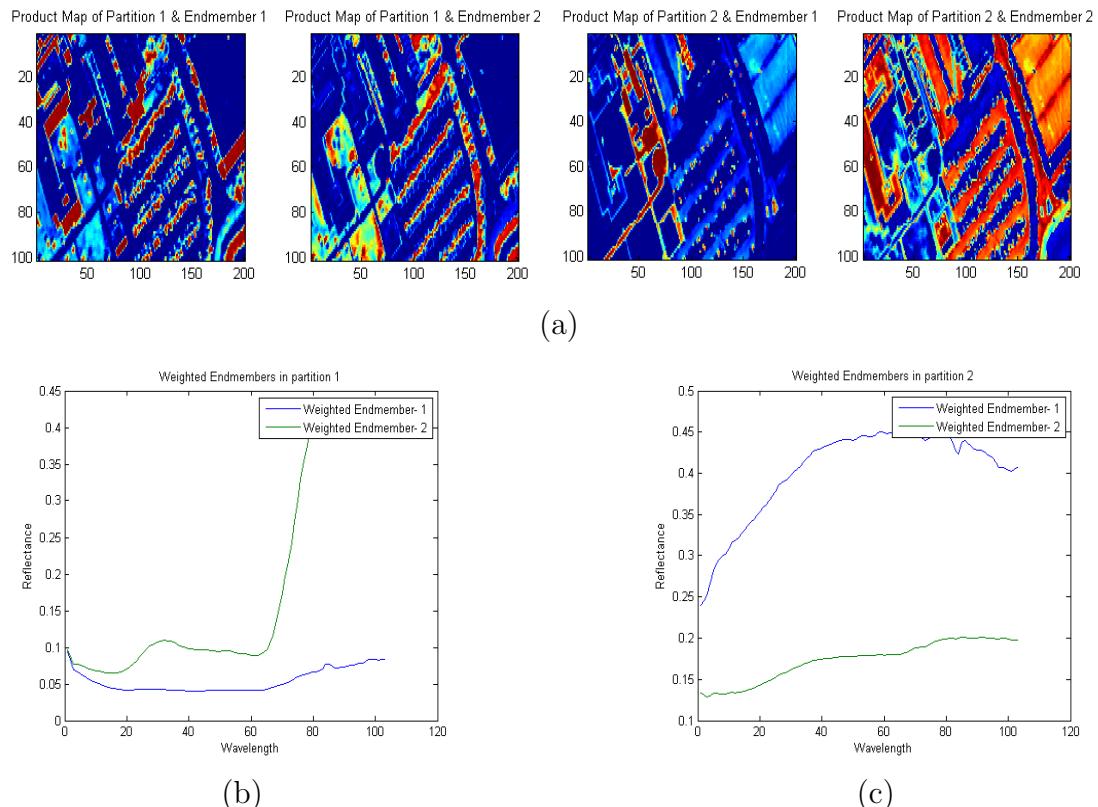


Figure 4.19: Pavia S1- SUBSUME-SCAD-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.2.1.3 SUBSUME-SCAD-Sparsity Promoting Results

In this experiment the input parameters, α and δ were set to 0.0048 and 0.3289 respectively. Fig. 4.20 (a) shows the product map of endmember 1 and 2 in partition 1 and 2. It has trees and shadows in partition 1 and asphalt and cement/sidewalk in partition 2. As compared to the PCOMMEND and SUBSUME-SCAD-Weight Decay approach, this method resulted in the more darker endmembers in each partition.

Fig. 4.20 (b,c) shows the weighted endmembers in partition 1 and 2 respectively. It can be observed that this method resulted in the sparse band weighting due to the presence of Laplacian prior in the objective function. Dimensionality of the data was reduced from 103 to 66 dimensions from this method.

4.2.2.1.4 SUBSUME-BST-Weight Decay Results

In this experiment the input parameters, α and Δ were set to 0.0499 and 5000 respectively. From the product maps in Fig. 4.21 (a) it can be seen that this method detected trees and bare soil and shadows as endmember 1 and 2 in partition 1 and on the other hand cement/sidewalk and asphalt as endmember 1 and 2 in partition 2. Unlike the product maps obtained from PCOMMEND and SUBSUME-BST-Sparsity Promotion, product map of vegetation (endmember 1 in partition 1) from this method has high intensity of bare soil with trees which makes these product maps better than PCOMMEND and SUBSUME-BST-Sparsity Promotion approach.

But, from Fig. 4.21 (b,c) it is clear that this method does not prove to be good in assigning band weights. Fig. 4.21 (b,c) shows the weighted endmembers in partition 1 and 2 respectively. Following the nature of Gaussian prior, there are several small non-zero weights instead of zero weights.

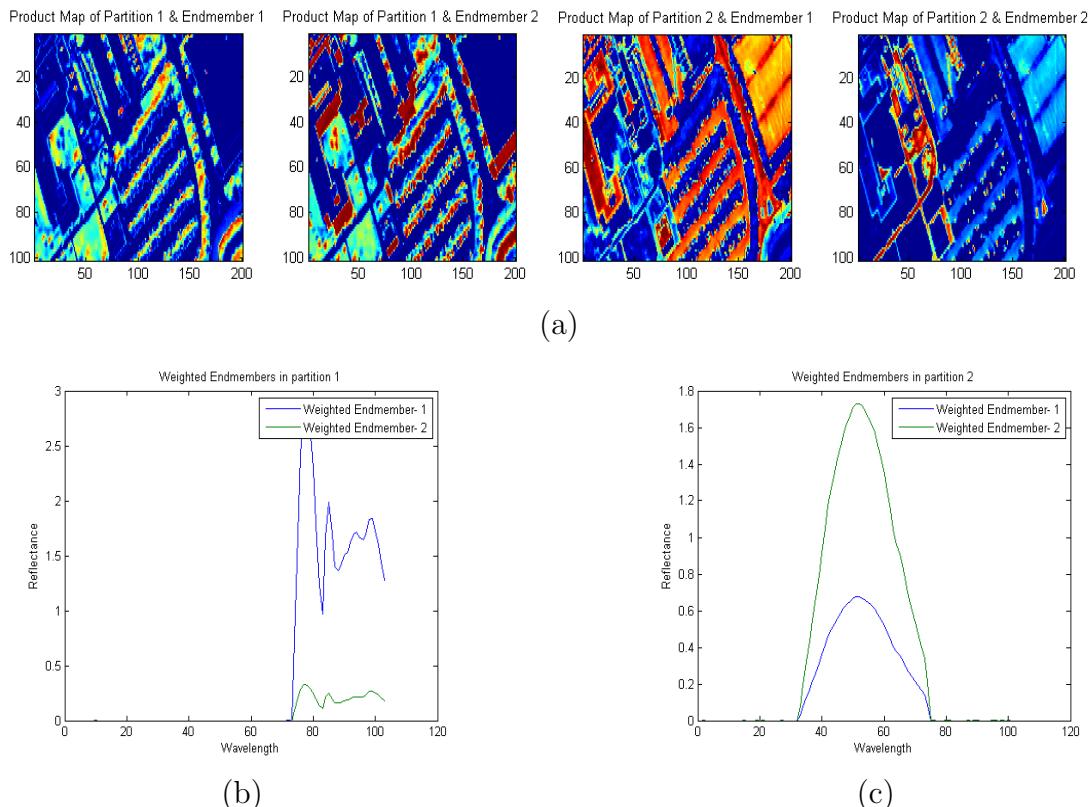


Figure 4.20: Pavia S1- SUBSUME-SCAD-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

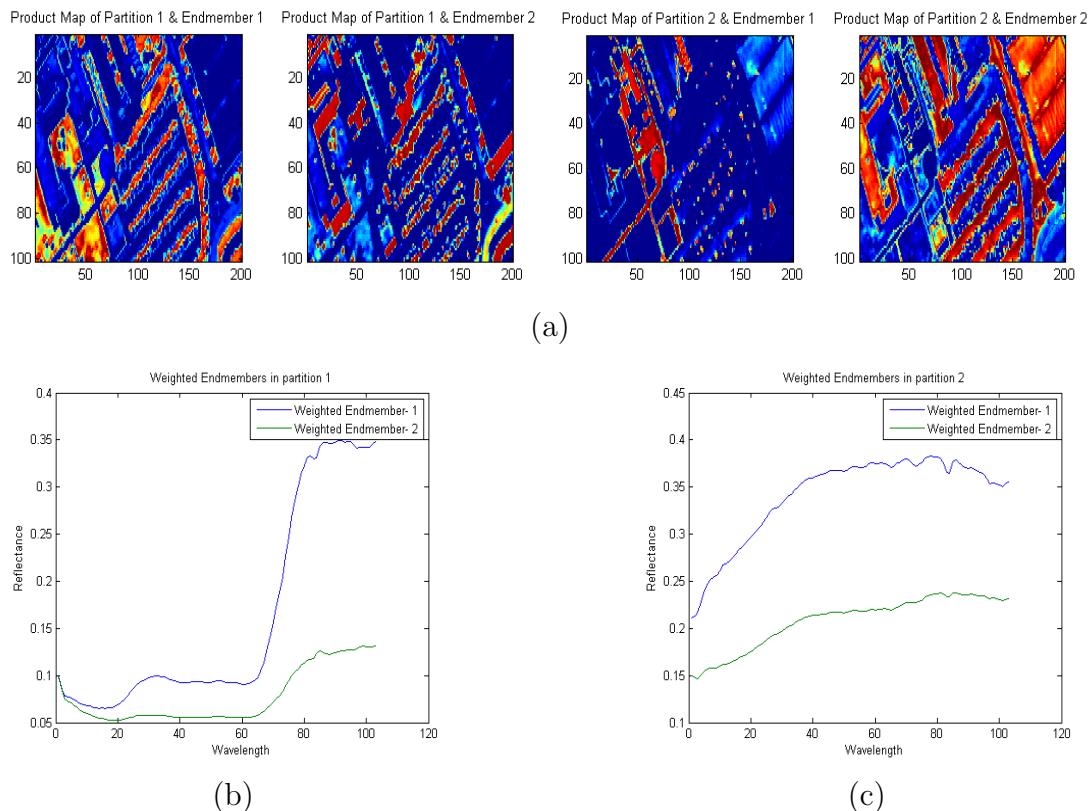


Figure 4.21: Pavia S1- SUBSUME-BST-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.2.1.5 SUBSUME-BST-Sparsity Promoting Results

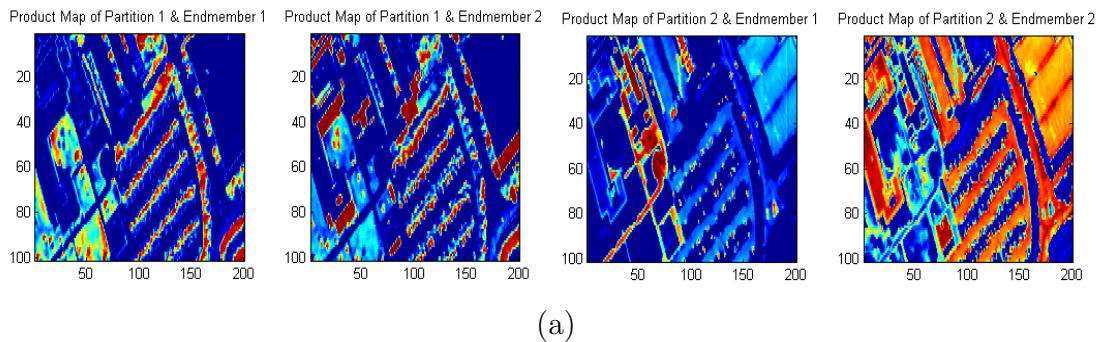
In this experiment α and Δ were set to 0.0012 and 5000, and band weighting is performed after 20 iterations. It can be observed from the product maps in Fig. 4.22 (a) that these product maps has very high intensity of the proportions. Trees and shadows was found as endmember 1 and 2 in partition 1 and cement/sidewalk and asphalt was found as endmember 1 and 2 in partition 2. As compared to the product maps of PCOMMEND, it has clear separation between natural and man-made endmembers.

Fig. 4.22(b,c) shows weighted endmembers in partition 1 and 2 respectively. It can be seen from these weighted endmember plots that the SUBSUME-BST-Sparsity Promoting approach results in sparse band weighting. After band weighting, this method retained 47 bands thus reducing the dimensionality of the data from 103 to 47 bands.

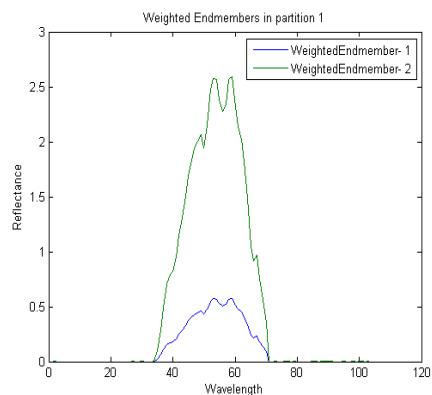
Also it can be observed that SUBSUME-BST-Sparsity Promotion resulted in more significant weights in each partition as compared to the SUBSUME-SCAD-Sparsity Promotion approach.

4.2.2.2 Subset-2

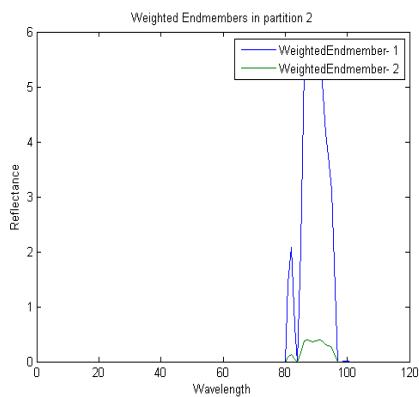
Fig. 4.23 shows the RGB composite of the subset-2 of the Pavia University data set, its ground truth and its spectra. This subset of the image scene has 101×200 pixels taken across 103 spectral bands in the wavelength range of 0.43 to 0.85μ meters. The RGB composite is generated using bands: 60, 55, and 13 for red, green, and blue respectively. This subset consists of shadows, bitumen, bare soil, cement, and asphalt. SUBSUME and PCOMMEND was applied on this subset with varying



(a)



(b)



(c)

Figure 4.22: Pavia S1- SUBSUME-BST-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

number of clusters and endmembers. Best results were found with the the parameter setting of ($C=2, M=2$) and $q = 2$ along with the other corresponding parameter settings.

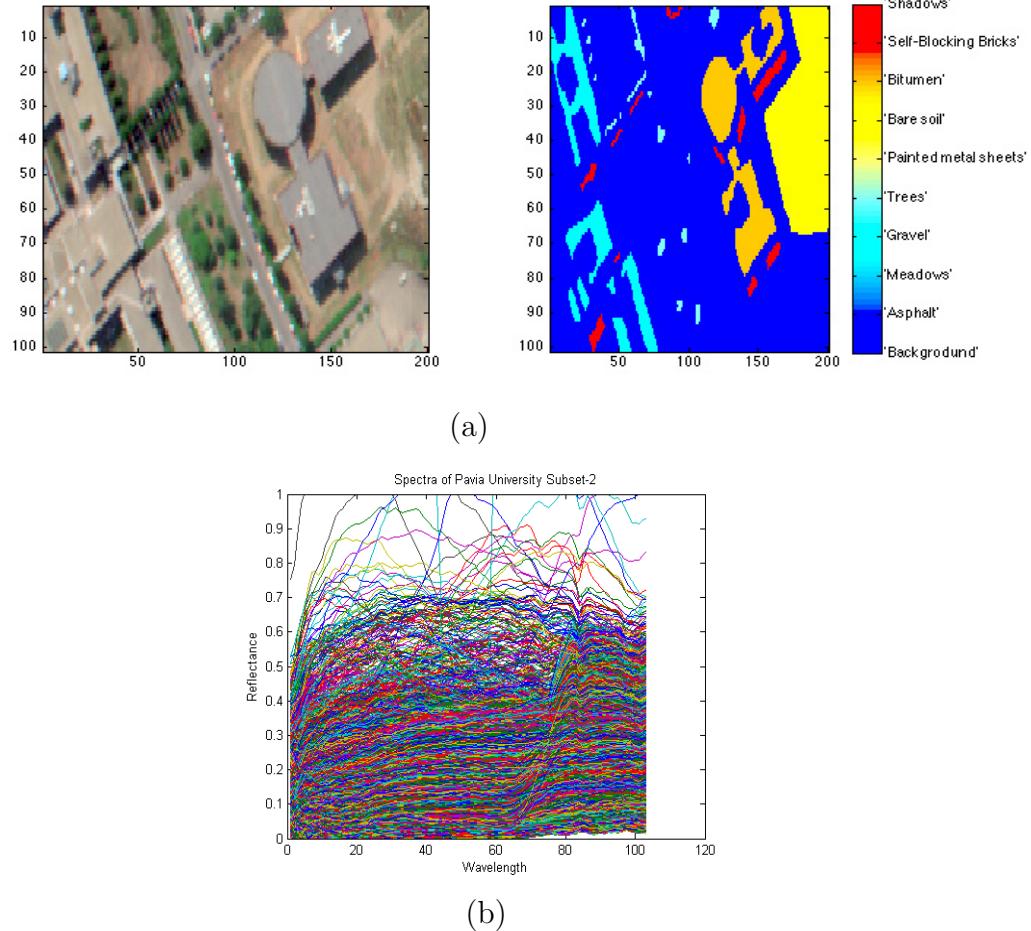


Figure 4.23: Pavia University Subset- 2 (a) RGB composite generated using bands: 60,55, and 13 for Red, Green, and Blue respectively, Ground truth showing ground labels (b) Spectra of the data

4.2.2.2.1 PCOMMEND Results

In this experiment, the input parameter α was set to 0.0011. It can be observed

from the Fig. 4.24 (a) that PCOMMEND identifies bare soil and trees in partition 1 and bitumen-asphalt and cement in partition 2. It is clear from these product maps that, it has very little proportion of cement as in partition 2, also, this method mixed natural material- shadows with man-made material- asphalt and bitumen. From these product maps it is clear that PCOMMEND did not partition the endmembers effectively.

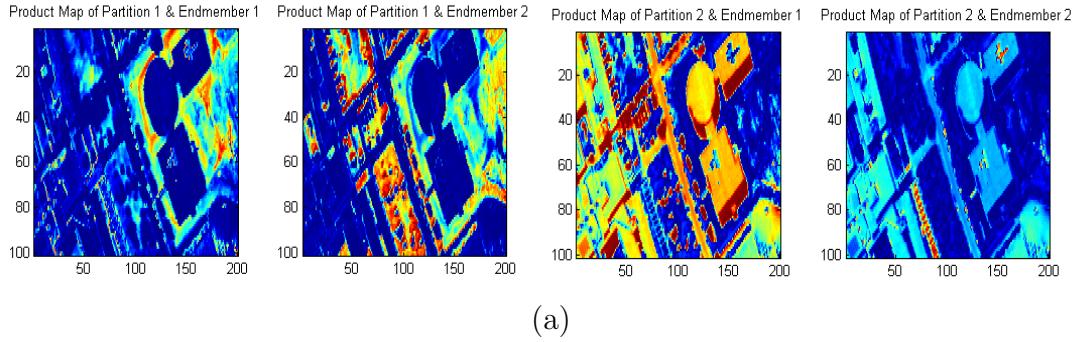


Figure 4.24: Pavia S2- PCOMMEND: Product maps of endmembers in partition 1 and partition 2

4.2.2.2.2 SUBSUME-SCAD-Weight Decay Results

In this experiment α and δ as 0.0368 and 0.2002 respectively. From Fig. 4.25 (a), it can be observed that in partition 1, bitumen-asphalt and cement/side walks (man-made materials) are identified as endmember 1 and 2 and in partition 2 trees and bare soil (vegetation) and shadows are identified as endmember 1 and 2 respectively. Also, as compared to the product maps obtained from PCOMMEND, SUBSUME-BST-WD resulted in very darker product maps in each partition.

Also, from Fig. 4.25 (b,c) it can be observed that due to the presence of Gaussian prior as weight decay term, SUBSUME-BST-WD assigned several small non-zero

weights to all the bands resulting in the all of the 103 dimensions.

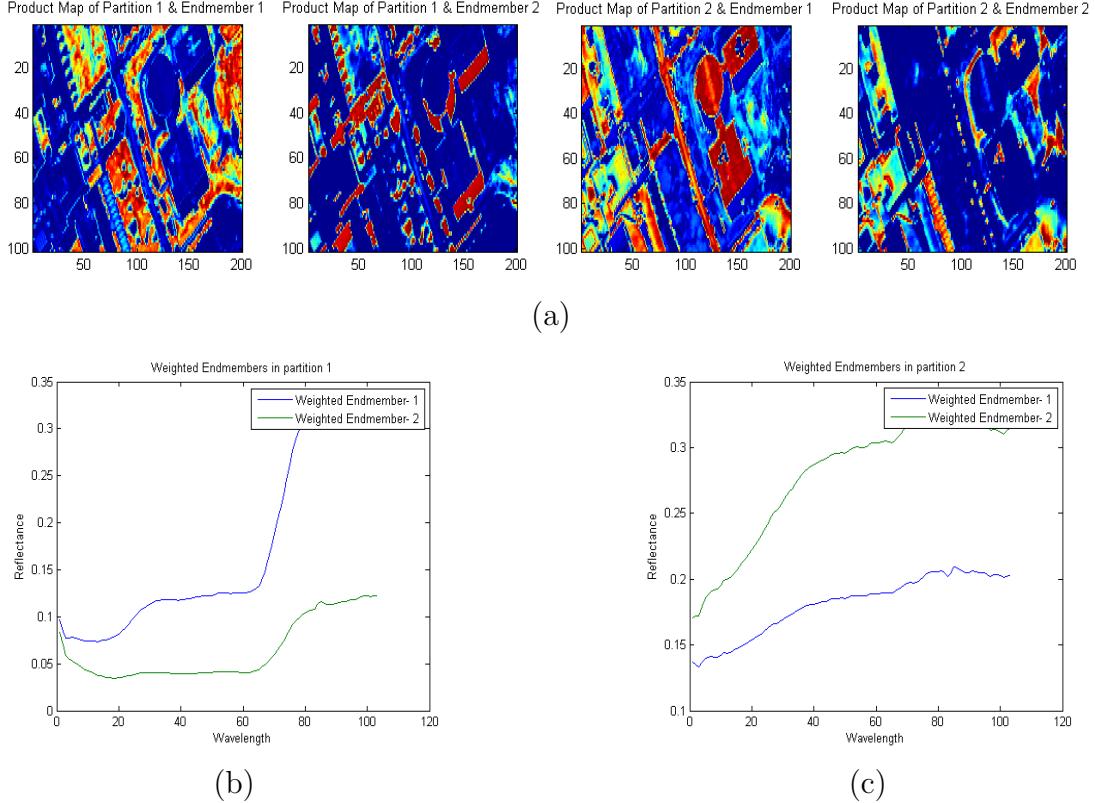


Figure 4.25: Pavia S2- SUBSUME-SCAD-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.2.2.3 SUBSUME-SCAD-Sparsity Promoting Results

In this experiment α and δ were set to 0.00064 and 0.8278 respectively. From Fig. 4.26 (a) it can be observed that proportions of some materials are washed out as compared to the results of the other SUBSUME methods. In partition 1, shadows and trees are identified as endmember 1 and 2 and in partition 2, asphalt and bitumen and cement is identified as endmember 1 and 2. Also it can be observed that the

product map for cement is very light in intensity leaving behind very little proportion of cement.

From the weighted endmembers in Fig. 4.26 (b,c) it can be seen that the due to the presence of Laplacian prior in the objective function, dimensionality of the data is reduced from 103 to 40.

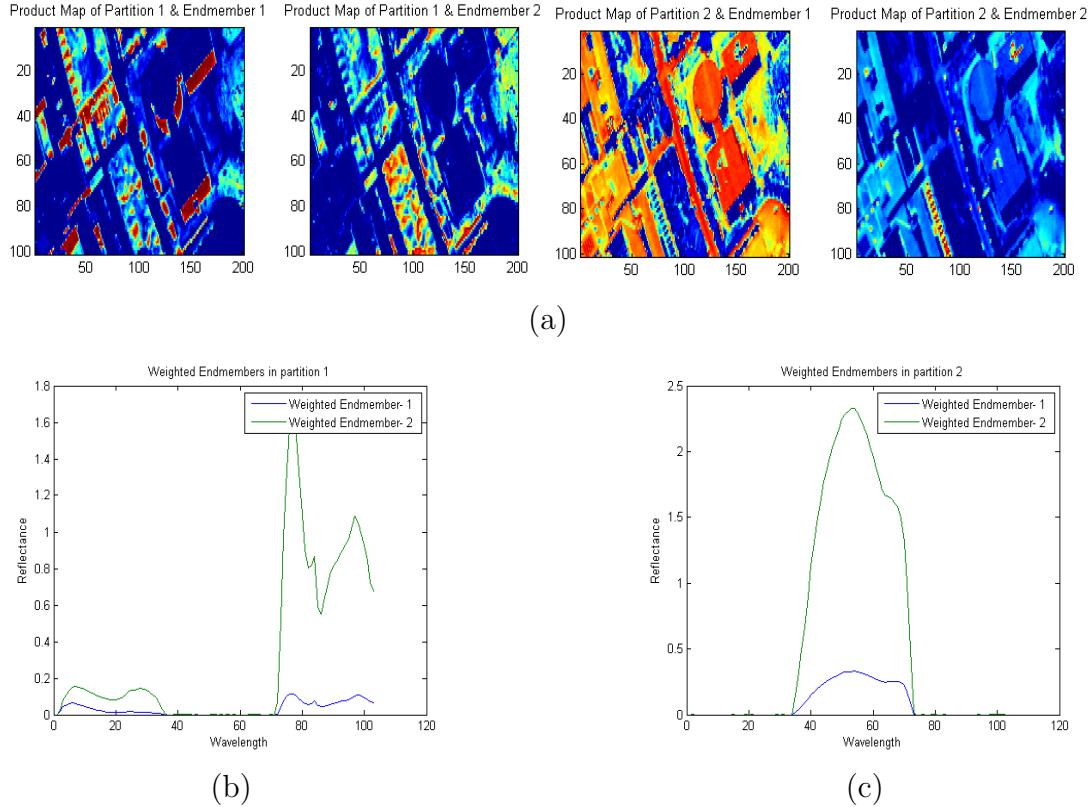


Figure 4.26: Pavia S2- SUBSUME-SCAD-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.2.2.4 SUBSUME-BST-Weight Decay Results

For this experiment the parameters α and Δ were set to 0.0478 and 5000 respectively.

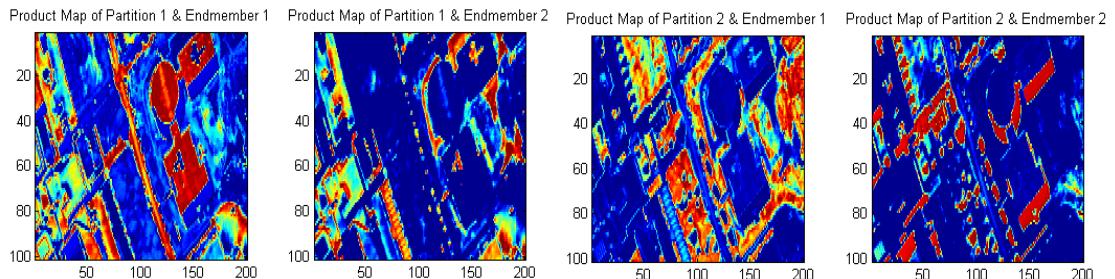
tively. It can be observed from the product maps in Fig. 4.28 (a) that, SUBSUME-BST-WD has identified bitumen-asphalt and cement/side walks (man-made endmembers) as endmember 1 and 2 and in partition 2 trees and bare soil (vegetation endmembers) and shadows as endmember 1 and 2 respectively. The product maps with this method shows very high intensity of the presence of above mentioned materials in both partitions.

But, from Fig. 4.13 (b,c) it is clear that this method has resulted in poor band weighting. Due to the presence of Gaussian prior in the objective function, there are several small non-zero weights assigned to each band of the data.

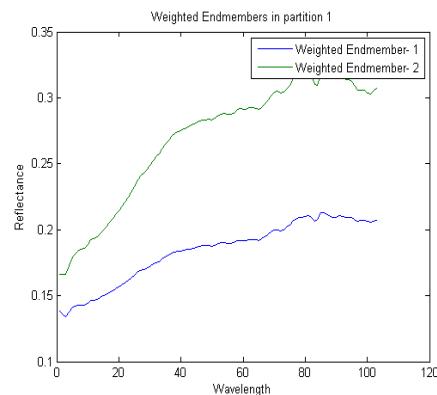
4.2.2.2.5 SUBSUME-BST-Sparsity Promoting Results

In this experiment the parameters α and Δ were set to 0.0907 and 5000 and band weighting is performed after 20 iterations. It can be observed from the product maps in Fig. 4.28 (a) that, in partition 1, it SUBSUME-BST-SP has identified trees and bare soil (vegetation endmembers) and shadows as endmember 1 and 2 respectively. In partition 2, SUBSUME-BST-SP has identified bitumen-asphalt and cement/side walks (man-made endmembers) as endmember 1 and 2. As compared to the PCOMMNED results and other SUBSUME methods, these product maps shows an effective separation of endmembers present in the scene corresponding to each partition.

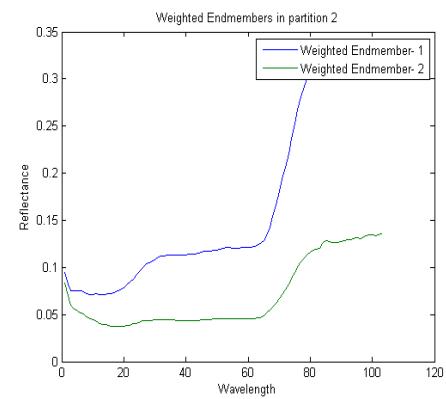
Fig.4.28(b,c) and Fig.4.28(b,c) shows weighted endmembers for both set of experiment. It can be seen from these weighted endmember plots that the SUBSUME-BST-Sparsity Promotinsg approach results in sparse band weighting. From this method dimensionality of the data was reduced from 103 to 62 dimension.



(a)



(b)



(c)

Figure 4.27: Pavia S2- SUBSUME-BST-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

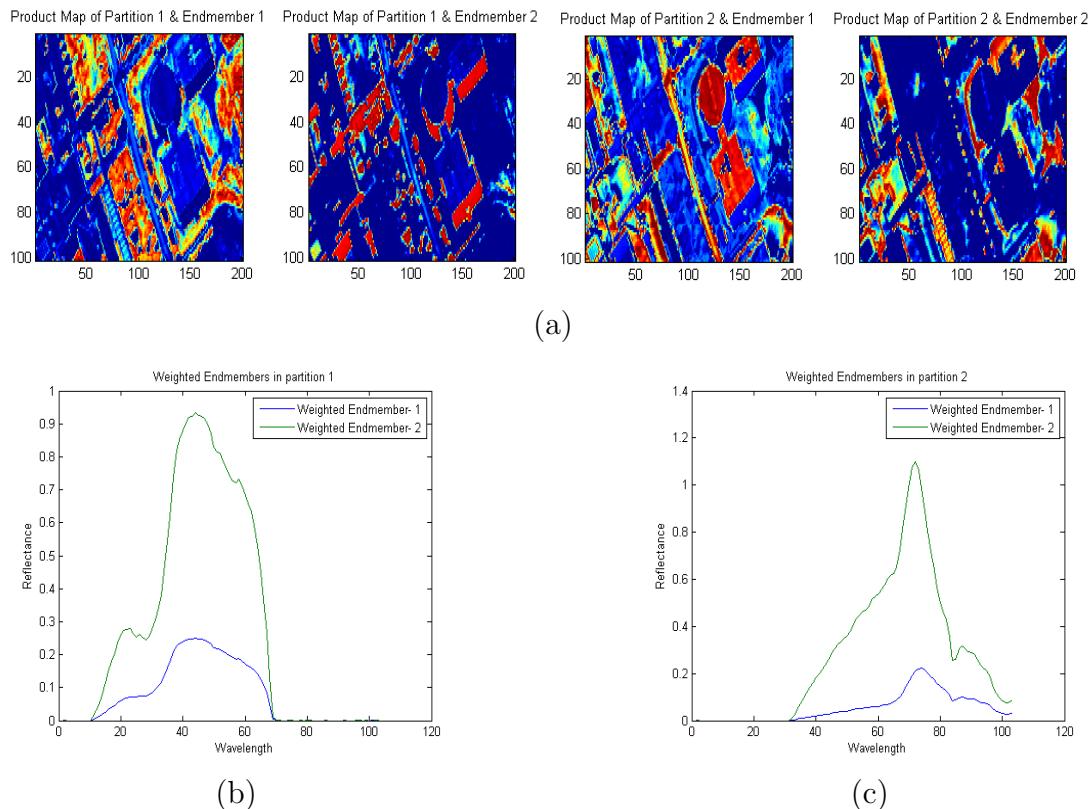


Figure 4.28: Pavia S2- SUBSUME-BST-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.3 MUUFL Gulf Port Data

MUUFL data has been acquired using a Gemini LiDAR and CASI-1500 flown in a single plane simultaneously, thereby provided co- registered hyperspectral and LiDAR data [87]. The data is acquired over the campus of the University of Southern Mississippi Gulfport, located in Long Beach, Mississippi (close to Gulfport, MS). The data mainly consists of grass, trees, shadows, roofs, side walks, and other man-made materials. In the experiments described in this section, subset of the data is taken. This subset of has 250×350 pixels taken across 72 spectral bands. Fig. 4.29 shows the RGB composite of the subset of the data and its spectra. From the RGB composite it is clear that the data has very high percentage of vegetation (trees and grass).

SUBSUME and PCOMMEND was applied on this data with varying number of clusters and endmembers. Best results were found with the parameter setting of ($C=2, M=2$) and $q = 2$ along with other corresponding parameter settings.

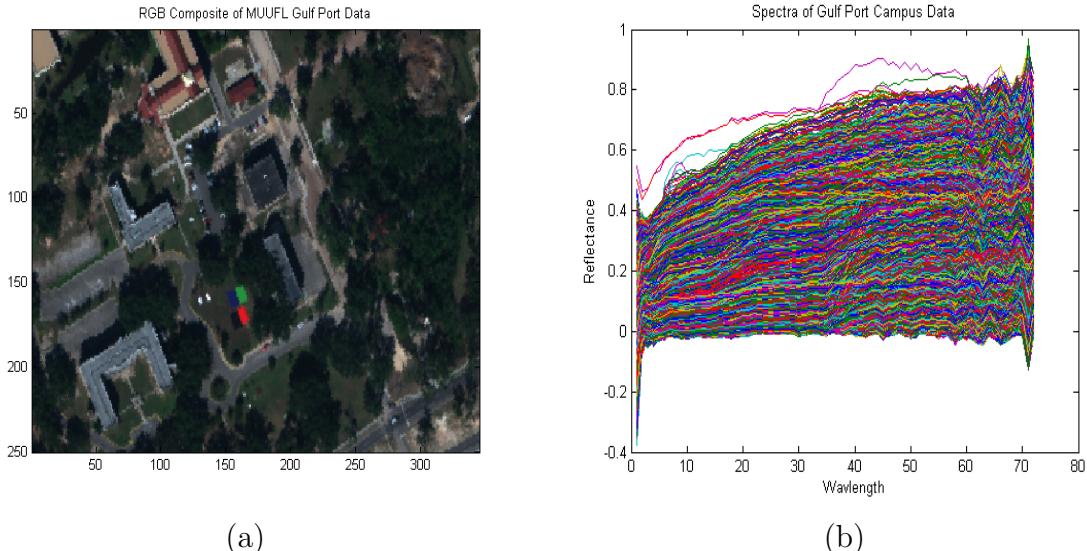


Figure 4.29: MUUFL Gulf Port Data: (a) RGB Composite (b) Spectra

4.2.3.1 PCOMMEND Results

In this experiment, the parameter α was set to 0.001. It can be observed from the Fig. 4.30 (a) that in partition 1 PCOMMEND has identified roofs and other man-made structure along with shadows as endmember 1 and 2. In partition 2, trees and grass has been identified. Looking at the product maps in partition 1, this method resulted in the mixing of man-made and natural endmember (man-made materials and shadows) in it.

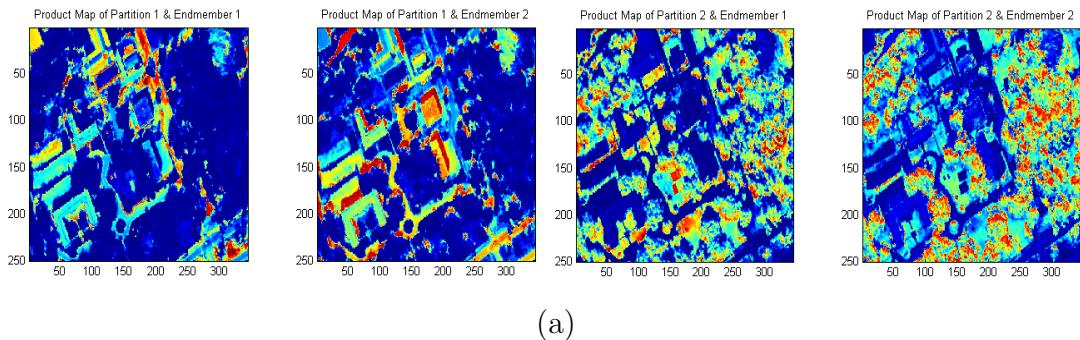


Figure 4.30: Gulf Port- PCOMMEND: Product maps of endmembers in partition 1 and partition 2

4.2.3.2 SUBSUME-SCAD-Weight Decay Results

In this experiment α and δ as 0.0342 and 0.1971 respectively. From Fig. 4.31 (a), it is clear that in this method trees and grass are identified as the endmembers 1 and 2 in partition 1 and roofs-side walks and shadows are identified as endmember 1 and 2 in partition 2. Although the product maps indicate mixing of man-made and natural endmememebrs, the product maps corresponding to shadows has very high intensity as compared to the product maps with PCOMMEND, SUBSUME-BST-Sparsity Promoting, and SUBSUME-BST-Weight Decay.

Fig. 4.31(c,d) shows the weighted endmembers in partition 1 and 2. It can be observed that due to the presence of Gaussian prior as weight decay term, SUBSUME-BST-WD assigned several small non-zero weights to all the bands resulting in the all of the 72 dimensions.

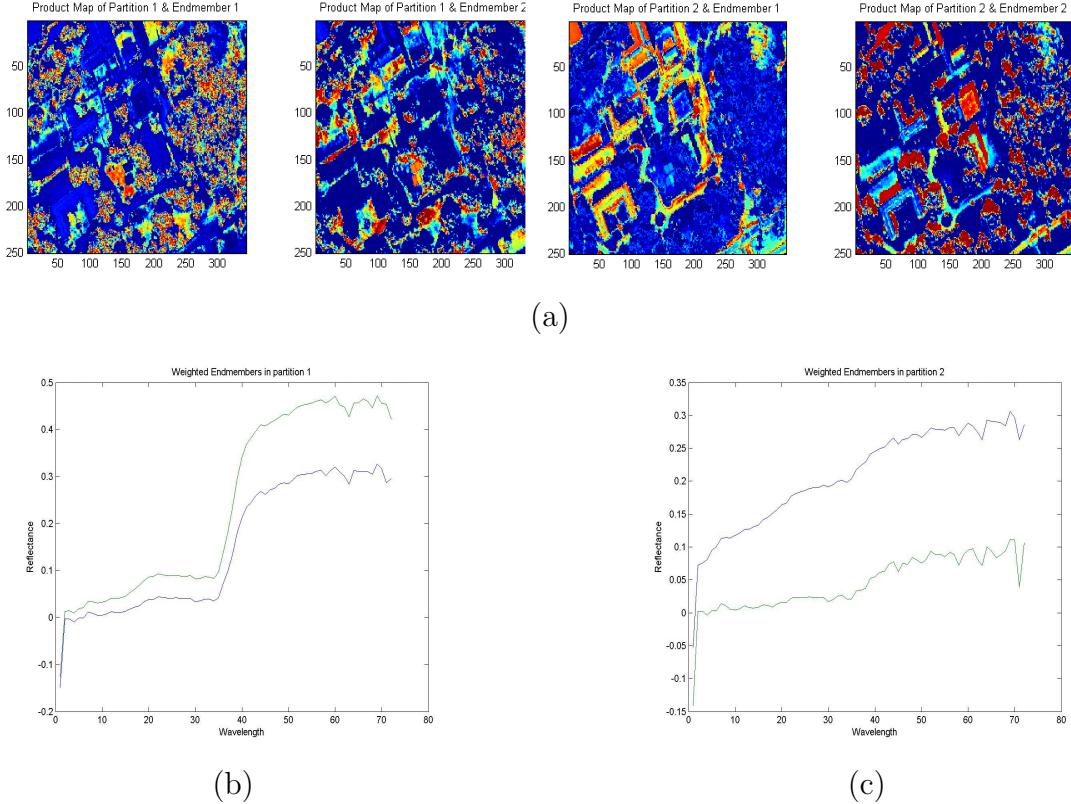


Figure 4.31: Gulf Port- SUBSUME-SCAD-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.3.3 SUBSUME-SCAD-Sparsity Promoting Results

In this experiment α and δ were set to 0.0011 and 0.5507 respectively. From Fig. 4.32(a) it can be observed that SUBSUME-SCAD-Sparsity Promoting approach re-

sulted in the endmembers which are well separated across the partitions. Unlike the product maps of PCOMMEND and SUBSUME-BST-Weight Decay approach, this method separated natural and man-made materials present in the scene apart. In the partition 1 it has identified trees along with the shadows and grass as endmember 1 and 2. On the other hand, in partition 2 it has identified roofs and other man-made materials along with side walk as endmember 1 and 2.

From the weighted endmembers in Fig. 4.32 (b,c) it can be seen that this method proved to be efficient in reducing the dimensionality of the data is reduced from 72 to 40.

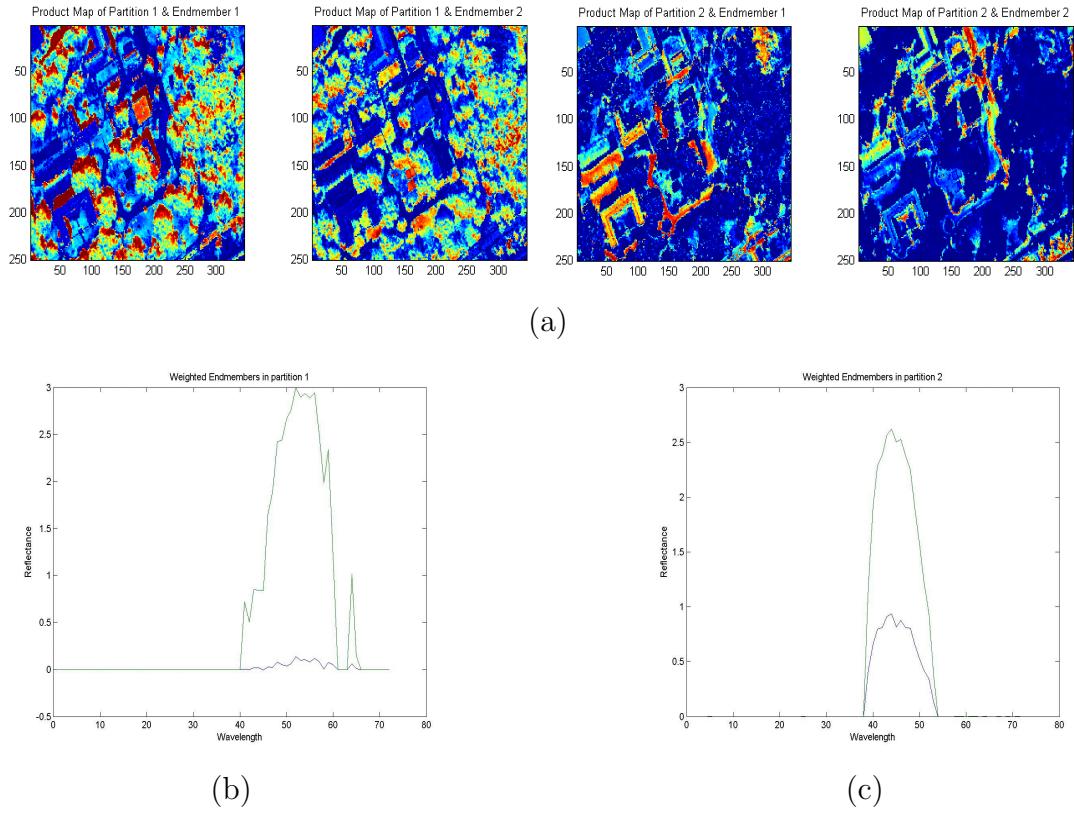


Figure 4.32: Gulf Port- SUBSUME-SCAD-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

4.2.3.4 SUBSUME-BST-Weight Decay Results

For this experiment the parameters α and Δ were set to 0.0043 and 1000 respectively. It can be observed from the product maps in figure 4.33 (a) that, similar to the results in PCOMMEND, product maps generated with this method also mixed natural endmember with the man-made material. With this method, grass and trees are found as endmember 1 and 2 in partition 1 and roofs, side walk and shadows as endmember 1 and 2 in partition 2. However, as compared to the PCOMMEND, the product maps with this method indicate presence of endmembers in high intensity in each partition.

Fig. 4.33 (b,c) shows the weighted endmembers in partition 1 and 2 respectively. It is clear that this method has resulted in poor band weighting as compared to the weighting by SUBSUME-BST-Sparsity Promoting method. Due to the presence of Gaussian prior in the objective function, weights have several small non-zero values.

4.2.3.5 SUBSUME-BST-Sparsity Promoting Results

In this experiment the input parameters α and Δ were set to 0.0907 and 5000. It can be observed from the product maps in Fig. 4.34 (a), in partition 1, it has identified roofs, side walks and other man-made structure present in the scene as endmember 1 and 2 respectively. In partition 2, it has identified natural materials i.e. trees, grass and shadows as endmember 1 and 2. As compared to the PCOMMEND and other SUBSUME results, these product maps shows an effective separation of endmembers corresponding to both partitions and these product maps shows very high amount of endmembers in each partition.

Fig. 4.34(b,c) shows weighted endmembers in partition 1 and 2 respectively. It can be seen from these weighted endmember plots that the SUBSUME-BST-Sparsity

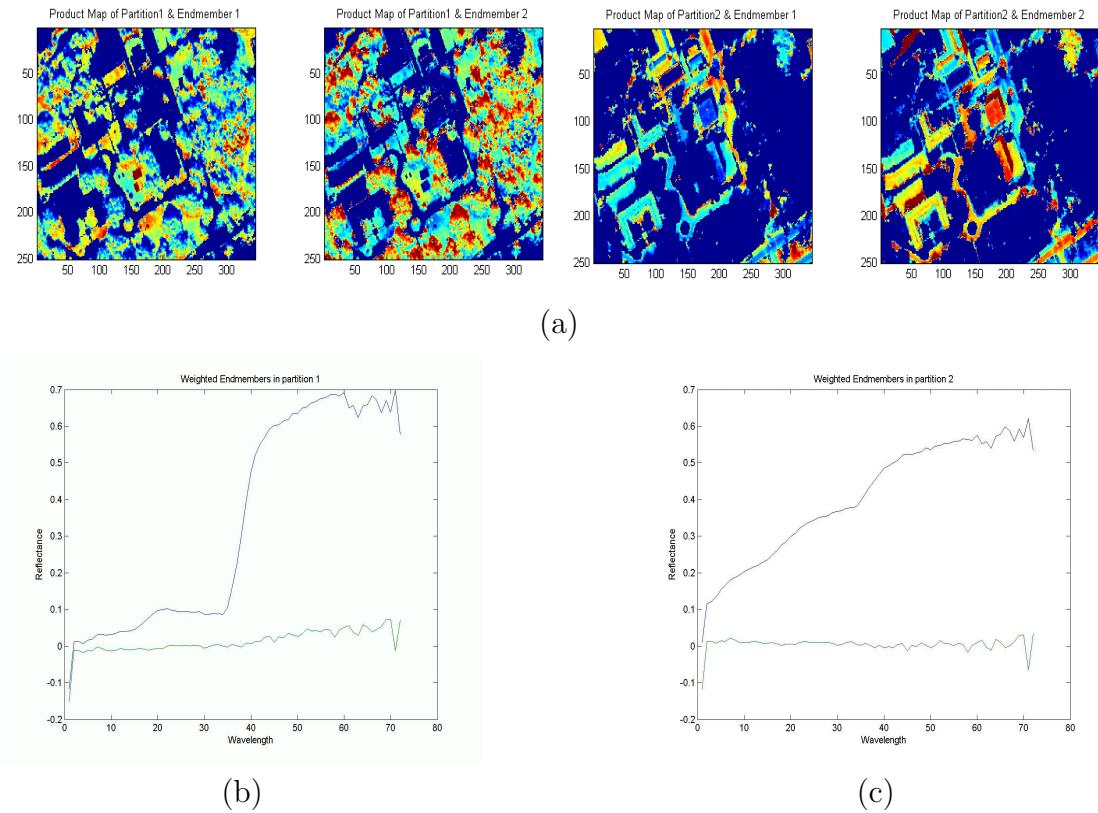


Figure 4.33: SUBSUME-BST-WD: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

Promoting approach is effective in reducing the dimensionality of the data. After band weighting, this method retained 47 bands thus reducing the dimensionality of the data from 120 to 47 bands.

Also it can be observed that SUBSUME-BST-Sparsity Promotion resulted in more significant weights in each partition as compared to the SUBSUME-SCAD-Sparsity Promotion approach.

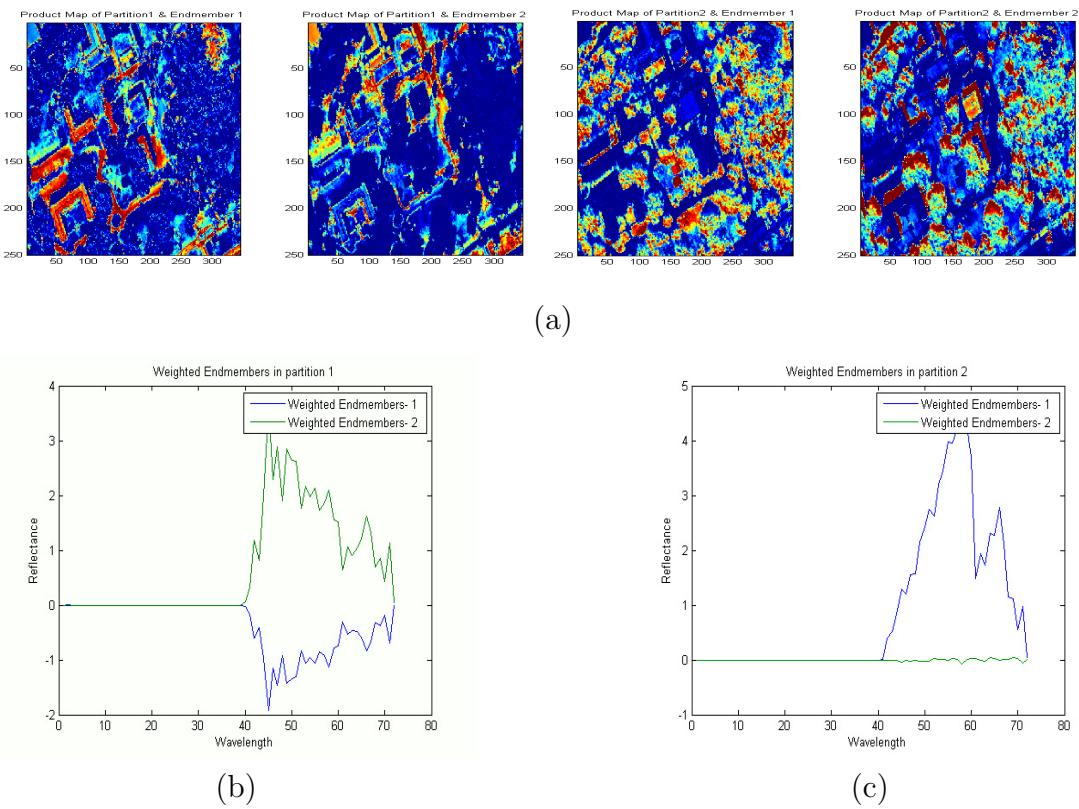


Figure 4.34: Gulf Port- SUBSUME-BST-SP: (a) Product maps of endmembers in partition 1 and 2 (b) Weighted endmembers in partition 1 (c) Weighted endmembers in partition 2

Chapter 5

Conclusion and Future Work

5.1 Conclusion

New SUBSUME methods for hyperspectral unmixing for non-convex data sets and simultaneous dimensionality reduction by assigning partition specific band weights are developed and investigated in this work.

Four different SUBSUME algorithms: SUBSUME-SCAD-Weight Decay, SUBSUME-BST-Weight Decay, SUBSUME-SCAD-Sparsity Promotion, and SUBSUME-BST-Sparsity Promotion are presented and discussed on the experimental results with simulated hyperspectral data generated using ASTER library and three real hyperspectral data: Indian Pines data set, Pavia University data set, and MUUFL Gulf Port data set. Quantitative results with the simulated hyperspectral data and qualitative results with the real hyperspectral data sets are presented.

All the proposed SUBSUME algorithms are compared with the existing hyperspec-

tral unmixing algorithm for non-convex data sets: PCOMMEND (Piecewise Convex Multiple Model Endmember Detection and spectral unmixing).

Quantitative results on simulated hyperspectral, obtained by estimating Earth Movers Distance-based Squared Euclidean Distance (EMD-SED) and Earth Movers Distance-based Spectral Angle Mapper (EMD-SAM), shows that the proposed SUBSUME algorithms outperforms the existing PCOMMEND. Qualitative results (product maps) are obtained on the three real hyperspectral data sets mentioned above. By the visual inspection of the product maps it can be observed that SUBSUME algorithms are effective at distinguishing between distinct materials, grouping similar material across endmembers and partitions, and reducing the dimensionality of the data set by band weighting.

Results on both simulated hyperspectral data set and real hyperspectral data sets shows that SUBSUME-BST-Sparsity Promoting approach outperforms among all the SUBSUME methods presented in this work. The sparsity promoting term and the BST term in SUBSUME-BST-Sparsity Promoting approach objective function results in more effective separation between different endmembers in different clusters and encourages exact zero weights to the irrelevant bands i.e. the bands having very high covariance with the neighboring bands.

5.2 Future Work

The SUBSUME algorithms presented in this work requires prior knowledge of number of partitions and number of endmembers per partition for a hyperspectral scene. Also, the algorithms presented in this work needs to have same number of endmembers in

all partitions. However there could be a case where there is an effective representation of the input non-convex hyperspectral data by the different number of endmembers per partition.

With this motivation, the future work would be an extension of SUBSUME approaches by automatically estimating number of partition and number of endmembers per partition required to represent a non-convex hyperspectral scene along with the flexibility of having different number of endmembers per partition.

Appendix A

Derivations: Weight Decay Approaches

This appendix contains derivation of all the update equations for endmembers, proportions, weights, and memberships for SUBSUME-BST-Weight Decay approach and SUBSUME-SCAD-Weight Decay approach.

A.1 SUBSUME-SCAD-WD

This section will derive endmembers, proportions, memberships, and weights from the SUBSUME-SCAD-Weight Decay objective function given in (A.1)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{d=1}^D v_{cd}^2 \right) \quad (\text{A.1})$$

subject to following constraints:

$$p_{cnm} \geq 0; \sum_{m=1}^M p_{cnm} = 1 \quad \forall c = \{1, \dots, C\}, \forall n = \{1, \dots, N\}$$

$$u_{cn} \geq 0; \sum_{c=1}^C u_{cn} = 1 \quad \forall c = \{1, \dots, C\}, \forall n = \{1, \dots, N\}, u_{cn} \in [0, 1]$$

$$v_{cd} \in [0, 1] \text{ and } \sum_{d=1}^D v_{cd} = D \quad \forall c = \{1, \dots, C\}, \forall d = \{1, \dots, D\} \quad (\text{A.2})$$

where u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in end member set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is a $1 \times M$ vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, δ and α are fixed parameter values used to balance the terms of the objective function.

A.1.1 Membership

The update equation for Membership is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint as shown in (A.2),

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) &= \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta_c \sum_{\mathbf{d}=1}^{\mathbf{D}} \mathbf{v}_{\mathbf{cd}}^2 \right) \\
&\quad - \sum_{n=1}^N \lambda_n \left(1 - \sum_{c=1}^C u_{cn} \right)
\end{aligned} \tag{A.3}$$

The partial derivative of (A.3) with respect to u_{cn} is shown in (A.4),

$$\frac{\partial J}{\partial u_{cn}} = q u_{cn}^{q-1} \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 - \lambda_n \tag{A.4}$$

Substituting (A.4) to 0 in order to solve for u_{cn} ,

$$u_{cn} = \left[\frac{\lambda_n}{q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2} \right]^{\frac{1}{q-1}} \quad 1 \leq c \leq C, 1 \leq n \leq N \tag{A.5}$$

Expression for λ_n can be found by taking the derivative of (A.3) with respect to λ_n and substituting the derivative to zero,

$$\sum_{c=1}^C u_{cn} = 1 \tag{A.6}$$

Equation (A.6) can be rewritten by substituting the value of u_{cn} from (A.5),

$$\begin{aligned}
\sum_{c=1}^C \left[\frac{\lambda_n}{q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2} \right]^{\frac{1}{q-1}} &= 1, 1 \leq \lambda_n \leq N \\
\lambda_n &= \sum_{c=1}^C \left(q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \right)^{-\frac{1}{q-1}}
\end{aligned} \tag{A.7}$$

Final update equation for u_{cn} can be found out by substituting the value of λ_n from (A.7) to (A.5)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left[\frac{\sum_{d=1}^D v_{cd}(x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}(x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right]^{\frac{1}{q-1}}} \quad (\text{A.8})$$

A.1.2 Weights

The update equation for Weights is found by solving $\frac{\partial J}{\partial v_{cd}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-D constraint in (A.2)

$$\begin{aligned} J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) &= \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta_c \sum_{d=1}^D \mathbf{v}_{cd}^2 \right) \\ &\quad - \sum_{c=1}^C \lambda \left(D - \sum_{d=1}^D v_{cd} \right) \end{aligned} \quad (\text{A.9})$$

The partial derivative of (A.9) with respect to v_{cd} is shown in (A.10),

$$\frac{\partial J}{\partial v_{cd}} = \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + 2\delta v_{cd} - \lambda_c \quad (\text{A.10})$$

Substituting (A.10) to 0 would result in the expression for weights and it is shown in (A.11),

$$v_{cd} = \frac{\lambda_c - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta} \quad (\text{A.11})$$

The expression for λ_c can be found by taking the derivative of (A.9) w.r.t λ_c and substituting it to 0,

$$\sum_{d=1}^D v_{cd} - D = 0 \quad (\text{A.12})$$

Equation (A.12) can be re-written by substituting the value of v_{cd} from (A.11) and solving for λ_c

$$\sum_{d=1}^D \frac{\lambda_c - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta} - D = 0$$

$$\sum_{d=1}^D \left[\lambda_c - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \right] - D2\delta = 0$$

$$\sum_{d=1}^D \lambda_c = D2\delta + \sum_{d=1}^D \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2$$

$$D\lambda_c = D2\delta + \sum_{d=1}^D \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2$$

$$\lambda_c = \frac{1}{D} \left[D2\delta + \sum_{d=1}^D \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \right] \quad (\text{A.13})$$

Equation (A.11) can be re-written by substituting the value of λ_c from (A.13) to (A.11) and it is shown in (A.14),

$$v_{cd} = \frac{\frac{1}{D} \left[D2\delta + \sum_{d=1}^D \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \right] - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta} \quad (\text{A.14})$$

The general form of p -norm is shown in A.15,

$$\|x\|_p = \left[\sum_{i=1}^n |x_i|^p \right]^{\frac{1}{p}} \quad (\text{A.15})$$

Using the general form of P-norm in (A.15), we can re-write the RSS term in (A.14) as,

$$\sum_{d=1}^D (x_{nd} - E_{cd}\mathbf{p}_{cn})^2 = \left[\left[\sum_{d=1}^D |x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn}|^2 \right]^{\frac{1}{2}} \right]^2 = \|x_n - \mathbf{E}_c\mathbf{p}_{cn}\|^2 \quad (\text{A.16})$$

The final update equation for weights is found by substituting the RSS term from (A.16) in (A.14),

$$v_{cd} = 1 + \frac{1}{2\delta} \sum_{n=1}^N u_{cn}^q \left[\frac{\|\mathbf{x}_n - \mathbf{E}_c\mathbf{p}_{cn}\|_2^2}{D} - (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \right]$$

A.1.3 Endmembers

It is clear from the objective function in (A.1), that endmembers depend on the SSD term of this objective function given in (A.17),

$$SSD = \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \quad (\text{A.17})$$

The SSD term given in (A.17) can be rewritten as shown in (A.18),

$$SSD = \sum_{m=1}^{M-1} \sum_{k=m+1}^M \sum_{d=1}^D v_{cd} (e_{cmd} - e_{ckd})^T (e_{cmd} - e_{ckd}) \quad (\text{A.18})$$

Further rewriting the SSD term in (A.18) in matrix form would result in the expression of SSD as shown in (A.19),

$$SSD = \sum_{d=1}^D v_{cd} (M.trace(\mathbf{E}_{cd}\mathbf{E}_{cd}^T) - 1_{1 \times M} \mathbf{E}_{cd}\mathbf{E}_{cd}^T 1_{M \times 1}) \quad (\text{A.19})$$

We can rewrite the objective function in (A.1) by replacing the original SSD term from the objective function by its equivalent matrix form as shown in (A.20),

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{d=1}^D v_{cd} (M.trace(\mathbf{E}_{cd}\mathbf{E}_{cd}^T) - 1_{1 \times M} \mathbf{E}_{cd}\mathbf{E}_{cd}^T 1_{M \times 1}) + \delta \sum_{d=1}^D v_{cd}^2 \right) \quad (\text{A.20})$$

The update equation for Endmembers is found by taking partial derivative of (A.20) with respect to \mathbf{E}_{cd} as shown in (A.21),

$$\frac{\partial J}{\partial \mathbf{E}_{cd}} = \sum_{n=1}^N u_{cn}^q v_{cd} (2\mathbf{E}_{cd}\mathbf{p}_{cn}^T \mathbf{p}_{cn} - 2\mathbf{p}_{cn}^T x_{nd}) + 2v_{cd}\alpha (MI_{M \times M} - 1_{M \times M}) \mathbf{E}_{cd} \quad (\text{A.21})$$

The endmembers can be solved by setting (A.21) to 0, as shown in (A.22)

$$\sum_{n=1}^N u_{cn}^q v_{cd} (2\mathbf{E}_{cd}\mathbf{p}_{cn}^T \mathbf{p}_{cn} - 2\mathbf{p}_{cn}^T x_{nd}) + 2\alpha v_{cd} (MI_{M \times M} - 1_{M \times M}) \mathbf{E}_{cd} = 0 \quad (\text{A.22})$$

$$\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{E}_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} - \sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T x_{nd} + v_{cd}\alpha (MI_{M \times M} - 1_{M \times M}) \mathbf{E}_{cd} = 0$$

$$\mathbf{E}_{cd} \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} + v_{cd}\alpha (MI_{M \times M} - 1_{M \times M}) \right) = \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T x_{nd} \right)$$

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} + v_{cd} \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q v_{cd} \mathbf{p}_{cn}^T x_{nd} \right)$$

The final update equation for endmembers is shown in (A.23),

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn}^T \mathbf{p}_{cn} + \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn}^T x_{nd} \right) \quad (\text{A.23})$$

A.1.4 Proportions

The update equation for Proportion is found by adding a Lagrange multiplier term to enforce the sum-to-one constraint and Karsuh-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraint given in (A.2),

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \right. \\ \left. + \delta \sum_{d=1}^D v_{cd}^2 \right) - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T \quad (\text{A.24})$$

By expanding and re-writing the RSS term in (A.24) as shown in (A.25),

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = & \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^T (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn}) \right. \\
& + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \\
& \left. + \delta \sum_{d=1}^D v_{cd}^2 \right) - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T
\end{aligned}$$

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = & \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd}^T x_{nd} - x_{nd}^T \mathbf{E}_{cd}\mathbf{p}_{cn} - \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T x_{nd} - \mathbf{E}_{cd}^T \mathbf{E}_{cd}\mathbf{p}_{cn}^T \mathbf{p}_{cn}) \right. \\
& + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{\mathbf{d}=1}^{\mathbf{D}} \mathbf{v}_{cd}^2 \\
& \left. - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T \right)
\end{aligned} \tag{A.25}$$

Proportions are solved by taking the partial derivative (A.25) with respect to \mathbf{p}_{cn} as shown in (A.26),

$$\frac{\partial J}{\partial p_{cn}^T} = -2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T + 2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_c \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T - \lambda_c 1_{M \times 1} - \gamma_{cn} \tag{A.26}$$

By substituting (A.26) to 0, we can rewrite p_{cn} as shown in (A.27),

$$-2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_c \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T - \lambda_c \mathbf{1}_{M \times 1} - \gamma_{cn} = 0$$

$$2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T = \lambda_c \mathbf{1}_{M \times 1} + \gamma_{cn} + 2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T$$

$$\mathbf{p}_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + \gamma_{cn} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T \right) \quad (\text{A.27})$$

As per the KKT conditions $\gamma_{cn} \geq 0$, $\mathbf{p}_{cn}^T \geq 0$ and $\gamma_{cn} \mathbf{p}_{cn}^T = 0$. If $\gamma_{cn} > 0$, then $\mathbf{p}_{cn}^T = 0$. Thus γ_{cn} should be equal to zero and $\mathbf{p}_{cn}^T \geq 0$. So, (A.27) can be re-written as shown in (A.28),

$$\mathbf{p}_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T \right) \quad (\text{A.28})$$

or,

$$\mathbf{p}_{cn}^{KKT} = \max(\mathbf{p}_{cn}^T, 0) \quad (\text{A.29})$$

p_{cn} can be re-written as shown in (A.30),

$$\mathbf{1}_{1 \times M} \mathbf{p}_{cn}^T = 1 \quad (\text{A.30})$$

By substituting the value of p_{cn} from (A.28) in (A.30) and solving for λ_c ,

$$\mathbf{1}_{1 \times M} \left[\left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T \right) \right] = 1$$

$$1_{1 \times M} \left[\left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} (\lambda_c 1_{M \times 1}) + \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T \right) \right] = 1$$

$$1_{1 \times M} \frac{\lambda_c}{2} \left(u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1} + 1_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T = 1$$

$$1_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T = 1 - 1_{1 \times M} \frac{\lambda_c}{2} \left(u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}$$

The final equation for λ_c is shown in (A.31),

$$\lambda_c = 2 \frac{1 - 1_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T}{1_{1 \times M} \left(u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}} \quad (\text{A.31})$$

A.2 SUBSUME-BST-WD

This section will derive endmembers, proportions, memberships, and weights from the SUBSUME-BST-Weight Decay objective function given in (A.32)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M \sum_{d=1}^D (e_{cmd} - e_{ckd})^T (\mathbf{e}_{cmd} - \mathbf{e}_{ckd}) + \mathbf{BST} \right) \quad (\text{A.32})$$

subject to the constraints given in (A.2). In (A.32), BST term is Band Sparsity Term and it is given as in (A.33),

$$BST = \sum_{d=1}^D \delta_{cd} v_{cd}^2 \quad (\text{A.33})$$

where δ_{cd} is the dissimilarity term between data and the estimated endmembers and it is given in (A.34),

$$\delta_{cd} = \frac{\Delta \left(\left(\sum_{m=1}^M \frac{1}{A} \sum_{n=1}^N u_{cn} \mathbf{p}_{cn} (x_{nd} - e_{cmd})^2 \right) + 1 \right)}{\left(\sum_{m=1}^M (e_{cmd} - \mu_{0cd})^2 \right) + 1} \quad (\text{A.34})$$

where Δ is a constant and μ_{0cd} is the global mean given by (A.35),

$$\mu_{0cd} = \frac{\sum_{n=1}^N x_{nd} u_{cn}}{\sum_{c=c}^N u_{cn}} \quad (\text{A.35})$$

Also A is given by:

$$A = \sum_{n=1}^N u_{cn} \quad (\text{A.36})$$

A.2.1 Membership

δ_{cd} in (A.32) is a scalar value for each band in each cluster. The update equation for membership is found by solving $\frac{\partial J}{\partial u_{cn}} = 0$ after adding a Lagrange multiplier term to enforce the sum-to-one constraint as shown in (A.2),

$$\frac{\partial J}{\partial u_{cn}} = \mathbf{q}u_{cn}^{q-1} \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 - \lambda_n \quad (\text{A.37})$$

Rest of the derivation follows similar steps as in A.1.1. Final update equation for membership for SUBSUME-BST-Weight Decay approach is given as in (A.38),

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left(\frac{\sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd} (x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right)^{\frac{1}{q-1}}} \quad (\text{A.38})$$

A.2.2 Weights

According to the constraints given in (A.2), weights v_{cd} are subject to sum-to-D constraints and non-negativity constraints, such that $v_{cd} \in [0, D]$, $\sum_{d=1}^D v_{cd} = D$, and $v_{cd} \geq 0$. Taking the partial derivative of the objective function in (A.32) with respect to v_{cd} as shown in (A.38),

$$\frac{\partial J}{\partial v_{cd}} = \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + 2\delta_{cd}v_{cd} - \lambda_c \quad (\text{A.38})$$

By substituting the (A.38) to 0 as shown in (A.39) and solving for weights,

$$\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + 2\delta_{cd}v_{cd} - \lambda_c = 0; \quad (\text{A.39})$$

$$2\delta_{cd}v_{cd} = \lambda_c - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \quad (\text{A.40})$$

$$v_{cd} = \frac{\lambda_c - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \quad (\text{A.41})$$

Taking the derivative of (A.32) with respect to λ_c ,

$$\sum_{d=1}^D v_{cd} = D \quad (\text{A.42})$$

Equation (A.42) can be re-written by substituting the value of v_{cd} from (A.41) as shown in (A.43) and solving for λ_c ,

$$\sum_{d=1}^D \frac{\lambda_c}{2\delta_{cd}} - \sum_{d=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \right] = D \quad (\text{A.43})$$

$$\lambda_c \sum_{d=1}^D \frac{1}{2\delta_{cd}} - \sum_{d=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \right] = D \quad (\text{A.44})$$

The final value of λ_c is given in (A.45),

$$\lambda_c = \frac{D + \sum_{d=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \right]}{\sum_{d=1}^D \frac{1}{2\delta_{cd}}} \quad (\text{A.45})$$

The expression of v_{cd} in (A.41) is further simplified by substituting the value of λ_c from (A.45) to (A.41) as shown in (A.46),

$$v_{cd} = \frac{\frac{D + \sum_{d=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \right]}{\sum_{d=1}^D \frac{1}{2\delta_{cd}}} - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{2\delta_{cd}} \quad (\text{A.46})$$

$$v_{cd} = \frac{D + \sum_{d=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2}{2\delta_{cd}} \right] - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \sum_{d=1}^D \frac{1}{2\delta_{cd}}}{\left[\sum_{d=1}^D \frac{1}{2\delta_{cd}} \right] 2\delta_{cd}} \quad (\text{A.47})$$

The final update equation for v_{cd} is shown in (A.48),

$$v_{cd} = \frac{D + \sum_{w=1}^D \left[\frac{\sum_{n=1}^N u_{cn}^q (x_{nw} - \mathbf{E}_{cw} \mathbf{p}_{cn})^2}{2\delta_{cw}} \right] - \sum_{n=1}^N u_{cn}^q (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 \sum_{w=1}^D \left[\frac{1}{2\delta_{cw}} \right]}{\left[\sum_{w=1}^D \frac{1}{2\delta_{cw}} \right] 2\delta_{cd}} \quad (\text{A.48})$$

where δ_{cd} is given by A.34

A.2.3 Endmembers

The update equation for Endmember is found by taking partial derivative of (A.32) with respect to E_{cd} as shown in (A.49),

$$\frac{\partial J}{\partial \mathbf{E}_{cd}} = \sum_{n=1}^N u_{cn}^q v_{cd} (2\mathbf{E}_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} - 2\mathbf{p}_{cn}^T x_{nd}) + 2v_{cd}\alpha (MI_{M \times M} - 1_{M \times M}) \mathbf{E}_{cd} \quad (\text{A.49})$$

Rest of the derivation follows similar steps as in A.1.3. Final update equation for endmembers is given in (A.50),

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} \mathbf{p}_{cn}^T + \alpha (MI_{M \times M} - 1_{M \times M}) \right)^{-1} \times \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} x_{nd}^T \right) \quad (\text{A.50})$$

A.2.4 Proportions

The update equation for Proportion is found by adding a Lagrange multiplier term to enforce the sum-to-one constraint and Karsuh-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraint given in (A.2),

$$\begin{aligned} J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) &= \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \sum_{d=1}^D \delta_{cd} \mathbf{v}_{cd}^2 \right) \\ &\quad - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T \end{aligned} \quad (\text{A.51})$$

Proportions are solved by taking the partial derivative (A.51) with respect to \mathbf{p}_{cn} as shown in (A.52),

$$\frac{\partial J}{\partial p_{cn}^T} = -2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{x}_n^T + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_c \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T - \lambda_c 1_{M \times 1} - \gamma_{cn} \quad (\text{A.52})$$

The rest of the derivation follows similar steps as in A.1.4. Final update equation for proportion is given as,

$$p_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c 1_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd} \right)$$

where,

$$\lambda_c = 2 \frac{1 - 1_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd}}{1_{1 \times M} \left(D \mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}} \quad (\text{A.53})$$

Appendix B

Derivations: Sparsity Promoting Approaches

This appendix contains derivation of all the update equations for endmembers, proportions, weights, and memberships for SUBSUME-BST-Sparsity Promoting approach and SUBSUME-SCAD-Sparsity Promoting approach.

B.1 SUBSUME-SCAD-SP

This section will derive endmembers, proportions, memberships, and weights from the SUBSUME-SCAD-Sparsity Promoting objective function given in (B.1)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{d=1}^D |\mathbf{v}_{cd}| \right) \quad (\text{B.1})$$

subject to following constraints:

$$p_{cnm} \geq 0; \sum_{m=1}^M p_{cnm} = 1 \quad \forall c = \{1, \dots, C\}, \forall n = \{1, \dots, N\}$$

$$u_{cn} \geq 0; \sum_{c=1}^C u_{cn} = 1 \quad \forall c = \{1, \dots, C\}, \forall n = \{1, \dots, N\}, u_{cn} \in [0, 1]$$

$$v_{cd} \in [0, 1] \text{ and } \sum_{d=1}^D v_{cd} = D \quad \forall c = \{1, \dots, C\}, \forall d = \{1, \dots, D\} \quad (\text{B.2})$$

where u_{cn} is the membership value of the n^{th} data point in the c^{th} endmember set, q is a fixed parameter which controls the degree of sharing across endmember sets, v_{cd} is the band weight for the d^{th} band in end member set c , x_{nd} is the d^{th} element of the n^{th} pixel, \mathbf{p}_{cn} is a $1 \times M$ vector of proportion values associated with pixel n for the c^{th} endmember set, \mathbf{E}_{cd} is a $1 \times M$ vector containing the d^{th} band value for each of the M endmembers in set c , \mathbf{e}_{cm} is the $d \times 1$ vector representing the m^{th} endmember in the c^{th} endmember set, δ and α are fixed parameter values used to balance the terms of the objective function.

B.1.1 Membership

Refer A.2.1 for complete derivation. The final update equation for membership is found out as in (B.3),

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left(\frac{\sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd} (x_{nd} - \mathbf{E}_{kd} \mathbf{p}_{kn})^2} \right)^{\frac{1}{q-1}}} \quad (\text{B.3})$$

B.1.2 Weights

It is clear from the objective function in (B.1), that the weights does not depend on the SSD term of objective function, so we can rewrite the objective function in B.1 by eliminating the SSD term as shown in (B.4),

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \delta \sum_{d=1}^D |v_{cd}| \right) \quad (\text{B.4})$$

Weights can be solved by using quadratic programming problem. The standard form of quadratic programming problem is given in (B.5),

$$\min_x \frac{1}{2} x^T H x + f^T x \quad (\text{B.5})$$

subject to following constraints:

$$Ax \leq b; \quad A_{eq}x = b_{eq}; \quad lowerLimit \leq x \leq upperLimit \quad (\text{B.6})$$

where H represents the quadratic term in the (B.5), f is the linear term in the (B.5), A is the linear coefficient in the constraint in $A.x \leq b$, b is the constant vector in the constraint in $A.x \leq b$, A_{eq} is the linear coefficient in $A_{eq}.x = b_{eq}$, b_{eq} is the constant vector in the constraint in $A_{eq}.x = b_{eq}$, and *Lower Limit* and *Upper Limit* are the lower bound and upper bound of the constraints given in (B.6).

By re-writing B.4 as shown in (B.7),

$$J = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \delta \sum_{d=1}^D v_{cd} \right) \quad (\text{B.7})$$

Assuming the number of cluster, $C = 1$ in B.7 in order to make the derivation easier.

We can re-write the (B.7) as shown in (B.8),

$$J = \left(\sum_{n=1}^N u_n^q \sum_{d=1}^D v_d (x_{nd} - \mathbf{E}_d \mathbf{p}_n)^2 + \delta_c \sum_{d=1}^D \mathbf{v}_d \right) \quad (\text{B.8})$$

where δ_c is a constant the does not depend on the clusters. B.8 can be re-written by expanding all its terms as shown in (B.9),

$$\begin{aligned} J = & u_1^q [v_1 (x_{11} - \mathbf{E}_{1m} \mathbf{p}_{1m})^2 + v_2 (x_{12} - \mathbf{E}_{2m} \mathbf{p}_{1m})^2 + \dots + v_d (x_{1d} - \mathbf{E}_{dm} \mathbf{p}_{1m})^2] \\ & + u_2^q [v_1 (x_{21} - \mathbf{E}_{1m} \mathbf{p}_{2m})^2 + v_2 (x_{22} - \mathbf{E}_{2m} \mathbf{p}_{2m})^2 + \dots + v_d (x_{2d} - \mathbf{E}_{dm} \mathbf{p}_{2m})^2] \\ & \vdots \\ & + u_n^q [v_1 (x_{n1} - \mathbf{E}_{1m} \mathbf{p}_{nm})^2 + v_2 (x_{n2} - \mathbf{E}_{2m} \mathbf{p}_{nm})^2 + \dots + v_d (x_{nd} - \mathbf{E}_{dm} \mathbf{p}_{nm})^2] \\ & + \delta [v_1 + v_2 + \dots + v_d] \end{aligned} \quad (\text{B.9})$$

Re-arranging the (B.9), by separating coefficients of $v_1, v_2 \dots v_d$,

$$\begin{aligned} J = & v_1 [u_1^q (x_{11} - \mathbf{E}_{1m} \mathbf{p}_{1m})^2 + \delta + u_2^q (x_{21} - \mathbf{E}_{1m} \mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{n1} - \mathbf{E}_{1m} \mathbf{p}_{nm})^2 + \delta] \\ & v_2 [u_1^q (x_{12} - \mathbf{E}_{2m} \mathbf{p}_{1m})^2 + \delta + u_2^q (x_{22} - \mathbf{E}_{2m} \mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{n2} - \mathbf{E}_{2m} \mathbf{p}_{nm})^2 + \delta] \\ & \vdots \\ & v_d [u_1^q (x_{1d} - \mathbf{E}_{dm} \mathbf{p}_{1m})^2 + \delta + u_2^q (x_{2d} - \mathbf{E}_{dm} \mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{nd} - \mathbf{E}_{dm} \mathbf{p}_{nm})^2 + \delta] \end{aligned}$$

where the dimensionality of each of its terms is given in (B.10),

$$\begin{array}{c} u \\ \downarrow \\ 1 \times n \end{array} \quad \left(\begin{array}{cccc} x & E & p & \delta \\ \downarrow & \downarrow & \downarrow & \downarrow \\ n \times d & d \times m & n \times m & 1 \times 1 \end{array} \right) \quad (\text{B.10})$$

The linear term f for the quadratic programming problem is given as in (B.11),

$$f = U(X - E^T P) .^* 2 + \delta \quad (\text{B.11})$$

It is clear that there is no H term which represents the quadratic term, in the objective function given in B.1. The weights are solved by using the MATLAB's *quadprog* function. Parameter settings for the *quadprog* is given in (B.12),

Parameter Initialization for quadprog function

$$f = [U(X - E^T P) .^* 2 + \delta_c]^T$$

$$H = \text{diag}(\text{ones}(D, 1))$$

$$lb = \text{zeros}(D, 1)$$

$$ub = D * \text{ones}(D, 1)$$

$$Aeq = \text{ones}(1, D)$$

$$Beq = D \quad (\text{B.12})$$

B.1.3 Endmembers

Refer A.2.3 for complete derivation. Final update equation for endmembers is shown in (B.13),

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} \mathbf{p}_{cn}^T + \alpha (MI_{M \times M} - 1_{M \times M}) \right)^{-1} \times \left(\sum_{n=1}^N u_{cn}^q \mathbf{p}_{cn} x_{nd}^T \right)$$

B.1.4 Proportions

Refer A.2.4 for complete derivation.

The update equation for Proportion is found by adding a Lagrange multiplier term to enforce the sum-to-one constraint and Karsuh-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraint given in (B.2),

$$\begin{aligned} J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) &= \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \delta \sum_{d=1}^D v_{cd} \right) \\ &- \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T \end{aligned} \quad (B.13)$$

Proportions are solved by taking the partial derivative (B.13) with respect to \mathbf{p}_{cn} . Final update equation for proportion is shown in (B.14),

$$p_{cn}^T = \left(2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c 1_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd} \right)$$

where,

$$\lambda_c = 2 \frac{1 - 1_{1 \times M} \left(\sum_{d=1}^D v_{cd} \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D v_{cd} \mathbf{E}_{cd}^T x_{nd}}{1_{1 \times M} \left(D u_{cn}^q \sum_{d=1}^D v_{cd} E_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}} \quad (\text{B.14})$$

B.2 SUBSUME-BST-SP

This section will derive endmembers, proportions, memberships, and weights from the SUBSUME-BST-Sparsity Promoting objective function given in (B.15)

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M \sum_{d=1}^D (e_{cmd} - e_{ckd})^T (e_{cmd} - e_{ckd}) + BST \right) \quad (\text{B.15})$$

Where BST,

$$BST = \sum_{d=1}^D \delta_{cd} |v_{cd}| \quad (\text{B.16})$$

where Δ is a constant and δ_{cd} , μ_{0cd} , and A are given by A.34, ??, and A.36 respectively.

B.2.1 Membership

The update equation for Membership is found by adding a Lagrange multiplier term to enforce the sum-to-one constraint as shown in (B.17),

$$\begin{aligned} J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) &= \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \sum_{d=1}^D \delta_{cd} |v_{cd}| \right) \\ &- \sum_{n=1}^N \lambda_n \left(1 - \sum_{c=1}^C u_{cn} \right) \end{aligned} \quad (\text{B.17})$$

By taking the partial derivative of (B.17) with respect to u_{cn} as shown in (B.18),

$$\frac{\partial J}{\partial u_{cn}} = \mathbf{q} u_{cn}^{q-1} \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 - \lambda_n \quad (\text{B.18})$$

Substituting (B.18) to 0 as shown in (B.19),

$$u_{cn} = \left[\frac{\lambda_n}{q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2} \right]^{\frac{1}{q-1}} \quad 1 \leq c \leq C, 1 \leq n \leq N \quad (\text{B.19})$$

Taking the derivative of (B.17) with respect to λ_n and setting it to zero as shown in (B.20),

$$\sum_{c=1}^C u_{cn} = 1 \quad (\text{B.20})$$

Re-writing (B.20) by substituting the value of u_{cn} from (B.19) as shown in (B.21) and solving for λ_n

$$\sum_{c=1}^C \left[\frac{\lambda_n}{q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2} \right]^{\frac{1}{q-1}} = 1, \quad 1 \leq \lambda_n \leq N \quad (\text{B.21})$$

$$\lambda_n = \sum_{c=1}^C \left(q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 \right)^{-\frac{1}{q-1}} \quad (\text{B.22})$$

Final update equation for membership is found by substituting the value of λ_n from (B.22) to (B.19)

$$u_{cn} = \frac{1}{\sum_{k=1}^C \left[\frac{\sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2}{\sum_{d=1}^D v_{kd}^2 (x_{nd} - \mathbf{E}_{kd}\mathbf{p}_{kn})^2} \right]^{\frac{1}{q-1}}} \quad (\text{B.23})$$

B.2.2 Weights

It is clear from the objective function in (B.15), that the weights does not depend on the SSD term of objective function, so we can rewrite the objective function in B.15 by eliminating the SSD term as shown in (B.24),

It can be observed that, SSD term of objective function given in *B.15* does not depend on weights, so we can eliminate SSD term. Rewriting *B.15* as

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \sum_{d=1}^D \delta_{cd} |v_{cd}| \right) \quad (\text{B.24})$$

Weights can be solved by using quadratic programming problem.

$$\underset{x}{\min} \frac{1}{2} x^T H x + f^T x \quad (\text{B.25})$$

subject to the following constraints:

$$Ax \leq b; \quad A_{eq}x = b_{eq}; \quad lowerLimit \leq x \leq upperLimit$$

Re-writing *B.24* as shown in *(B.26)*,

$$J = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \sum_{d=1}^D \delta_{cd} v_{cd} \right) \quad (\text{B.26})$$

Assuming the number of cluster, $C = 1$ in *(B.26)* in order to make the derivation easier. We can re-write the *(B.26)* as shown in *(B.27)*,

$$J = \left(\sum_{n=1}^N u_n^q \sum_{d=1}^D v_d (x_{nd} - \mathbf{E}_d \mathbf{p}_n)^2 + \delta_c \sum_{d=1}^D \mathbf{v}_d \right) \quad (\text{B.27})$$

Expanding first term of *B.27* in order to solve for the quadratic term (H) of the quadratic programming problem as shown in *(B.28)*,

$$\begin{aligned}
J = & u_1^q [v_1^2 (x_{11} - \mathbf{E}_{1m}\mathbf{p}_{1m})^2 + v_2^2 (x_{12} - \mathbf{E}_{2m}\mathbf{p}_{1m})^2 + \dots + v_d^2 (x_{1d} - \mathbf{E}_{dm}\mathbf{p}_{1m})^2] \\
& + u_2^q [v_1^2 (x_{21} - \mathbf{E}_{1m}\mathbf{p}_{2m})^2 + v_2^2 (x_{22} - \mathbf{E}_{2m}\mathbf{p}_{2m})^2 + \dots + v_d^2 (x_{2d} - \mathbf{E}_{dm}\mathbf{p}_{2m})^2] \\
& \vdots \\
& + u_n^q [v_1^2 (x_{n1} - \mathbf{E}_{1m}\mathbf{p}_{nm})^2 + v_2^2 (x_{n2} - \mathbf{E}_{2m}\mathbf{p}_{nm})^2 + \dots + v_d^2 (x_{nd} - \mathbf{E}_{dm}\mathbf{p}_{nm})^2]
\end{aligned} \tag{B.28}$$

Equation B.28 can be rewritten by separating the coefficients of $v_1^2, v_2^2 \dots v_d^2$,

$$\begin{aligned}
J = & v_1^2 [u_1^q (x_{11} - \mathbf{E}_{1m}\mathbf{p}_{1m})^2 + \delta_c + u_2^q (x_{21} - \mathbf{E}_{1m}\mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{n1} - \mathbf{E}_{1m}\mathbf{p}_{nm})^2 + \delta_c] \\
& v_2^2 [u_1^q (x_{12} - \mathbf{E}_{2m}\mathbf{p}_{1m})^2 + \delta_c + u_2^q (x_{22} - \mathbf{E}_{2m}\mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{n2} - \mathbf{E}_{2m}\mathbf{p}_{nm})^2 + \delta_c] \\
& \vdots \\
& v_d^2 [u_1^q (x_{1d} - \mathbf{E}_{dm}\mathbf{p}_{1m})^2 + \delta_c + u_2^q (x_{2d} - \mathbf{E}_{dm}\mathbf{p}_{2m})^2 + \delta_c + \dots + u_n^q (x_{nd} - \mathbf{E}_{dm}\mathbf{p}_{nm})^2 + \delta_c]
\end{aligned} \tag{B.29}$$

(B.30)

where the dimensionality of the each term is given in (B.31),

$$\begin{array}{cccc}
\begin{matrix} u \\ \downarrow \\ 1 \times n \end{matrix} & \begin{matrix} V^2 \\ \downarrow \\ 1 \times d \end{matrix} & \left(\begin{matrix} x \\ \downarrow \\ n \times d \end{matrix} \quad \begin{matrix} E \\ \downarrow \\ d \times m \end{matrix} \quad \begin{matrix} p \\ \downarrow \\ n \times m \end{matrix} \quad \begin{matrix} \delta_c \\ \downarrow \\ 1 \times 1 \end{matrix} \right) & \text{(B.31)}
\end{array}$$

Re-writing (B.30) as shown in (B.32),

$$V^T \text{diag} (U (X - E^T P)) V \quad (\text{B.32})$$

From (B.32), the quadratic term (H) of the quadratic programming problem can be written as shown in (B.33),

$$H = \text{diag} (U (X - E^T P)) \quad (\text{B.33})$$

Expanding second term of B.27 in order to solve for the linear term (f) of the quadratic programming. Assuming $c = 1$ for making the derivation easy.

$$J = \sum_{d=1}^D \delta_{1d} v_{1d} \quad (\text{B.34})$$

or,

$$J = \delta_d [v_1 + v_2 + \dots + v_d] \quad (\text{B.35})$$

Weights are solved by using the MATLAB's *quadprog* for the quadratic programming problem. Initialization of the *quadprog* function is shown in (B.36),

Parameter Initialization for quadprog function:

$$f = [\delta_d]^T$$

$$H = diag \left(U \left(\mathbf{X} - \mathbf{E}^T \mathbf{P} \right) \right)$$

$$lb = zeros(D, 1)$$

$$ub = D * ones(D, 1)$$

$$Aeq = ones(1, D)$$

$$Beq = D \quad (B.36)$$

B.2.3 Endmembers

The update equation for endmembers is found by rewriting SSD term of (B.15) in matrix form as shown in (B.37),

$$J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd} \mathbf{p}_{cn})^2 + \alpha \sum_{d=1}^D v_{cd}^2 \left(M.trace(\mathbf{E}_{cd}^T \mathbf{E}_c d) - 1_{1 \times M} \mathbf{E}_{cd}^T \mathbf{E}_c d 1_{M \times 1} \right) + \sum_{d=1}^D \delta_{cd} |v_{cd}| \right) \quad (B.37)$$

The partial derivative of (B.37) with respect to E_{cd} as shown in (B.38),

$$\frac{\partial J}{\partial E_{cd}} = \sum_{n=1}^N u_{cn}^q v_{cd}^2 (2\mathbf{E}_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} - 2\mathbf{p}_{cn}^T x_{nd}) + 2v_{cd}\alpha (M I_{M \times M} - 1_{M \times M}) \mathbf{E}_{cd} \quad (B.38)$$

Substituting (B.38) to 0 and solving for \mathbf{E}_{cd} ,

$$\sum_{n=1}^N u_{cn}^q v_{cd}^2 (2\mathbf{E}_{cd}\mathbf{p}_{cn}^T \mathbf{p}_{cn} - 2\mathbf{p}_{cn}^T x_{nd}) + 2\alpha v_{cd}^2 (M I_{M \times M} - 1_{M \times M}) E_{cd} = 0$$

$$\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{E}_{cd} \mathbf{p}_{cn}^T \mathbf{p}_{cn} - \sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T x_{nd} + v_{cd}^2 \alpha (M I_{M \times M} - 1_{M \times M}) E_{cd} = 0$$

$$\mathbf{E}_{cd} \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T \mathbf{p}_{cn} + v_{cd}^2 \alpha (M I_{M \times M} - 1_{M \times M}) \right) = \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T x_{nd} \right)$$

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T \mathbf{p}_{cn} + v_{cd}^2 \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T x_{nd} \right)$$

The final update equation for \mathbf{E}_{cd} is shown in (B.39),

$$\mathbf{E}_{cd} = \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T \mathbf{p}_{cn} + \alpha (M I_{M \times M} - 1_{M \times M}) \right)^{-1} \left(\sum_{n=1}^N u_{cn}^q v_{cd}^2 \mathbf{p}_{cn}^T x_{nd} \right) \quad (\text{B.39})$$

B.2.4 Proportions

The update equation for Proportion is found by adding a Lagrange multiplier term to enforce the sum-to-one constraint and Karsuh-Kuhn-Tucker (KKT) conditions to enforce the non-negativity constraint given in (B.2),

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = & \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd}^2 (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^2 + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \right. \\
& \left. + \sum_{d=1}^D \delta_{cd} |\mathbf{v}_{cd}| \right) - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) \sum_{d=1}^D v_{cd}^2 - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T
\end{aligned} \tag{B.40}$$

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = & \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn})^T (x_{nd} - \mathbf{E}_{cd}\mathbf{p}_{cn}) \right. \\
& + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) \\
& \left. + \sum_{d=1}^D \delta_{cd} v_{cd} \right) - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) \sum_{d=1}^D v_{cd}^2 - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T
\end{aligned} \tag{B.41}$$

(B.42)

Equation (B.42) can be re-written as shown in (B.43),

$$\begin{aligned}
J(\mathbf{E}, \mathbf{P}, \mathbf{U}, \mathbf{V}) = & \sum_{c=1}^C \left(\sum_{n=1}^N u_{cn}^q \sum_{d=1}^D v_{cd} (x_{nd}^T x_{nd} - x_{nd}^T \mathbf{E}_{cd}\mathbf{p}_{cn} - \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T x_{nd} - \mathbf{E}_{cd}^T \mathbf{E}_{cd}\mathbf{p}_{cn}^T \mathbf{p}_{cn}) \right. \\
& + \alpha \sum_{m=1}^{M-1} \sum_{k=m+1}^M (\mathbf{e}_{cm} - \mathbf{e}_{ck})^T (\mathbf{e}_{cm} - \mathbf{e}_{ck}) + \sum_{\mathbf{d}=\mathbf{1}}^{\mathbf{D}} \delta_{\mathbf{cd}} \mathbf{v}_{\mathbf{cd}} \Big) \\
& - \sum_{c=1}^C \sum_{n=1}^N \lambda_c (1_{1 \times M} \mathbf{p}_{cn}^T - 1) \sum_{d=1}^D v_{cd}^2 - \sum_{c=1}^C \sum_{n=1}^N \gamma_{cn} \mathbf{p}_{cn}^T
\end{aligned} \tag{B.43}$$

The partial derivative of (B.43) with respect to \mathbf{p}_{cn}^T is shown in (B.44),

$$\frac{\partial J}{\partial p_{cn}^T} = -2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd}^2 \mathbf{E}_{cd}^T x_{nd} + 2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd}^2 \mathbf{E}_{cd} \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T - \lambda_c \mathbf{1}_{M \times 1} \sum_{d=1}^D v_{cd}^2 - \gamma_{cn} \quad (\text{B.44})$$

Substituting (B.44) to 0 and solving for p_{cn} ,

$$-2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd}^2 \mathbf{E}_{cd}^T \mathbf{x}_{nd} + 2\mathbf{u}_{cn}^q \sum_{d=1}^D v_{cd}^2 \mathbf{E}_{cd} \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T - \lambda_c \mathbf{1}_{M \times 1} \sum_{d=1}^D v_{cd}^2 - \gamma_{cn} = 0 \quad (\text{B.45})$$

Re-writing equation (B.45) by taking out v_{cd}^2 from the LHS of (B.45),

$$2\mathbf{u}_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \mathbf{p}_{cn}^T = \lambda_c \mathbf{1}_{M \times 1} \sum_{d=1}^D + \gamma_{cn} [v_{cd}^2]^{-1} + 2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} \quad (\text{B.46})$$

As λ_c is independent of number of dimensions (D), we can write (B.46) as shown in (B.47),

$$\mathbf{p}_{cn}^T = \left(2\mathbf{u}_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(\lambda_c \mathbf{1}_{M \times 1} D + \gamma_{cn} [v_{cd}^2]^{-1} + 2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} \right) \quad (\text{B.47})$$

As per KKT conditions $\gamma_{cn} \geq 0, \mathbf{p}_{cn}^T \geq 0$ and $\gamma_{cn} \mathbf{p}_{cn}^T = 0$. If $\gamma_{cn} > 0$, then $\mathbf{p}_{cn}^T = 0$. Thus γ_{cn} should be equal to zero and $\mathbf{p}_{cn}^T \geq 0$. Equation (B.47) can be written as shown in (B.48),

$$\mathbf{p}_{cn}^T = \left(2\mathbf{u}_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(D\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} \right) \quad (\text{B.48})$$

$$\mathbf{p}_{cn}^{KKT} = \max(\mathbf{p}_{cn}^T, 0) \quad (\text{B.49})$$

\mathbf{p}_{cn} can also be written as shown in (B.50),

$$\mathbf{1}_{1 \times M} \mathbf{p}_{cn}^T = 1 \quad (\text{B.50})$$

Substituting (B.48) in (B.50) and solving for λ_c

$$\begin{aligned} \mathbf{1}_{1 \times M} \left[\left(2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(D\lambda_c \mathbf{1}_{M \times 1} + 2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} \right) \right] &= 1 \\ \mathbf{1}_{1 \times M} \left[\left(2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} (D\lambda_c \mathbf{1}_{M \times 1}) + \left(2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \left(2u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} \right) \right] &= 1 \end{aligned}$$

$$\mathbf{1}_{1 \times M} \frac{\lambda_c}{2} \left(Du_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \mathbf{1}_{M \times 1} + \mathbf{1}_{1 \times M} \left(\sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} = 1$$

$$\mathbf{1}_{1 \times M} \left(\sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd} = 1 - \mathbf{1}_{1 \times M} \frac{\lambda_c}{2} \left(Du_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \mathbf{1}_{M \times 1}$$

Final equation for λ_c is shown in (B.2.4)

$$\lambda_c = 2 \frac{1 - 1_{1 \times M} \left(\sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} \sum_{d=1}^D \mathbf{E}_{cd}^T x_{nd}}{1_{1 \times M} \left(D u_{cn}^q \sum_{d=1}^D \mathbf{E}_{cd} \mathbf{E}_{cd}^T \right)^{-1} 1_{M \times 1}}$$

Bibliography

- [1] G. Vane, R. O. Green, T. G. Chrien, H. T. Enmark, E. G. Hansen, and W. M. Porter, “The airborne visible/infrared imaging spectrometer (aviris),” *Remote Sensing of Environment*, vol. 44, no. 2, pp. 127–143, 1993.
- [2] R. W. Basedow, D. C. Carmer, and M. E. Anderson, “Hydice system: Implementation and performance,” in *SPIE’s 1995 Symposium on OE/Aerospace Sensing and Dual Use Photonics*, pp. 258–267, International Society for Optics and Photonics, 1995.
- [3] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, and P. Gader, “Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, pp. 354–379, April 2012.
- [4] J.-P. Bibring, A. Soufflot, M. Berthé, Y. Langevin, B. Gondet, P. Drossart, M. Bouyé, M. Combes, P. Puget, A. Semery, *et al.*, “Omega: Observatoire pour la minéralogie, l’eau, les glaces et l’activité,” in *Mars Express: the scientific payload*, vol. 1240, pp. 37–49, 2004.
- [5] S. Douté, B. Schmitt, Y. Langevin, J. Bibring, F. Altieri, G. Bellucci, B. Gondet,

- and F. Poulet, “South pole of mars: Nature and composition of the icy terrains from mars express omega observations,” *Planetary and Space Science*, vol. 55, no. 1, pp. 113–133, 2007.
- [6] A. Gowen, C. O’Donnell, P. Cullen, G. Downey, and J. Frias, “Hyperspectral imaging—an emerging process analytical tool for food quality and safety control,” *Trends in Food Science & Technology*, vol. 18, no. 12, pp. 590–598, 2007.
- [7] D. Liu, X.-A. Zeng, and D.-W. Sun, “Recent developments and applications of hyperspectral imaging for quality evaluation of agricultural products: a review,” *Critical Reviews in Food Science and Nutrition*, no. just-accepted, 2013.
- [8] A. Meraoumia, S. Chitroub, and A. Bouridane, “An efficient palmprint identification system using multispectral and hyperspectral imaging,” in *Modeling Approaches and Algorithms for Advanced Computer Applications*, pp. 155–164, Springer, 2013.
- [9] M. J. Abrams and D. C. Comer, “Multispectral and hyperspectral technology and archaeological applications,” in *Mapping Archaeological Landscapes from Space*, pp. 57–71, Springer, 2013.
- [10] S. H. Savage, T. E. Levy, and I. W. Jones, “Archaeological remote sensing in jordans faynan copper mining district with hyperspectral imagery,” in *Mapping Archaeological Landscapes from Space*, pp. 97–110, Springer, 2013.
- [11] D. Thompson, R. Castano, and M. Gilmore, “Sparse superpixel unmixing for exploratory analysis of crism hyperspectral images,” in *Hyperspectral Image and*

Signal Processing: Evolution in Remote Sensing, 2009. WHISPERS '09. First Workshop on, pp. 1–4, 2009.

- [12] M. Parente, J. Bishop, and J. Bell III, “Spectral unmixing for mineral identification in pancam images of soils in gusev crater, mars,” *Icarus*, vol. 203, no. 2, pp. 421–436, 2009.
- [13] D. Goltz, M. Attas, G. Young, E. Cloutis, and M. Bedynski, “Assessing stains on historical documents using hyperspectral imaging,” *Journal of Cultural Heritage*, vol. 11, no. 1, pp. 19–26, 2010.
- [14] N. Keshava and J. F. Mustard, “Spectral unmixing,” *IEEE Signal Processing Magazine*, vol. 19, pp. 44–57, 2002.
- [15] R. B. Singer and T. B. McCord, “Mars-large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance,” in *Lunar and Planetary Science Conference Proceedings*, vol. 10, pp. 1835–1848, 1979.
- [16] S. De Backer, P. Kempeneers, W. Debruyn, and P. Scheunders, “A band selection technique for spectral classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 3, pp. 319–323, 2005.
- [17] C.-I. Chang, Q. Du, T.-L. Sun, and M. Althouse, “A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 6, pp. 2631–2641, 1999.
- [18] R. Huang and M. He, “Band selection based on feature weighting for classification

of hyperspectral data,” *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 2, pp. 156–159, 2005.

- [19] (2004, Sep) AVIRIS free standard data products. Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA. URL <http://aviris.jpl.nasa.gov/html/aviris.freedata.html>.
- [20] A. R. Webb, *Statistical pattern recognition*. Wiley. com, 2003.
- [21] A. Zare, *Hyperspectral Endmember Detection and Band Selection using Bayesian Methods*. PhD thesis, University of Florida, 2009.
- [22] J. W. Boardman, F. A. Kruse, and R. O. Green, “Mapping target signatures via partial unmixing of aviris data,” 1995.
- [23] M. E. Winter, “N-findr: an algorithm for fast autonomous spectral end-member determination in hyperspectral data,” in *SPIE’s International Symposium on Optical Science, Engineering, and Instrumentation*, pp. 266–275, International Society for Optics and Photonics, 1999.
- [24] R. Neville, K. Staenz, T. Szeregi, J. Lefebvre, and P. Hauff, “Automatic end-member extraction from hyperspectral data for mineral exploration,” in *Proc. 21st Canadian Symp. Remote Sens*, pp. 21–24, 1999.
- [25] J. M. Nascimento and J. M. Dias, “Vertex component analysis: A fast algorithm to unmix hyperspectral data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898–910, 2005.

- [26] F. Chaudhry, C.-C. Wu, W. Liu, C.-I. Chang, and A. Plaza, “Pixel purity index-based algorithms for endmember extraction from hyperspectral imagery,” *Recent Advances in Hyperspectral Signal and Image Processing*, pp. 29–62, 2006.
- [27] A. A. Green, M. Berman, P. Switzer, and M. D. Craig, “A transformation for ordering multispectral data in terms of image quality with implications for noise removal,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 26, no. 1, pp. 65–74, 1988.
- [28] C.-I. Chang and A. Plaza, “A fast iterative algorithm for implementation of pixel purity index,” *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 1, pp. 63–67, 2006.
- [29] C.-I. Chang, *Hyperspectral imaging: techniques for spectral detection and classification*. Springer, 2003.
- [30] *N-FINDR: an algorithm for fast autonomous spectral end-member determination in hyperspectral data*, vol. 3753, 1999.
- [31] M. Berman, H. Kiiveri, R. Lagerstrom, A. Ernst, R. Dunne, and J. F. Huntington, “Ice: A statistical approach to identifying endmembers in hyperspectral images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 10, pp. 2085–2095, 2004.
- [32] S. Wright and J. Nocedal, *Numerical optimization*, vol. 2. Springer New York, 1999.
- [33] A. Zare and P. Gader, “Sparsity promoting iterated constrained endmember de-

- tection in hyperspectral imagery,” *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 3, pp. 446–450, 2007.
- [34] M. Ehler and M. Hirn, “Sparse endmember extraction and demixing,” in *2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 1385–1388, July 2012.
- [35] J. Wang and C.-I. Chang, “Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 6, pp. 1586–1600, 2006.
- [36] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
- [37] J. C. Harsanyi and C.-I. Chang, “Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 4, pp. 779–785, 1994.
- [38] S. Venkataraman, L. M. Bruce, A. Cheriyadat, and A. Mathur, “Hyperspectral dimensionality reduction via localized discriminant bases,” in *IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS 2005.*, vol. 2, p. 1245, 2005.
- [39] M. He and S. Mei, “Dimension reduction by random projection for endmember extraction,” in *2010 the 5th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 2323–2327, IEEE, 2010.
- [40] S. Chen and D. Zhang, “Semisupervised dimensionality reduction with pairwise

constraints for hyperspectral image classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 2, pp. 369–373, 2011.

- [41] L. Qiao, S. Chen, and X. Tan, “Sparsity preserving projections with applications to face recognition,” *Pattern Recognition*, vol. 43, no. 1, pp. 331–341, 2010.
- [42] L. Zhang, Y. Zhong, B. Huang, J. Gong, and P. Li, “Dimensionality reduction based on clonal selection for hyperspectral imagery,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 12, pp. 4172–4186, 2007.
- [43] L. Bruzzone, F. Roli, and S. B. Serpico, “An extension of the jeffreys-matusita distance to multiclass cases for feature selection,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, no. 6, pp. 1318–1321, 1995.
- [44] S. G. Mallat, “A theory for multiresolution signal decomposition: the wavelet representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.
- [45] L. M. Bruce, C. H. Koger, and J. Li, “Dimensionality reduction of hyperspectral data using discrete wavelet transform feature extraction,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 10, pp. 2331–2338, 2002.
- [46] E. T. Gormus, N. Canagarajah, and A. Achim, “Dimensionality reduction of hyperspectral images with wavelet based empirical mode decomposition,” in *2011 18th IEEE International Conference on Image Processing (ICIP)*, pp. 1709–1712, IEEE, 2011.
- [47] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, “The empirical mode decomposition and the hilbert

spectrum for nonlinear and non-stationary time series analysis,” *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1971, pp. 903–995, 1998.

- [48] B.-C. Kuo and D. A. Landgrebe, “Nonparametric weighted feature extraction for classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 5, pp. 1096–1105, 2004.
- [49] C.-H. Li, B.-C. Kuo, C.-T. Lin, and C.-C. Hung, “Dimension reduction for hyperspectral image classification via support vector based feature extraction,” vol. 5, pp. V–389, IEEE, 2008.
- [50] A. Zare and P. Gader, “Hyperspectral band selection and endmember detection using sparsity promoting priors,” *IEEE Geoscience and Remote Sensing Letters*, vol. 5, pp. 256–260, Apr. 2008.
- [51] S. Stearns, B. Wilson, and J. Peterson, “Optimal band selection for dimensionality reduction of hyperspectral imagery,” in *JPL, Summaries of the 4 th Annual JPL Airborne Geoscience Workshop.*, vol. 1, 1993.
- [52] J. A. Richards, *Remote sensing digital image analysis: an introduction*. Springer, 2013.
- [53] C. Conese and F. Maselli, “Selection of optimum bands from tm scenes through mutual information analysis,” *ISPRS journal of photogrammetry and remote sensing*, vol. 48, no. 3, pp. 2–11, 1993.
- [54] C.-I. Chang, Q. Du, T.-L. Sun, and M. L. Althouse, “A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image

- classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 6, pp. 2631–2641, 1999.
- [55] N. Keshava, “Best bands selection for detection in hyperspectral processing,” in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01).*, vol. 5, pp. 3149–3152, IEEE, 2001.
- [56] N. Keshava, “Distance metrics and band selection in hyperspectral processing with applications to material identification and spectral libraries,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 7, pp. 1552–1565, 2004.
- [57] H. Du, H. Qi, X. Wang, R. Ramanath, and W. E. Snyder, “Band selection using independent component analysis for hyperspectral image processing,” in *Applied Imagery Pattern Recognition Workshop, 2003. Proceedings. 32nd*, pp. 93–98, IEEE, 2003.
- [58] S. B. Serpico and L. Bruzzone, “A new search algorithm for feature selection in hyperspectral remote sensing images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 7, pp. 1360–1367, 2001.
- [59] B. Guo, S. R. Gunn, R. Damper, and J. Nelson, “Band selection for hyperspectral image classification using mutual information,” *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 4, pp. 522–526, 2006.
- [60] A. Martínez-Usó, F. Pla, J. M. Sotoca, and P. García-Sevilla, “Clustering-based hyperspectral band selection using information measures,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 12, pp. 4158–4171, 2007.

- [61] J. H. Ward Jr, “Hierarchical grouping to optimize an objective function,” *Journal of the American statistical association*, vol. 58, no. 301, pp. 236–244, 1963.
- [62] Q. Du, “Band selection and its impact on target detection and classification in hyperspectral image analysis,” in *2003 IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, pp. 374–377, IEEE, 2003.
- [63] J. Harsanyi, W. Farrand, and C.-I. Chang, “Determining the number and identity of spectral endmembers: an integrated approach using neyman-pearsoneigen-thresholding and iterative constrained rms error minimization,” in *Proceedings of the Thematic Conference on Geologic Remote Sensing*, vol. 1, pp. 395–395, ENVIRONMENTAL RESEARCH INSTITUTE OF MICHIGAN, 1993.
- [64] H. Yang, Q. Du, H. Su, and Y. Sheng, “An efficient method for supervised hyperspectral band selection,” *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 1, pp. 138–142, 2011.
- [65] A. Zare and P. Gader, “PCE: Piece-wise convex endmember detection,” vol. 48, pp. 2620–2632, Jun. 2010.
- [66] A. Zare, P. Gader, and G. Casella, “Sampling piecewise convex unmixing and endmember extraction,” 2013.
- [67] O. Bchir, H. Frigui, A. Zare, and P. Gader, “Multiple model endmember detection based on spectral and spatial information,” in *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1–4, 2010.

- [68] A. Zare, P. Gader, O. Bchir, and H. Frigui, “Piecewise convex multiple-model endmember detection and spectral unmixing,” *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1 –10, 2012.
- [69] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*. Kluwer Academic Publishers, 1981.
- [70] R. Krishnapuram, O. Nasraoui, and H. Frigui, “The fuzzy c spherical shells algorithm: A new approach,” *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 663–671, 1992.
- [71] A. Zare, O. Bchir, H. Frigui, and P. Gader, “Spatially-smooth piece-wise convex endmember detection,” in *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1–4, IEEE, 2010.
- [72] A. Zare and P. Gader, “Piece-wise convex spatial-spectral unmixing of hyperspectral imagery using probabilistic and fuzzy clustering,” in *2011 IEEE International Conference on Fuzzy Systems (FUZZ)*, pp. 741 –746, june 2011.
- [73] A. Zare and P. Gader, “Piece-wise convex spatial-spectral unmixing of hyperspectral imagery using probabilistic and fuzzy clustering,” in *2011 IEEE International Conference on Fuzzy Systems (FUZZ)*, pp. 741–746, IEEE, 2011.
- [74] S. Krinidis and V. Chatzis, “A robust fuzzy local information c-means clustering algorithm,” *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp. 1328–1337, 2010.
- [75] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, “A probabilistic fuzzy c-

- means clustering algorithm,” *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.
- [76] M. Rifqi, “Constructing prototypes from large databases,” in *Proc. of IPMU96*, pp. 301–306, Citeseer, 1996.
- [77] D. T. Anderson and A. Zare, “Spectral unmixing cluster validity index for multiple sets of endmembers,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 4, pp. 1282–1295, 2012.
- [78] H. W. Kuhn and A. W. Tucker, “Nonlinear programming,” in *Proceedings of the second Berkeley symposium on mathematical statistics and probability*, vol. 5, California, 1951.
- [79] H. Frigui and O. Nasraoui, “Simultaneous clustering and attribute discrimination,” in *The Ninth IEEE International Conference on Fuzzy Systems, 2000. FUZZ IEEE 2000.*, vol. 1, pp. 158 –163 vol.1, may 2000.
- [80] P. M. Williams, “Bayesian regularization and pruning using a laplace prior,” *Neural computation*, vol. 7, no. 1, pp. 117–143, 1995.
- [81] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.
- [82] J. Denker, D. Schwartz, B. Wittner, S. Solla, R. Howard, L. Jackel, and J. Hopfield, “Large automatic learning, rule extraction, and generalization,” *Complex systems*, vol. 1, no. 5, pp. 877–922, 1987.

- [83] M. A. Figueiredo, “Adaptive sparseness for supervised learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1150–1159, 2003.
- [84] A. Zare and D. Anderson, “Earth movers distance-based simultaneous comparison of hyperspectral endmembers and proportions,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. PP, no. 99, pp. 1–12, 2013.
- [85] D. Landgrebe, “Aviris nw indianas indian pines 1992 data set,” 1992.
- [86] S. Holzwarth, A. Muller, M. Habermeyer, R. Richter, A. Hausold, S. Thiemann, and P. Strobl, “HySens - DAIS 7915/ ROSIS imaging spectrometers at DLR,” in *3rd EARSeL Workshop on Imaging Spectroscopy*, May 2003.
- [87] P. Gader, A. Zare, R. Close, J. Aitken, G. Tuell, . G. Hyperspectral, and L. A. D. Set, *University of Florida, Gainesville, FL*. Tech. Rep, Oct. 2013.
- [88] M. Graña and J. Gallego, “Associative morphological memories for spectral unmixing.,” in *ESANN*, pp. 481–486, 2003.

VITA

Piyush Pradeep Khopkar received his Bachelor of Engineering degree with Honor's certificate from the department of Electronics and Communication Engineering at the Mahakal Institute of Technology, Ujjain, India in 2009. After working for two years as a Software Engineer, he continued his graduate studies at the University of Missouri - Columbia with the guidance of Dr. Alina Zare. He graduated with Master of Science degree from the department of Electrical and Computer Engineering in 2014.