

## ABSTRACT

ROUSE, DAVID MARSHALL Estimation of Finite Mixture Models. (Under the direction of Professor H. Joel Trussell).

A recorded signal frequently results from the mixture of many signals from several classifiable sources. Knowledge of the contribution of the underlying sources to the recorded signal is valuable in several applications, such as remote sensing. Such mixtures may be analyzed using finite mixture models. Historically, finite mixture models decompose a density as the sum of a finite number of component densities. Current methods for estimating the contribution of each component assume a parametric form for the mixture components. Furthermore, these methods assume a collection of samples from the mixture are observed rather than an aggregate representation of the samples, such as a histogram.

This work introduces a method to address the many practical cases where parametric mixture models are insufficient to describe the mixture components. The observed mixture is assumed to occur in an aggregate representation of samples. Thus, the mixture components are represented as finite-length signals or vectors. The proposed method incorporates the first and second order statistics of the mixture components obtained from previously collected samples of the mixture components. The new method is based on the set theoretic method of successive projections onto convex sets (POCS). The set theoretic approach defines a set of feasible solutions as the intersection of sets consistent with the prior knowledge of a desirable solution. POCS is an iterative procedure used to find a point in the set of feasible solutions. This work considers several sets describing the finite mixture model, including a new model set generalizing a set based on the error-in-variables model.

To illustrate the viability of the new method, comparisons are made with the expectation-maximization (EM) algorithm for mixtures with parametric components. Simulations of mixture with nonparametric components emphasize the advantages of the new method, since no other methods address mixtures with nonparametric components. The new method is applied to the problem of resolving hyperspectral data representing the mixture of several component spectra.

**Estimation of Finite Mixture Models**

by

**David Marshall Rouse**

A thesis submitted to the Graduate Faculty of  
North Carolina State University  
in partial satisfaction of the  
requirements for the Degree of  
Master of Science

**Electrical Engineering**

Raleigh

2005

**Approved By:**

---

Dr. Wesley E. Snyder

---

Dr. Carl D. Meyer

---

Dr. H. Joel Trussell  
Chair of Advisory Committee

To my family, my friends, and my teachers . . .

## Biography

David M. Rouse was born in Waxhaw, North Carolina on March 6, 1981. He attend public schools in Union County North Carolina for his primary and secondary education. Noted interests in science and mathematics in high school encouraged him to pursue an undergraduate study of engineering with a focus in electrical engineering at North Carolina State University in the fall of 1999. After his sophomore year of study, he encountered the area of signal processing and proceeded to direct his concentration to that field. During the summer after his sophomore year, he worked exclusively with Professor H. Joel Trussell to compose over 100 pages of original supplementary notes for a new course introducing mathematics frequently encountered by electrical engineers. In the following summer, David began research to identify the proportion of internet applications across a network router from the distribution of packet sizes of an interval. This research was generalized to consider finite mixtures of probability densities with components described by samples. Applications to hyperspectral images were considered by his senior year of undergraduate study. Fascinated by this research, he chose to pursue a Master of Science degree in electrical engineering under the direction of Professor H. Joel Trussell in the fall of 2003. His master's research was a much more extensive study of the finite mixture problem posed after his junior year of undergraduate study. This thesis documents that research. In the summer of 2004, he instructed the junior-level linear signals and systems course offered by the electrical and computer engineering department at North Carolina State University. This experience inspired him to consider a future in academia. He finished his master's research during the summer of 2005 while he served as an intern at the Johns Hopkins University Applied Physics Laboratory in Laurel, Maryland. There he investigated numerical integration methods to approximate the Kullback-Liebler distance for probability densities defined by the linear mixture of a finite number of Gaussian densities. Upon conclusion of his master's study, David will begin as a doctoral candidate in electrical engineering at Cornell University in the fall of 2005.

## Acknowledgements

Any concise acknowledgement will unfortunately exclude people who certainly provided invaluable support throughout the time taken to compose this thesis. Nonetheless, several specific people deserve recognition for their assistance, and I would to take this occasion to acknowledge their contributions.

The Electrical and Computer Engineering Department of North Carolina State University arranged a meeting between Professor H. Joel Trussell and myself through a course he designed to introduce the mathematics necessary for electrical engineers. The seemingly arcane material of that course ultimately provided the fundamentals for my concentration in signal processing. Following that course, we explored a variety research projects. Several such research endeavors were shared with the research community via conference papers. I am ever grateful to have had Professor Trussell arrange my introduction to the research community.

As an advisor, Professor Trussell demonstrated innovative techniques for decomposing problems and elegantly eradicating ostensibly insurmountable obstructions encountered in my research. In time, I observed that his guidance ingeniously imparted these techniques as I found myself successfully exploring new ideas with less direct supervision. This thesis documents the exploration of a vision presented by my advisor and friend, Professor Trussell. In addition, its existence attests to my attainment of many of his masterful research techniques. Professor Trussells contributions deserve acknowledgements far exceeding what I could attempt to express. Needless to say, his involvement in my education is immensely appreciated.

Another professor of the Electrical and Computer Engineering Department bestowed clever research perspectives during my graduate study and deserves recognition. Professor Wesley Snyder conducted insightful lectures that presented a deep understanding of an unknown method that simplified its implementation, interpretation, and analysis. Every lecture instilled a clear perception of a complex method. It is my hope that this thesis illustrates that I have learned from his gift to comprehend and communicate complex ideas present in all of his lectures and discussions. I thank Professor Snyder for his role in my graduate education.

Professor Carl Meyer of the Mathematics Department provided a formidable understanding of matrix theory, which became an integral tool to the development of this

thesis. Once Professor Meyer indicated his unfamiliarity with a topic in this thesis. This was astonishing and ironic, for he educated me in the necessary fundamentals to comprehend that topic. I greatly appreciate Professor Meyers involvement in my mathematical development, which allowed me to tackle the mathematical concepts in this thesis.

My brother, Jerry, offered immeasurable support throughout my education. Eight years older than me, he has shared his encounters with mathematics for as long as I can remember. I remember several car trips made memorable due to fascinating conversations about mathematics from his classes. During my graduate study at NCSU, Jerrys recent experience with writing a masters thesis proved invaluable as unforeseen hurdles emerged. His advice provided the motivation to overcome the difficult times. This experience constantly reminded me of my appreciation for my brother, Jerry.

Finally, none of this would have been possible without my mother, Cheryl, and my father, Marshall. Upon reflection, it is obvious that my parents share the responsibility for the qualities and characteristics that have led to my success. My father taught me to analyze and study the world with the perspective of an engineer. My mother exemplified the determination necessary to successfully achieve my ambitions. The culmination of these two gifts along with many others from my parents permitted my excellence in education. With this opportunity, I would like to remind my parents of my appreciation for their countless contributions, everlasting support, and unconditional love.

# Contents

<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>x</b>
<b>List of Symbols</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Finite Mixture Models . . . . .	4
1.1.1 Finite Mixtures of Probability Densities . . . . .	5
1.1.2 Finite Mixtures of Spectral Densities . . . . .	6
1.2 Definition of the Problem . . . . .	11
<b>2 Theoretical Background</b>	<b>14</b>
2.1 Maximum Likelihood and Maximum A Posteriori . . . . .	14
2.2 Maximum Likelihood Estimation and Incomplete Data . . . . .	16
2.3 Previous Methods for Finite Mixture Models with Parametric Components	17
2.3.1 EM Algorithm . . . . .	17
2.3.2 Unsupervised Learning of Finite Mixture Models . . . . .	21
2.4 Approximations of Nonparametric Probability Densities . . . . .	22
2.4.1 General Approximations . . . . .	23
2.4.2 Histograms . . . . .	25
<b>3 Estimation Methods</b>	<b>27</b>
3.1 Least Squares . . . . .	28
3.1.1 Least Squares Solution via the Normal Equations . . . . .	28
3.1.2 Least Squares Solution via the SVD . . . . .	30
3.1.3 Remarks . . . . .	30
3.2 Total Least Squares . . . . .	31
3.2.1 The Total Least Squares Problem . . . . .	31
3.2.2 TLS solution via the SVD . . . . .	33
3.2.3 Closed Form TLS Solution . . . . .	34
3.2.4 Relationship Between the TLS and LS Solutions . . . . .	36

3.2.5	TLS solution with Weighting Matrices . . . . .	36
3.2.6	Remarks . . . . .	37
3.3	Set Theoretic Estimation . . . . .	37
3.3.1	Method Overview . . . . .	38
3.3.2	Closed and Convex Sets . . . . .	40
3.3.3	Projection Methods . . . . .	41
3.3.4	Effect of Initial Estimate and the Order of Projection . . . . .	43
<b>4</b>	<b>Estimation of Finite Mixture Models with Nonparametric Components</b>	<b>45</b>
4.1	Sets for Finite Mixture Density Estimation . . . . .	46
4.1.1	Finite Mixture Model Set . . . . .	46
4.1.2	Sets Describing Properties of Feasible Contribution Estimates . . . . .	51
4.2	Projection onto Sets . . . . .	52
<b>5</b>	<b>Results</b>	<b>55</b>
5.1	Simulations of Finite Mixtures of Probability Densities . . . . .	58
5.1.1	Finite Mixtures of Parametric Densities . . . . .	60
5.1.2	Finite Mixtures of Nonparametric Densities . . . . .	71
5.1.3	Discussion . . . . .	77
5.2	Simulations of Finite Mixtures of Spectral Densities . . . . .	78
5.2.1	Target Contribution Simulation . . . . .	84
5.2.2	Target Detection with a Simulated Hyperspectral Image . . . . .	93
5.2.3	Discussion . . . . .	105
<b>6</b>	<b>Conclusions</b>	<b>107</b>
<b>A</b>	<b>Generation of Color Images from Hyperspectral Data</b>	<b>110</b>
<b>B</b>	<b>Derivations for the Error-in-Variables POCS Method</b>	<b>116</b>
B.1	Equivalence of $S_{TLS}$ and $\Gamma$ sets . . . . .	116
B.2	Derivation of $\tau$ for EVPOCS . . . . .	119
	<b>Bibliography</b>	<b>121</b>

# List of Figures

1.1	Illustration of image capture by low spatial resolution digital camera . . . .	8
3.1	Illustration of Kaczmarz's Projection Method . . . . .	42
3.2	Illustration of the Effect of Relaxation Parameter $\lambda$ on Projections . . . . .	42
3.3	Illustration of Cimmino's Projection Method . . . . .	43
3.4	Illustration of the Effect of Initial Estimate and the Order of Projection . .	44
5.1	Actual class probability densities used for simulations of parametric mixture densities. . . . .	60
5.2	Discrete estimates of class probability densities for the simulations of parametric mixture densities. The estimates were formed from $H_k = 10$ sample histograms, where $N = 100$ samples were used to form each histogram. . .	61
5.3	Sample discrete mixture density obtained from $KN = 300$ samples. Refer to Table 5.1 for estimates of the class proportions. . . . .	65
5.4	Comparison of observed mixture $\mathbf{r}$ from Figure 5.3 and the reconstructed mixture based on the standard and modified EM algorithm estimates in Table 5.1. The EM algorithm estimates were substituted into Eq. (1.5) to produce reconstructed mixture density. . . . .	67
5.5	Comparison of observed mixture $\mathbf{r}$ from Figure 5.3 and the reconstructed mixture based on the DPOCS (ls)* estimate in Table 5.1. $\bar{\mathbf{S}}$ is used reconstruct the mixture for the DPOCS (ls)* estimate. . . . .	69
5.6	Actual class probability densities for the simulations of nonparametric mixture densities. . . . .	73
5.7	Discrete estimates of component probability densities for the simulations of nonparametric mixture densities. The estimates were formed from 10 sample histograms, where $S_{\Omega_k} = 100$ samples were used to form each histogram. .	74
5.8	Sample discrete mixture density obtained from $KN = 300$ samples. Refer to Table 5.5 for estimates of class proportions. . . . .	75
5.9	Class Basis Spectra for the First Set of Spectral Signatures . . . . .	80
5.10	Estimates of the Class Spectral Signatures . . . . .	81
5.11	Modified Class Basis Spectra for the First Set of Spectral Signatures . . . .	83
5.12	Estimates of the Class Spectral Signatures . . . . .	84

5.13	Actual target contributions for the target contribution simulation . . . . .	85
5.14	Errors with respect to the Target Proportion Using Class 1 as Target Class	87
5.15	Errors with respect to the Target Proportion Using Class 2 as Target Class	88
5.16	Errors with respect to the Target Proportion Using Class 3 as Target Class	90
5.17	Errors with respect to the Target Proportion Using Class 4 as Target Class	92
5.18	Comparison of the original image and the observed image. The observed image is obtained from the original image by blurring, subsampling, and adding noise to achieve an SNR of 30 dB. This image was constructed with the modified set of basis spectra. . . . .	95
5.19	Actual contributions of class 4 (target) . . . . .	96
5.20	LS solution of class 4 (target) contributions . . . . .	97
5.21	TLS solution of class 4 (target) contributions . . . . .	98
5.22	SPOCS (ls) estimate of class 4 (target) contributions . . . . .	99
5.23	DPOCS (tls)* estimate of class 4 (target) contributions . . . . .	100
5.24	EVPOCS (tls) estimate of class 4 (target) contributions . . . . .	101
5.25	GEVPOCS (tls) estimate of class 4 (target) contributions . . . . .	102
5.26	Side-by-side (pixel-by-pixel) comparison of EVPOCS (tls) and GEVPOCS (tls) Absolute Errors. The columns of Figures 5.24(b) and 5.25(b) are interleaved beginning with the pixels corresponding to the EVPOCS (tls) errors in the far left column. . . . .	103
A.1	Power Spectral Distribution of Illuminants A, D65, and F2 in the Visible Spectrum . . . . .	111
A.2	Comparison of illuminants applied to hyperspectral image used for simulations	112
A.3	Components of the CIE XYZ color matching functions . . . . .	114

# List of Tables

5.1	Estimates of Class Proportions for Sample Mixture of Gaussian Probability Densities in Figure 5.3 . . . . .	64
5.2	Results for EM Algorithm for Gaussian Mixture Density in Figure 5.3 . . . . .	67
5.3	Comparison of the Number of Iterations and the Average Elapsed Time for the Estimation Methods for Sample Mixture of Gaussian Probability Densities in Figure 5.3 . . . . .	70
5.4	Results for Finite Mixtures with Gaussian Probability Densities from 1000 Simulations . . . . .	71
5.5	Estimates of Component Proportions for Sample Mixture of Nonparametric Probability Densities in Figure 5.8 . . . . .	75
5.6	Comparison of the Number of Iterations and the Average Elapsed Time for the Estimation Methods for Sample Mixture of Nonparametric Probability Densities in Figure 5.8 . . . . .	76
5.7	Results for Finite Mixtures with Nonparametric Components from 1000 Simulations . . . . .	77
5.8	Comparison of the Representation of Spectral Classes by the other Classes . . . . .	89
5.9	Iteration Results for Set Theoretic Methods Estimates for Target Contribution Simulations . . . . .	91
5.10	Overall Results for Hyperspectral Mixtures from 225 Pixels Using Modified Basis Set of Spectral Densities . . . . .	104
5.11	Average Absolute MSE (decibels) of Individual Class Estimates for Hyperspectral Mixtures from 225 Pixels Using Modified Basis Set of Spectral Densities	104
5.12	False Alarms Comparisons for POCS Estimates of Hyperspectral Mixtures . . . . .	105

# List of Symbols

$a_k$ , contribution of class $k$ to finite mixture density .....	5
$\hat{a}_k$ , estimate of $a_k$ .....	18
$\mathbf{a}$ , $K \times 1$ vector denoting contributions of $K$ classes in mixture density .....	3
$\hat{a}_k^{(i)}$ , estimate of $a_k$ at iteration $i$ .....	19
$\hat{\mathbf{a}}$ , estimate of contributions of $K$ classes in mixture density $\mathbf{r}$ .....	4
$\mathbf{A}^T$ , transpose of the matrix $\mathbf{A} \in \mathbb{C}^{M \times K}$ .....	28
$\mathbf{A}^*$ , Hermitian transpose of the matrix $\mathbf{A} \in \mathbb{C}^{M \times K}$ .....	34
$\mathbf{b}_c$ , $M \times 1$ vector denoting bin centers for each element of $\mathbf{s}_k$ .....	25
$\mathbf{b}_w$ , $M \times 1$ vector denoting interval widths for each element of $\mathbf{s}_k$ .....	11
$\delta_a$ , approximation of $\ \mathbf{a}\ _2^2$ .....	48
$\delta_\eta$ , estimate of the variance of the Euclidean norm of the residual $\boldsymbol{\eta}$ .....	46
$\mathbf{D}_\theta[\cdot]$ , differential operator with respect to the vector $\boldsymbol{\theta}$ .....	15
$E\{\cdot\}$ , expected value operator .....	12
$f(x)$ , probability density function .....	5
$f_k(x)$ , probability density of class $k$ .....	5
$f_{\mathcal{N}}(x; \mu, \sigma)$ , Gaussian density with mean $\mu$ and standard deviation $\sigma$ .....	19
$f_e(x; \lambda)$ , exponential density with mean $\frac{1}{\lambda}$ .....	72
$f_R(x; \omega)$ , Rayleigh density with mean $\sqrt{\frac{\pi}{2}}\omega$ .....	72
$\Gamma$ , equivalent set to error-in-variables set $S_{TLS}$ , also referenced by EVPOCS .....	49
$\Gamma_g$ , generalized error-in-variables set, also referenced by GEVPOCS .....	51
$\boldsymbol{\eta}$ , $M \times 1$ vector denoting measurement uncertainty .....	12
$H_k$ , number of sample histograms formed from samples in set $\Omega_k$ .....	25
$\mathbf{I}_M$ , identity matrix of order $M$ .....	28

$\boldsymbol{\theta}$ , parameters of a probability density function .....	5
$\hat{\boldsymbol{\theta}}_{map}$ , maximum a posteriori estimate of probability density parameters $\boldsymbol{\theta}$ .....	15
$\hat{\boldsymbol{\theta}}_{ml}$ , maximum likelihood estimate of probability density parameters $\boldsymbol{\theta}$ .....	16
$\hat{\boldsymbol{\theta}}^{(i)}$ , estimate of $\boldsymbol{\theta}$ at iteration $i$ .....	18
$K$ , number of classes in finite mixture density .....	3
$L(\boldsymbol{\theta})$ , likelihood function .....	15
$\mu$ , mean of a Gaussian random variable .....	19
$\hat{\mu}_k^{(i)}$ , estimate of $\mu$ for class $k$ at iteration $i$ .....	19
$M$ , number of elements to approximate class densities .....	3
$\nu$ , parameter defined for error-in-variables set .....	49
$N$ , number of scalar samples observed .....	12
$N_k$ , number of scalar samples from class $k$ .....	12
$N(\mathbf{A})$ , nullspace of the matrix $\mathbf{A}$ .....	29
$\boldsymbol{\xi}_k$ , parameter of probability density function describing class $k$ .....	5
$\hat{\boldsymbol{\xi}}_k$ , estimate of $\boldsymbol{\xi}_k$ .....	18
$\hat{\boldsymbol{\xi}}_k^{(i)}$ , estimate of $\boldsymbol{\xi}_k$ at iteration $i$ .....	19
$\rho$ , approximation of $E\ \Delta\mathbf{S}\ _2^2$ .....	47
$\mathbf{r}$ , $M \times 1$ vector denoting observed mixture density with $K$ classes .....	3
$R(\mathbf{A})$ , range space of the matrix $\mathbf{A}$ .....	28
$\mathbb{R}^K$ , vector space of real $K$ -vectors .....	13
$\mathbf{s}_k$ , $M \times 1$ vector denoting a discrete approximation of $f_k(x)$ .....	11
$\bar{\mathbf{s}}_k$ , $k^{th}$ column of $\bar{\mathbf{S}}$ .....	12
$\Delta\mathbf{s}_k$ , perturbation accounting for error in estimate $\bar{\mathbf{s}}_k$ for class $k$ .....	12
$\sigma$ , standard deviation of a Gaussian random variable .....	19
$\hat{\sigma}_k^{(i)}$ , estimate of $\sigma$ for class $k$ at iteration $i$ .....	19
$\sigma_K(\bar{\mathbf{S}})$ , smallest singular value of the $M \times K$ matrix $\bar{\mathbf{S}}$ .....	49
$S_{\Omega_k}$ , number of samples in set $\Omega_k$ .....	25
$S_v$ , residual variance set motivated by LS optimization, also referenced by SPOCS .....	46
$S_a$ , nonnegativity and sum-to-one constraint .....	13
$S_\Sigma$ , set defining sum-to-one constraint .....	51
$S_k$ , set defining nonnegativity of $k^{th}$ element of $\mathbf{a}$ .....	52
$S_{v'}$ , dynamic residual set, also referenced by DPOCS .....	48
$S_{TLS}$ , error-in-variables set based on TLS optimization .....	49

$\bar{\mathbf{S}}$ , $M \times K$ matrix denoting estimates of class densities .....	3
$\Sigma_{\eta}$ , $M \times M$ covariance matrix for uncertainty $\boldsymbol{\eta}$ .....	12
$\tau$ , parameter defined for error-in-variables set .....	49
$\tau_k(x; \boldsymbol{\theta})$ , posterior probability that $x$ belongs to the $k^{th}$ class given $\boldsymbol{\theta}$ .....	16
$x_n$ , $n^{th}$ scalar sample in the set $\mathcal{X}$ .....	14
$X$ , scalar random variable with probability density $f(x)$ .....	5
$X_k$ , scalar random variable with probability density function $f_k(x)$ .....	11
$X_k^{(n)}$ , $n^{th}$ scalar sample from class $k$ .....	25
$\mathcal{X}$ , set of $N$ observed scalar samples .....	12
$\Omega_k$ , sample data set of scalars for class $k$ .....	25

# Chapter 1

## Introduction

A recorded signal frequently contains a mixture of many signals from several classifiable sources. Various applications could capitalize upon knowledge of the contribution of the underlying sources to the recorded signal. For instance, spatial resolution limitations in remote sensing images inevitably produce pixels composed of various proportions of several spectral classes of ground cover [1]. A target, an entity of interest, ordinarily appears within a few pixels, mixed with its surroundings. One may detect the target's presence, or contribution, in those pixels using estimates of the target's spectral density. Accurate estimates of the target's contribution reduce the dependency of target detection from geometric features, which requires a spatial resolution sufficient to resolve the target. Detecting a target's presence using its spectral densities also eradicates the need for an elaborate database of the various geometric orientations of the target. Furthermore, detection algorithms relying on geometric features become increasingly complex when the target is partially occluded. The observed spectral density for a given pixel may contain both the target spectral density and the obstructing object's spectral density. With knowledge of the spectral density characteristics of both the target and the obstructing object, one may decompose or "unmix" the observed spectral density according to the contribution of the two sources.

Another example when a composite of several signals is observed occurs in the analysis of network traffic across a particular node. Several internet applications have been shown to exhibit characteristic frequency distributions of packet sizes [2]. The overall

frequency distribution of packet sizes across a network node on a particular time interval is the mixture of the distributions of individual applications. With an accurate estimate of the proportion of each application sent through the node, routers could give precedence to applications with time sensitive packets such as streaming media to improve the quality of service (QoS). Alternatively, unusual proportions of network traffic at a node could indicate a potential security breach.

These and many other applications may be analyzed using finite mixture models [3, 4, 5]. Finite mixture models decompose a density<sup>1</sup> function into component density functions, where each component describes a particular class. However, the current estimation of finite mixture models assumes the mixture component density functions have a classical parametric form (i.e. Gaussian, Poisson, etc.). Algorithms estimate both the parameters of these distributions, as well as the contribution of each class in the mixture. Historically, estimates of the parameters of the class probability densities in mixture densities have been found via the expectation-maximization (EM) algorithm [6]. The EM algorithm essentially finds a maximum likelihood estimate of the parameters of the class probability densities that produced a collection of observations. This method assumes that a few parameters completely define the underlying class probability densities. In addition, this method surmises knowledge of the number of classes. More recent algorithms also estimate the number of classes [7, 8]. Obviously, practical applications involving mixture densities, such as those described, may not have classes with classical parametric probability densities. Thus, applications with classes described by nonparametric probability densities clearly need a reliable method to decompose mixture densities.

Nonparametric probability densities must be approximated from sampled data. A histogram approximates a probability density according to the frequency distribution of sampled data with respect to predefined intervals, or bins. The histogram approximation is discrete. On the other hand, Parzen windows [9, 10] are used to provide a continuous approximation of a density from classified sample data. Approximations of the classes of interest could be formed with Parzen windows, but a large database of the classified samples is necessary to produce accurate approximations. Furthermore, the observation may not be a single sample but a collection of samples in an aggregate form, such as a histogram. In

---

<sup>1</sup>We shall consider a density as either a probability density or a spectral density. A probability density has the additional constraint of summing or integrating to one. No such constraints are required for a spectral density.

such cases, a continuous approximation would be inappropriate.

For our problem, the collection of observations is available only in an aggregate form. Consider the target detection problem using spectral densities. The spectral density of an object is given by the electromagnetic radiation of that object as a function of wavelength, or frequency. One may consider the energy detected at a particular wavelength as the random variable, and for several wavelengths, a random vector is appropriate. An approximation of the spectral density is measured by digital hyperspectral imaging cameras. A pixel in the hyperspectral image contains spectral data for up to several hundred contiguous frequency bands. Thus, a sample is a finite-length vector where each element corresponds to the energy over a specific frequency band. Similarly, the network traffic problem considers the frequency distribution with respect to specific packet sizes, for the size of a packet is quantized by design. Here the sample is a discrete random vector where the elements correspond to the number of packets of certain size(s). In both applications, the goal is to determine the proportion of each class from a representation of the accumulation of samples.

The aggregate form for our problem results from the accumulation of samples over an interval of time. It is reasonable to represent the collection of observed samples as a histogram. Histograms constitute the discrete version of Parzen windows using a rectangular window function. The component densities believed to contribute to the observation must also be approximated by histograms. All of the histograms must correspond to the same set of bins. By considering the uncertainty in the component density approximations, one can employ the approximations to estimate the proportion of each component in the observed mixture.

This work considers finite mixture models whose underlying class probability densities have been approximated by finite-length discrete signals. Suppose the columns of the  $M \times K$  matrix  $\mathbf{S}$  represent the  $M$ -element approximation of  $K$  class probability densities. Then, a finite mixture defined by the column vector  $\mathbf{r}$  is given by

$$\mathbf{r} = \mathbf{S}\mathbf{a}, \tag{1.1}$$

where the  $k^{th}$  element of the column vector  $\mathbf{a}$  corresponds to the contribution of the  $k^{th}$  class. Classical methods such as least squares (LS) or total least squares (TLS) produce insufficient estimates of  $\mathbf{a}$ , because the components of  $\mathbf{a}$  must possess certain properties. Namely, the vector describing the contribution of each class must be nonnegative and sum-

to-one. Therefore, the contribution of each class in an observed finite mixture signal is estimated via a set theoretic approach [11, 12]. Set theoretic estimation produces a feasible estimate, which satisfies the *a priori* knowledge of the system. In addition to satisfying Eq. (1.1), the feasible estimate  $\hat{\mathbf{a}}$  also must be nonnegative and sum-to-one.

Before proceeding with a formal description of the problem, a brief review of finite mixture models will illuminate the fundamentals of the finite mixture problem.

## 1.1 Finite Mixture Models

Finite mixture models historically have been analyzed as the composition of parametric component probability densities [3, 5, 4]. The earliest account of finite mixture models, according to [4], is attributed to a study of the frequency distribution of measurements of the carapace of 2000 female shore crabs. Half of the of the samples were obtained from crabs at Plymouth Sound, and the remaining half were acquired at the Bay of Naples [13]. Weldon, a biologist, collected the samples to explore “Galton’s function,” which purports that the ratio of the measurements between pairs of organs of a particular family is a constant [14]. From the data collected, Weldon observed that the measurements of the frontal breadth of the shore crabs at the Bay of Naples generated an asymmetric frequency distribution. Weldon sought the aide of Karl Pearson to investigate the supposition that the apparent asymmetry in the frequency distribution implied the existence of two species of crabs in the samples. Pearson generalized this phenomenon stating that “asymmetry may arise from the fact that the units grouped together in the measured material are not really homogenous” [15]. Accepting the task, Pearson searched for the most likely set of components given the samples. He employed the method of moments to fit a two component Gaussian mixture density to Weldon’s data set. This led to calculating the roots of a ninth order polynomial. In the end, Pearson’s calculations supported Weldon’s supposition that a two component Gaussian mixture density fits the data. Modern methods to estimate the parameters of the component probability densities do not rely on the formidable method of moments approach Pearson presented.

This section considers two applications of finite mixture models. First, a description of finite mixtures of probability densities captures the general aspects of such models. Then, the application of finite mixture models to the hyperspectral target estimation prob-

lem is presented.

### 1.1.1 Finite Mixtures of Probability Densities

For a random variable<sup>2</sup>  $X$ , finite mixture models decompose a probability density function  $f(x)$  into the sum of  $K$  class probability density functions. A general density function  $f(x)$  is considered semiparametric, since it may be decomposed into  $K$  components. Let  $f_k(x)$  denote the  $k^{\text{th}}$  class probability density function. The finite mixture model with  $K$  components expands as

$$f(x) = \sum_{k=1}^K a_k f_k(x), \quad (1.2)$$

where  $a_k$  denotes the proportion of the  $k^{\text{th}}$  class. The proportion  $a_k$  may be interpreted as the prior probability of observing a sample from class  $k$ . Furthermore, the prior probabilities  $a_k$  for each distribution must be nonnegative and sum-to-one, or

$$a_k \geq 0 \quad \text{for } k = 1, \dots, K, \quad (1.3)$$

where

$$\sum_{k=1}^K a_k = 1. \quad (1.4)$$

In this work, the objective is to determine the parameters of the class probability densities as well as the proportion  $a_k$  of each class to the overall mixture from a collection of  $N$  samples drawn according to the probability density  $f(x)$ . Consider the following formulation to describe finite mixture models.

Let  $X$  be a scalar random variable with a probability density function  $f(x; \boldsymbol{\theta})$ , parameterized by the elements in the vector  $\boldsymbol{\theta}$ . Thus,  $x$  is a realization of  $X$ . The random variable  $X$  is an outcome of a  $K$  class finite mixture distribution if its probability density function is written as

$$f(x; \boldsymbol{\theta}) = \sum_{k=1}^K a_k f_k(x; \boldsymbol{\xi}_k), \quad (1.5)$$

where  $a_k$  is the proportion of the  $k^{\text{th}}$  class with probability density function  $f_k(x; \boldsymbol{\xi}_k)$  with parameters  $\boldsymbol{\xi}_k$ . As described above, the objective is to determine the parameters  $\boldsymbol{\theta} = [a_1, a_2, \dots, a_K, \boldsymbol{\xi}_1^T, \boldsymbol{\xi}_2^T, \dots, \boldsymbol{\xi}_K^T]^T$  from a collection of samples. A well-known algorithm to

---

<sup>2</sup>This discussion could be extended to random vectors, but for our problem it suffices to consider a random scalar.

determine the parameters is the EM algorithm [6]. This algorithm may be constructed to find either the maximum likelihood (ML) or maximum a posteriori (MAP) estimate of the parameters. Section 2.3.1 presents an overview of this algorithm.

When a probability density cannot be uniquely defined by parameters, then the probability density function is considered nonparametric. For the case of nonparametric class probability densities, the parameters  $\xi_k$  in Eq. (1.5) must be relinquished, and one must revert to the expansion given in Eq. (1.2). The class densities  $f_k(x)$  may have any form, and the objective is only to determine the proportion of each class. However, to determine the proportion of each class, in the very least, an approximation of  $f_k(x)$  is necessary.

A histogram provides an approximation to a nonparametric probability density function. The continuous space  $\mathbb{R}$  is partitioned into contiguous regions, or bins. The histogram is defined by counting the number of samples that lie within a particular bin. The histogram may be normalized according to the bin widths such that the histogram sums to one like a probability density. The arithmetic average of a collection of such normalized histograms provides an approximation to the actual probability density function. Given a normalized histogram of new samples belonging to a  $K$  class finite mixture density, the contribution  $a_k$  of the each class is to be determined using the normalized histogram approximation of the nonparametric density for each class.

### 1.1.2 Finite Mixtures of Spectral Densities

The availability of high spectral resolution sensors permits the characterization of a material by its spectral signature rather than geometric features [1]. The molecular composition and shape of a material characterizes its reflected, absorbed, and/or emitted electromagnetic radiation [16]. Across a specific range of the electromagnetic spectrum, this radiation is measured by a sensor resulting in a spectral density as a function of wavelength  $\lambda$ . In most cases, materials reflect and absorb radiation emitted by an illumination source, e.g. the sun. Thus, the spectral density of interest is a reflectance spectrum that is characteristic of the object. This characteristic spectral density is also called the spectral signature of that object. Currently, hyperspectral pixel data is analyzed to detect the presence of a spectral signature mixed with other spectra.

Imaging devices capturing pixels containing more than 10 spectral bands are called

hyperspectral pixels. For example, sampling the visible spectral range (400 – 700nm) at a 10nm resolution leads to 31 spectral bands. Current sensors can record the electromagnetic radiation present at a spectral resolution less than 2nm in the visible and infrared spectral range.

Satellite-based remote sensing cameras capture digital images of the earth’s surface by recording the electromagnetic radiation over certain wavelengths. Due to the limited spatial resolution of these cameras and/or the distance of the observed scene, the images invariably contain pixels composed of several materials. It is desirable to resolve the contributions of the constituents from the observed image without relying on high spatial resolution images. Remote sensing cameras have been designed to capture a wide spectral range motivating the use of post-processing techniques to distinguish materials via their spectral signatures.

To illustrate the formation of a mixture of spectral signatures, consider the image in Figure 1.1(a). Let this image represent the true scene to be digitally captured by a camera, i.e. a remote sensor, at a great distance. The camera may record the image shown in Figure 1.1(b). This system may be described as the blurring and sub-sampling of the original image in Figure 1.1(a). Additive noise may be present in the recording sensor as well. For hyperspectral pixels, the blurring process mixes the spectral densities of neighboring objects. This results in hyperspectral pixels composed of several spectral signatures. In the literature, some authors have adopted the term *mixel* to describe a mixed pixel [17].

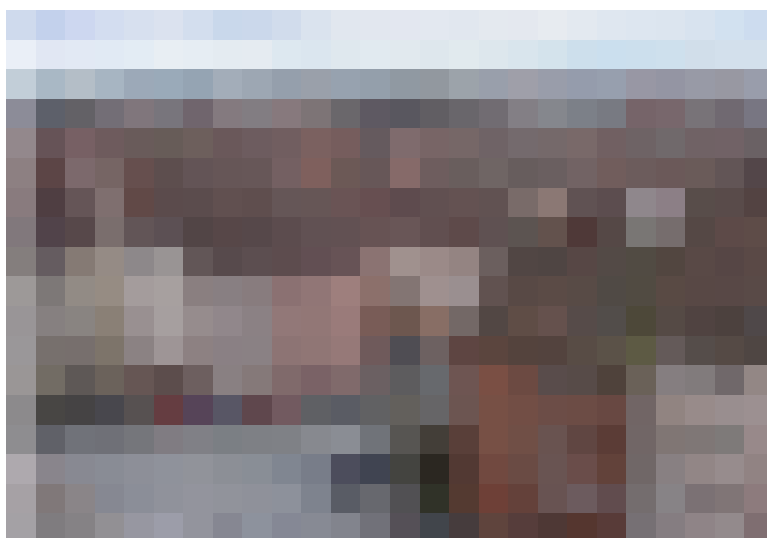
It is desirable to determine the contributions of spectral signatures in a mixture. This may be determined with estimates of the constituent spectral signatures. Suppose there are up to  $K$  unique spectral signatures known to contribute to an observed hyperspectral pixel. Let  $f(\lambda)$  denote the spectral density at wavelength  $\lambda$  of an observed hyperspectral pixel. The spectral density  $f(\lambda)$  may be decomposed according to the  $K$  spectral signatures  $f_k(\lambda)$  according to the finite mixture model

$$f(\lambda) = \sum_{k=1}^K a_k f_k(\lambda), \quad (1.6)$$

where  $a_k$  is the contribution of the  $k^{th}$  spectral signature. Several names are found in the literature to describe  $a_k$  including proportion, abundance, and amount. For finite mixtures of spectral densities,  $a_k$  will refer to the contribution of the  $k^{th}$  spectral signature.



(a) Original Scene



(b) Image Captured

Figure 1.1: Illustration of image capture by low spatial resolution digital camera

Hyperspectral sensors usually record spectral data of up to several hundred contiguous bands. The spectral data recorded by the hyperspectral sensor approximates the actual spectral density of the sensed object. An  $M$ -band hyperspectral pixel is denoted by the  $M \times 1$  column vector  $\mathbf{r}$ . From an  $M$ -band hyperspectral pixel  $\mathbf{r}$ , the contributions of the  $K$  spectral classes remain to be determined. Writing Eq. (1.6) in terms of vectors representing hyperspectral pixels leads to

$$\mathbf{r} = \sum_{k=1}^K a_k \mathbf{s}_k, \quad (1.7)$$

where the  $M \times 1$  vector  $\mathbf{r}$  approximates  $f(\lambda)$  and the  $M \times 1$  vector  $\mathbf{s}_k$  approximates the spectral density of class  $k$ ,  $f_k(\lambda)$ . Defining the vector  $\mathbf{a} = [a_1, a_2, \dots, a_K]^T$  and the matrix  $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_K]$ , Eq. (1.7) is equivalently described as the matrix-vector product

$$\mathbf{r} = \mathbf{S}\mathbf{a}. \quad (1.8)$$

This formulation assumes that the observed spectral density is an additive linear combination of  $K$  class spectral densities. Thus, authors often refer to Eqs. (1.6-1.8) as the linear mixture model.

Numerous papers have explored methods to determine the contributions of the constituent spectral classes from an observed hyperspectral pixel with the linear mixture model [1, 16, 18, 19, 20, 21, 22, 23, 24, 25, 26]. A brief overview of a few of these methods follows.

An orthogonal subspace projection (OSP) approach to classify hyperspectral images is proposed in [21]. This method assumes the presence of  $K$  spectral signatures in an observed hyperspectral pixel with  $K_t$  target classes of interest. For an observed hyperspectral pixel, the OSP approach removes the interference caused by the  $K - K_t$  classes and determines the contribution of the target classes. This method implicitly assumes that the  $K$  spectral signatures are explicitly known and any perturbations are attributed to independent measurement noise. The subsequent paper [18] reviews this method and several other methods in an attempt to establish comparisons based on performance.

The Optical Real-time Adaptive Spectral Identification System (ORASIS) [26] autonomously establishes a set of  $K$  spectral signatures, or endmembers, from the image under investigation and determines the contribution of each endmember in the observed hyperspectral pixels. The actual ORASIS algorithm is more complex, and the reader is

encouraged to refer to [26] for a detailed description of the ORASIS algorithm. Simply put, the algorithm designates the endmembers as the  $K$  points defining the convex hull of the observed hyperspectral pixels. Thus, the mixtures lie inside the convex hull and are represented by an additive linear mixture of the endmembers.

A total least squares approximation of the spectral signature means from a set of training examples of mixed pixels was proposed in [22]. In addition, the authors of [22] find estimates of the class contributions using the quadratic programming (QP) method. The QP approach finds a solution  $\hat{\mathbf{a}}$  minimizing  $\|\mathbf{r} - \mathbf{S}\mathbf{a}\|$  and satisfying all of the constraints of the linear mixture model. The authors compare the QP method to a Lagrange multiplier formulation of the problem proposed and find marginal improvements with the QP approach.

The current interest in independent component analysis (ICA) [27] to find an appropriate representation of a collection of data as a linear transform of the original data has led to its application to the spectral mixture problem. Succinctly stated, ICA is an unsupervised method that finds a linear representation of observed data with statistically independent or uncorrelated components. The authors of [24] investigate the use of ICA for the mixture problem with hyperspectral data. The constraint that the contribution vector must sum-to-one immediately violates the ICA assumption of mutually independent sources. The authors find that estimates found via ICA improve as the signal-to-noise ratio increases or as the spectral signatures of the classes become more distinct. This conclusion follows from the notion of ICA that the components are independent.

Due to natural material variations and changes in the illuminant intensity in a scene, the observed spectral density of that material will vary. The methods described above assume that the spectral densities of material remain constant in a given scene. Considering the natural material variations, it is more accurate to describe the spectral class of a material according to its first and second order statistics. Approximations of the means of the spectral class may be used if the actual spectral signature in an observed pixel is not accurately known. Two methods are known to consider the presence of variation in the classes [28, 29]. An overview of each method follows.

The authors of [28] address the fuzzy nature of the spectral classes with a stochastic mixture model (SMM) interpretation. The model defines the spectral classes as multivariate Gaussians and use the Stochastic-Expectation-Maximization (SEM) algorithm to estimate the contribution and parameters of each spectral class. The SEM algorithm is a modification

of the well-known EM algorithm. The technique described in [28] assumes the presence of pure spectral classes and mixed spectral classes in the collection of observations. The mixed spectral classes are defined by discretizing the class contributions  $a_k$ . Thus, if the class contributions have a resolution of 0.1, the technique assumes  $K_m = 9$  mixed classes. Assuming  $K_p = 2$  pure classes, the SEM algorithm uses the observed pixels to find the parameters and contributions of the multivariate Gaussian mixture with  $K = K_m + K_p = 11$  components. Then, each pixel is assigned to one of the  $K$  classes based on posterior probabilities determined by the SEM algorithm. This leads to an approximation of the contribution of the classes in the pixels. The authors of [28] lament the problem of dimensionality with their formulation of the problem, for as the number of classes increases the algorithm becomes impractical, especially for high-dimensional spectral data.

The paper [29] considers the variation in knowledge of the class spectral signatures and implements a restricted total least squares (RTLS) algorithm. The RTLS formulation solves the common TLS problem with constraints on the solution. The authors find improved performance with the RTLS algorithm when compared to least squares methods; however, the authors also note the increased complexity of the proposed algorithm.

The formulation of the problem considered in this work in some ways blends ideas from both the linear and stochastic mixture model interpretations. The fundamental structure of the problem is based on the linear mixture model.

## 1.2 Definition of the Problem

The problem of interest is to estimate the proportion of each of the  $K$  classes present in a finite mixture model. Since, the underlying probability densities of the mixture are, in general, nonparametric, one must estimate the densities from samples of each class. This section formally defines the problem of interest. As the discussion develops, the mathematical notation used will be introduced.

Consider the scalar random variable  $X_k$  from class  $k$  with a probability density  $f_k(x)$ . Let the  $M \times 1$  vector  $\mathbf{s}_k$ , previously obtained from samples, denote a discrete approximation of the probability density function  $f_k(x)$ . The sum of the elements of  $\mathbf{s}_k$  must equal one with respect to the size of the intervals, defined by the  $M \times 1$  vector  $\mathbf{b}_w$ , whose elements correspond with the elements of  $\mathbf{s}_k$ . Denote the arithmetic mean of a

collection of the vectors  $\mathbf{s}_k$  as  $\bar{\mathbf{s}}_k$ . A collection of unclassified random variables  $X$  from  $K$  classes has a probability density function  $f(x)$ . The  $M \times 1$  vector  $\mathbf{r}$  is a representation of a set  $\mathcal{X}$  of  $N$  samples with a density function  $f(x)$ . For example,  $\mathbf{r}$  may be a histogram of the  $N$  samples in  $\mathcal{X}$ . The formation of the vector  $\mathbf{r}$  from sample data must follow the formation of the  $\mathbf{s}_k$  vectors from sample data for class  $k$ . In other words, the vectors  $\mathbf{r}$  and  $\mathbf{s}_k$  must be normalized with respect to the same size intervals or

$$\mathbf{r}^T \mathbf{b}_w = \mathbf{s}_k^T \mathbf{b}_w = 1. \quad (1.9)$$

The elements of  $\mathbf{b}_w$  are essentially the bin widths of an  $M$ -bin histogram.

It is known that

$$E\{\mathbf{r}\} = \sum_{k=1}^K a_k \bar{\mathbf{s}}_k \quad (1.10)$$

where  $\bar{\mathbf{s}}_k = E\{\mathbf{s}_k\}$ . Let  $N_k$  denote the number of samples from the  $k^{\text{th}}$  class such that  $N = \sum_{k=1}^K N_k$ . Notice that the true contribution from class  $k$ ,  $\frac{N_k}{N} \mathbf{s}_k$ , may only be approximated by  $\frac{N_k}{N} \bar{\mathbf{s}}_k$ . This approximation is accounted for by inserting a perturbation term  $\Delta \mathbf{s}_k$  for the estimate of the mean of class  $k$ . The perturbation term  $\Delta \mathbf{s}_k$  is given by

$$\Delta \mathbf{s}_k = \bar{\mathbf{s}}_k - \mathbf{s}_k, \quad (1.11)$$

and the perturbation is assumed to be a zero mean random process with covariance  $\Sigma_k$ . Considering the deviations in the estimates of the mean of each class, the linear mixture  $\mathbf{r}$  is written

$$\mathbf{r} = \sum_{k=1}^K a_k (\bar{\mathbf{s}}_k + \Delta \mathbf{s}_k). \quad (1.12)$$

In matrix-vector notation, the linear mixture may be written

$$\mathbf{r} = (\bar{\mathbf{S}} + \Delta \mathbf{S}) \mathbf{a}, \quad (1.13)$$

where  $\bar{\mathbf{S}} = [\bar{\mathbf{s}}_1, \bar{\mathbf{s}}_2, \dots, \bar{\mathbf{s}}_k]$ ,  $\Delta \mathbf{S} = [\Delta \mathbf{s}_1, \Delta \mathbf{s}_2, \dots, \Delta \mathbf{s}_k]$ , and  $\mathbf{a} = [a_1, a_2, \dots, a_k]^T$ . In some cases, there may be additional perturbations attributed to measurement uncertainty. For this case, we have the model

$$\mathbf{r} = (\bar{\mathbf{S}} + \Delta \mathbf{S}) \mathbf{a} + \boldsymbol{\eta}, \quad (1.14)$$

where  $\boldsymbol{\eta}$  is zero mean signal independent noise with covariance  $\Sigma_\eta$ .

The nature of this mixture problem imposes constraints on  $\mathbf{a}$ . Since  $\mathbf{a}$  represents proportions of the  $K$  classes in the recorded mixture, the elements of  $\mathbf{a}$  must sum-to-one and lie on the interval  $[0, 1]$ . The vector  $\mathbf{a}$  is constrained to the set  $S_a$  defined in Eq. (1.15).

$$S_a = \left\{ \mathbf{a} \in \mathbb{R}^K \mid \sum_{k=1}^K a_k = 1, \quad a_k \geq 0 \right\} \quad (1.15)$$

Estimation methods that can take the perturbations associated with both  $\Delta \mathbf{S}$  and  $\boldsymbol{\eta}$  into account include total least squares (TLS) and projection onto convex sets (POCS). The constraints in Eq. (1.15) on an estimate of  $\mathbf{a}$  motivate the use of set theoretic estimation using POCS to estimate the proportions  $\mathbf{a}$ .

## Chapter 2

# Theoretical Background

The estimation of the parameters of a probability density function develops from Bayes's theorem. This section first reviews the well documented maximum likelihood and maximum a posteriori (MAP) formulations for arbitrary probability density functions. The EM algorithm was developed by considering the observations of a finite mixture model as being incomplete, and the lack of class labels for each of the observations motivates the idea of incomplete data. Thus, the subsequent discussion, adapted from [5], of the maximum likelihood problem with incomplete data is presented.

Throughout this section, the objective is to find the appropriate parameters  $\boldsymbol{\theta}$  in the  $K$  component finite parametric mixture model defined in Eq. (1.5) from a set  $\mathcal{X}$  of  $N$  samples. It is assumed that the  $N$  samples in  $\mathcal{X}$  are independent and identically distributed (i.i.d.) with conditional probability density  $f(x; \boldsymbol{\theta})$ . Furthermore, it is assumed that the  $n^{\text{th}}$  sample  $x_n \in \mathcal{X}$  is associated with only one of the  $K$  component densities.

### 2.1 Maximum Likelihood and Maximum A Posteriori

From elementary statistics, it is known that the joint probability density function of a single observation  $x$  and the vector of parameters  $\boldsymbol{\theta}$  may be written

$$f(x, \boldsymbol{\theta}) = f(x; \boldsymbol{\theta})f(\boldsymbol{\theta}) = f(\boldsymbol{\theta}; x)f(x) \quad (2.1)$$

which leads to Bayes's theorem, or

$$f(\boldsymbol{\theta}; x) = \frac{f(x; \boldsymbol{\theta})f(\boldsymbol{\theta})}{f(x)}. \quad (2.2)$$

Note that the denominator in Eq. (2.2) merely indicates the probability of observing  $x$  and scales the numerator such that the integral of the posterior probability density function  $f(\boldsymbol{\theta}; x)$  equals one. The term  $f(x; \boldsymbol{\theta})$  in the numerator is called the likelihood, and  $f(\boldsymbol{\theta})$  is the prior probability associated with the vector  $\boldsymbol{\theta}$ . For the entire collection of  $N$  samples, the posterior probability density function is given as

$$f(\boldsymbol{\theta}; \mathcal{X}) = \prod_{n=1}^N \frac{f(x_n; \boldsymbol{\theta})f(\boldsymbol{\theta})}{f(x_n)}. \quad (2.3)$$

For the finite mixture model in Eq. (1.5), the maximum posterior estimate,  $\hat{\boldsymbol{\theta}}_{map}$ , maximizes the posterior probability in Eq. (2.3) given the observation set  $\mathcal{X}$ . Notice that neglecting the denominator in Eq. (2.3) still yields the maximizing  $\hat{\boldsymbol{\theta}}_{map}$ .

The MAP estimate  $\hat{\boldsymbol{\theta}}_{map}$  may be determined using differential calculus and is readily found by solving

$$\hat{\boldsymbol{\theta}}_{map} = \arg \max_{\boldsymbol{\theta}} \prod_{n=1}^N f(x_n; \boldsymbol{\theta})f(\boldsymbol{\theta}), \quad (2.4)$$

which implies that  $\hat{\boldsymbol{\theta}}_{map}$  solves

$$\mathbf{D}_{\boldsymbol{\theta}} \left[ \prod_{n=1}^N f(x_n; \boldsymbol{\theta})f(\boldsymbol{\theta}) \right] = \mathbf{0}, \quad (2.5)$$

where  $\mathbf{D}_{\boldsymbol{\theta}}[\cdot]$  is the differential operator with respect to  $\boldsymbol{\theta}$ , and  $\mathbf{0}$  is a vector of zeros with the same dimensionality as the vector  $\boldsymbol{\theta}$  of parameters. The form of the probability density function  $f(x; \boldsymbol{\theta})$  often contains an exponential term. In such cases, maximizing the logarithm of  $f(x; \boldsymbol{\theta})f(\boldsymbol{\theta})$  is equally sufficient, since the logarithm is a monotonic function. Thus, the MAP estimate can also be determined by solving

$$\hat{\boldsymbol{\theta}}_{map} = \arg \max_{\boldsymbol{\theta}} \sum_{n=1}^N \log \{f(x_n; \boldsymbol{\theta})f(\boldsymbol{\theta})\}. \quad (2.6)$$

In the case that the prior probability density function  $f(\boldsymbol{\theta})$  is unknown, the prior probability density function may be assumed to be uniform, and any value of  $\boldsymbol{\theta}$  is equally likely. Neglecting the denominator and assuming a uniform distribution for  $\boldsymbol{\theta}$  leads to the likelihood function:

$$L(\boldsymbol{\theta}) = \prod_{n=1}^N f(x_n; \boldsymbol{\theta}). \quad (2.7)$$

Again, given the observation set  $\mathcal{X}$ , the estimate  $\hat{\boldsymbol{\theta}}_{ml}$  maximizing the likelihood function is said to be the maximum likelihood (ML) estimate and is found by solving

$$\hat{\boldsymbol{\theta}}_{ml} = \arg \max_{\boldsymbol{\theta}} \log \{L(\boldsymbol{\theta})\} = \arg \max_{\boldsymbol{\theta}} \log \left\{ \prod_{n=1}^N f(x_n; \boldsymbol{\theta}) \right\} = \arg \max_{\boldsymbol{\theta}} \sum_{n=1}^N \log f(x_n; \boldsymbol{\theta}). \quad (2.8)$$

For the same reasons mentioned for the MAP estimate, the logarithm of the likelihood function is maximized to find  $\hat{\boldsymbol{\theta}}_{ml}$ . The function  $\log L(\boldsymbol{\theta})$  is commonly called the log-likelihood function.

## 2.2 Maximum Likelihood Estimation and Incomplete Data

Suppose that an unobservable  $K \times 1$  random vector  $\mathbf{Z}_n$  is associated with observation  $x_n$ . Let  $\mathbf{z}_n$  be a realization of  $\mathbf{Z}_n$ . The vector  $\mathbf{z}_n$  assumes a value of one in its  $k^{\text{th}}$  element if  $x_n$  belongs to class  $k$ , and all remaining elements of  $\mathbf{z}_n$  are zero. Recall that the probability that  $x_n$  comes from the  $k^{\text{th}}$  distribution is  $a_k$ . Thus, the random vector  $\mathbf{Z}_n$  is said to come from a multinomial distribution with one trial and  $K$  outcomes with respective probabilities  $a_k$ . In other words, the probability mass function is defined as

$$P(\mathbf{Z}_n = \mathbf{z}_n) = \prod_{k=1}^K a_k^{[\mathbf{z}_n]_k}, \quad (2.9)$$

where  $[\mathbf{z}_n]_k$  denotes the  $k^{\text{th}}$  element of the vector  $\mathbf{z}_n$ . This perspective indicates, in general, that  $f(x; \boldsymbol{\theta})$  is the unconditional density of  $X$  and  $f_k(x; \boldsymbol{\xi}_k)$  is the conditional density of  $X$  given  $[\mathbf{Z}]_k = 1$ . Considering  $a_k$  as the prior probability that an observation  $x$  belongs to the  $k^{\text{th}}$  class in the mixture, the posterior probability that  $x$  belongs to the  $k^{\text{th}}$  class given  $x$  and  $\boldsymbol{\theta}$  is given by

$$\tau_k(x; \boldsymbol{\theta}) = E\{[\mathbf{Z}]_k = 1; x, \boldsymbol{\theta}\} = P([\mathbf{Z}]_k = 1; x, \boldsymbol{\theta}) = \frac{a_k f_k(x; \boldsymbol{\xi}_k)}{\sum_{h=1}^K a_h f_h(x; \boldsymbol{\xi}_h)}. \quad (2.10)$$

The literature refers to the data in  $\mathcal{X}$  as the incomplete data. Let the complete data set  $\mathcal{X}_c$  include the data in  $\mathcal{X}$  and the categorical label vectors  $\mathbf{z}_n$ . The logarithm of the likelihood function assuming the complete data set  $\mathcal{X}_c$  is given by

$$\log \{L_c(\boldsymbol{\theta})\} = \sum_{n=1}^N \sum_{k=1}^K [\mathbf{z}_n]_k \log \{a_k f_k(x_n; \boldsymbol{\xi}_k)\}, \quad (2.11)$$

where  $L_c(\boldsymbol{\theta})$  denotes the complete likelihood function. The relationship between Eq. (2.11) and Eq. (2.8) is straightforward if it is assumed that a given sample  $x_n$  belongs to only one component. Thus, the MLE assuming incomplete data is found by solving

$$\hat{\boldsymbol{\theta}}_{ml} = \arg \max_{\boldsymbol{\theta}} \log \{L_c(\boldsymbol{\theta})\}. \quad (2.12)$$

## 2.3 Previous Methods for Finite Mixture Models with Parametric Components

The discussion in the previous section introduced the notion of incomplete data from a finite mixture model. To make the data complete, labels must be attributed to each observation. The EM algorithm utilizes this formulation to determine estimates of the parameters for a finite mixture model with parametric component probability density functions. This section reviews the general EM algorithm. Subsequent algorithms have emerged based on the EM algorithm. Many of these algorithms consider the problem when the number of components is unknown [7]. A brief overview of these methods in Section 2.3.2 concludes the discussion of finite mixture models with parametric components.

### 2.3.1 EM Algorithm

The EM algorithm was introduced by Dempster, et al. in 1977 as an iterative algorithm to solve the ML problem when the observations can be viewed as incomplete data [6]. Subsequent texts and articles review this algorithm and provide further insight to the algorithm [4, 5]. The formulation of the EM algorithm to solve the MAP problem is addressed in [5]. In this section, the EM algorithm for the ML problem following the discussion in [5] is examined. This formulation relates more closely to our nonparametric finite mixture model problem, since no statistics for the class proportions are available. However, the modifications to solve the MAP problem are straightforward.

#### General EM Algorithm

The EM algorithm finds the estimate of Eq. (2.12) by repeating two steps until some criterion for convergence is met. The algorithm assumes an initial estimate of the

parameters  $\hat{\boldsymbol{\theta}}^{(0)}$ . Let  $i$  denote the index of iteration.

The first step is the expectation (E) step and concerns the unobserved data  $\mathbf{z}_n$ . This step computes the expectation of the complete data log-likelihood function given the set of observations  $\mathcal{X}$  and the current estimate  $\hat{\boldsymbol{\theta}}^{(i)}$ . Observe that the complete log-likelihood function  $\log L_c(\boldsymbol{\theta})$  in Eq. 2.11 is linear with respect to  $[\mathbf{z}_n]_k$ . Given the conditional expectation of  $[\mathbf{Z}_n]_k$  in Eq. (2.10), the E-step at iteration  $i + 1$  is formally defined as

$$Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}^{(i)}) = \sum_{n=1}^N \sum_{k=1}^K \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) \log \{a_k f_k(x_n; \boldsymbol{\xi}_k)\}, \quad (2.13)$$

where

$$\tau_k(x_n; \hat{\boldsymbol{\theta}}) = \frac{\hat{a}_k f_k(x_n; \hat{\boldsymbol{\xi}}_k)}{\sum_{h=1}^K \hat{a}_h f_h(x_n; \hat{\boldsymbol{\xi}}_h)}, \quad (2.14)$$

$\hat{a}_k$  denotes the estimate of  $a_k$ , and  $\hat{\boldsymbol{\xi}}_k$  denotes the estimate of  $\boldsymbol{\xi}_k$ . The second step is the maximization (M) step, which finds the update  $\hat{\boldsymbol{\theta}}^{(i+1)}$  that globally maximizes  $Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}^{(i)})$ .

Thus, the update solves

$$\hat{\boldsymbol{\theta}}^{(i+1)} = \arg \max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}^{(i)}). \quad (2.15)$$

The prior probabilities  $a_k$  are independent of the component density parameters  $\{\boldsymbol{\xi}_k\}_{k=1}^K$  and are determined by maximizing  $Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}^{(i)})$  subject to the constraint that  $\sum_{k=1}^K a_k = 1$ . This is accomplished with Lagrange multipliers [30]. Consider the function  $\phi_a(\boldsymbol{\theta})$  defined as

$$\phi_a(\boldsymbol{\theta}) = Q(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}^{(i)}) + \lambda_a \left( \sum_{k=1}^K a_k - 1 \right), \quad (2.16)$$

where  $\lambda_a$  is the Lagrange multiplier. The maximizing prior probabilities,  $a_k$ , are found with differential calculus. Thus,  $a_k$  and  $\lambda_a$  must satisfy the following equations

$$\frac{\partial \phi_a(\boldsymbol{\theta})}{\partial a_k} = \lambda_a + \frac{1}{a_k} \sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) = 0, \quad (2.17)$$

and

$$\frac{\partial \phi_a(\boldsymbol{\theta})}{\partial \lambda_a} = \sum_{k=1}^K a_k - 1 = 0. \quad (2.18)$$

Solving Eq. (2.17) for  $\lambda_a$  and substituting that result into Eq. (2.18) yields

$$\sum_{k=1}^K \left( -\frac{1}{\lambda_a} \sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) \right) = 1 \implies \sum_{n=1}^N \sum_{k=1}^K \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) = -\lambda_a. \quad (2.19)$$

Observe that from the definition of  $\tau_k(x; \hat{\boldsymbol{\theta}}^{(i)})$

$$\sum_{k=1}^K \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) = \sum_{k=1}^K \frac{\hat{a}_k^{(i)} f_k(x_n; \hat{\boldsymbol{\xi}}_k^{(i)})}{\sum_{h=1}^K \hat{a}_h^{(i)} f_h(x_n; \hat{\boldsymbol{\xi}}_h^{(i)})} = 1. \quad (2.20)$$

Thus, from Eq. (2.19),  $-\lambda_a = N$ . Solving Eq. (2.17) for  $a_k$  and substituting  $N = -\lambda_a$  produces the update  $a_k^{(i+1)}$

$$a_k^{(i+1)} = \frac{1}{N} \sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}). \quad (2.21)$$

The remaining parameters  $\{\boldsymbol{\xi}_k\}_{k=1}^K$  must be determined with respect to the chosen underlying component densities. It is readily noted that the update  $\{\boldsymbol{\xi}_k^{(i+1)}\}_{k=1}^K$  is the root of

$$\sum_{n=1}^N \sum_{k=1}^K \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) \mathbf{D}_{\boldsymbol{\xi}_k} [\log f_k(x_n; \boldsymbol{\xi}_k)] = \mathbf{0}, \quad (2.22)$$

where  $\mathbf{D}_{\boldsymbol{\xi}_k}[\cdot]$  denotes the differential operator with respect to  $\boldsymbol{\xi}_k$ .

The EM algorithm continues until the difference  $L(\boldsymbol{\theta}^{(i+1)}) - L(\boldsymbol{\theta}^{(i)})$  reaches a threshold value. Recall that  $L(\boldsymbol{\theta}^{(i)})$  is to be maximized, and [6] illustrates that the likelihood is nondecreasing with each iteration or  $L(\boldsymbol{\theta}^{(i+1)}) \geq L(\boldsymbol{\theta}^{(i)})$ .

## EM Algorithm for Finite Mixture Models with Gaussian Components

The application of the EM algorithm to finite mixture models with Gaussian components is straightforward. Let the function  $f_{\mathcal{N}}(x; \mu_k, \sigma_k)$  denote a Gaussian (normal) probability density with mean  $\mu_k$  and standard deviation  $\sigma_k$ . The probability density function,  $f_k(x; \boldsymbol{\xi}_k)$ , of the  $k^{\text{th}}$  class is given by

$$f_k(x; \boldsymbol{\xi}_k) = f_{\mathcal{N}}(x; \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}} \quad \forall x \in \mathbb{R}. \quad (2.23)$$

Thus, the parameter  $\boldsymbol{\xi}_k$  defines the mean  $\mu_k$  and standard deviation  $\sigma_k$  of the  $k^{\text{th}}$  component density. As mentioned in the general formulation of the EM algorithm, the updates of the prior probabilities  $a_k$  are independent of the form of the component densities. Thus, the update for  $a_k$  is given by Eq. (2.21) for  $f_k(x; \boldsymbol{\xi}_k) = f_{\mathcal{N}}(x; \mu_k, \sigma_k)$ . The updates of the parameters  $\boldsymbol{\xi}_k$  at iteration  $i$  are found by find the roots of Eq. (2.22). Note that the logarithm of  $f_{\mathcal{N}}(x; \mu_k, \sigma_k)$  is given by

$$\log f_{\mathcal{N}}(x; \mu_k, \sigma_k) = -\log(\sqrt{2\pi}\sigma_k) - \frac{(x - \mu_k)^2}{2\sigma_k^2}. \quad (2.24)$$

First, consider finding the updates for the mean  $\mu_k$  of each class. The derivative of  $\log f_{\mathcal{N}}(x_n; \mu_k, \sigma_k)$  with respect to  $\mu_k$  is given by

$$\frac{\partial \log f_{\mathcal{N}}(x_n; \mu_k, \sigma_k)}{\partial \mu_k} = -\frac{x - \mu_k}{\sigma_k^2}, \quad (2.25)$$

and substituting this result into Eq. (2.22) leads to

$$\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) \frac{(x - \mu_k)}{\sigma_k^2} = 0. \quad (2.26)$$

Simplifying Eq. (2.26) and solving for  $\hat{\mu}_k^{(i+1)}$ , we find the update to be

$$\hat{\mu}_k^{(i+1)} = \frac{\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) x_n}{\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)})}. \quad (2.27)$$

Now, find the updates of the standard deviation  $\sigma_k$ . The derivative of  $f_{\mathcal{N}}(x_n; \mu_k, \sigma_k)$  with respect to  $\sigma_k$  is given by

$$\frac{\partial \log f_{\mathcal{N}}(x_n; \mu_k, \sigma_k)}{\partial \sigma_k} = -\frac{1}{\sigma_k} + \frac{(x - \mu_k)^2}{\sigma_k^3}, \quad (2.28)$$

and substituting this result into Eq. (2.22) leads to

$$\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) \left( \frac{(x - \mu_k)^2}{\sigma_k^3} - \frac{1}{\sigma_k} \right) = 0. \quad (2.29)$$

Simplifying Eq. (2.29) and solving for  $\sigma_k^2$ , we find the update to be

$$\hat{\sigma}_k^{(i+1)} = \sqrt{\frac{\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)}) (x_n - \hat{\mu}_k^{(i+1)})^2}{\sum_{n=1}^N \tau_k(x_n; \hat{\boldsymbol{\theta}}^{(i)})}}. \quad (2.30)$$

Notice that the update of the standard deviation depends on the updated mean  $\hat{\mu}_k^{(i+1)}$ .

## Remarks

The EM algorithm may converge to a large relative local maxima, and this typically indicates that the algorithm attempted to fit a component with a very small variance to the data [5]. Running the EM algorithm for several different initial estimates may remedy this problem. The estimate yielding the greatest maximum likelihood among these candidate estimates of the model could expose insufficient estimates associated with local maxima.

Suspicious solutions sometimes lie near the boundary of the parameter space. For instance, one of the proportion estimates could approach zero and encourage the variance of the density to become arbitrarily small. The culprit in this scenario may be an overestimate of the number of components, which could generate clusters containing very few samples. Methods to detect such behaviors are addressed in [5]. For this work, estimates found via the EM algorithm simply provide a benchmark for the proposed set theoretic approach. Implementing the numerous subsequent variations and modifications of the EM algorithm is superfluous. The basic EM algorithm suffices for the purposes of this work.

### 2.3.2 Unsupervised Learning of Finite Mixture Models

Algorithms have been proposed to mitigate the problem of selecting the number of components in a finite mixture model when using the EM algorithm. While numerous approaches have been proposed, this short overview highlights both the fundamental approach to selecting the number of components and the specific method proposed by [7]. For a more thorough discussion of these methods, refer to [3], [5], and [7] and the references therein.

The approach introduced by [7] was inspired by previous methods that essentially choose the “best” estimate among a set of EM estimates for several values of  $K$ , the number of components. It was noted in [5] that the likelihood increases as the number of components increases. Thus, the ML estimate, when  $K$  is not fixed, naturally tends towards finite mixture models with more components. Mindful of this tendency, the methods introduce a function to penalize mixture models with more components. Define  $\mathcal{M}_K$  as the class of all possible  $K$ -component mixture models for chosen component densities. The methods find EM estimates for a range of values for  $K$ , i.e. for  $K_{min}$  to  $K_{max}$ . A cost function penalizing mixture models with more components selects the “best” estimate. The authors of [7] indicate that these method can become computationally intensive due to running the EM algorithm several times.

The method proposed by [7] distinguishes itself from previous methods by searching for the “best” finite mixture model among the collection of candidate models given by the set

$$\bigcup_{K=K_{min}}^{K_{max}} \mathcal{M}_K. \quad (2.31)$$

The distinction is subtle but important. In contrast to the previous methods, which choose the best estimate of the parameters from a set of  $K_{max} - K_{min} + 1$  candidates, the new method searches throughout all values of  $K$  under consideration for the “best” estimate of the parameters. Furthermore, the proposed method reduces the sensitivity of the initial estimate and avoids estimates near the boundary of the parameter space.

To produce an estimate, a variant of the EM algorithm is constructed, which is derived from the minimum message length criterion (MML) prevalent in information theory. The MML criterion assumes that the ability to produce short messages for data correlates well with a good data generation model. Consider a source that transmits data modeled by the random variable  $X$  distributed according to the probability density function  $f(x; \theta)$  with parameters  $\theta$ . When knowledge of the parameters  $\theta$  are unknown by the receiver to some extent, the transmitted message must also include this information. A two part message containing the data  $X$  and the parameters  $\theta$  has a length defined as

$$\text{length}(\theta; X) = \text{length}(\theta) + \text{length}(X; \theta). \quad (2.32)$$

The goal is to minimize the length  $\text{length}(\theta; X)$  of the transmitted message.

To utilize the MML criterion, the parameters  $\theta$  must be quantized to some arbitrary precision in order to minimize a meaningful measure of message length. The specific cost function developed by [7] is beyond the scope of this work. However, the justification for this approach is that reducing the length of a message corresponds to pruning unnecessary component densities. A cost function to be minimized, adhering to this notion, is presented in [7]. Several experiments in [7] illustrate the good performance of the proposed MML method to estimate the finite mixture model from sample data.

## 2.4 Approximations of Nonparametric Probability Densities

When the components of the finite mixture model are nonparametric, estimates of the densities must be utilized to determine the presence of each component in a mixture. This section briefly reviews the ideas set forth by Parzen [9] and Rosenblatt [31] regarding the estimation of an arbitrary density function  $f(x)$ . General approximations of probability densities are discussed first, where we follow the presentation in [10]. The specific approximation given by a histogram is considered.

### 2.4.1 General Approximations

Suppose that the random variable  $X$  has a probability density function  $f(x)$ . It is well known that the probability that an observation  $x$  will lie on the interval  $[\alpha, \beta]$  is given by

$$P = \int_{\alpha}^{\beta} f(x)dx. \quad (2.33)$$

For a very small interval, i.e.  $\alpha \approx \beta$  with  $\beta > \alpha$ , the probability  $P$  is an approximation to the density function  $f(x)$ .

Suppose that  $N$  observations  $x_1, x_2, \dots, x_N$  of the random variable  $X$  are available. Then, the probability  $P_k$  that  $k$  samples will lie on the interval  $[\alpha, \beta]$  is given by the binomial density

$$P_k = \binom{N}{k} P^k (1 - P)^{N-k}. \quad (2.34)$$

The expected value of the binomial distribution is  $E\{k\} = NP$ . This indicates that the true probability  $P$  is given by

$$P = \frac{E\{k\}}{N}. \quad (2.35)$$

Let  $n_{[\alpha, \beta]}$  denote the number of samples in the interval  $[\alpha, \beta]$ . Then, a reasonable approximation of  $P$  on the interval  $[\alpha, \beta]$  is  $\frac{n_{[\alpha, \beta]}}{N}$ . Define  $h = |\alpha - \beta|$  as the width of the interval  $[\alpha, \beta]$ , then

$$P = \int_{\alpha}^{\beta} f(x')dx' \approx f(x)h, \quad (2.36)$$

and on the interval  $[\alpha, \beta]$

$$f(x) \approx \frac{\frac{n_{[\alpha, \beta]}}{N}}{h} = \frac{n_{[\alpha, \beta]}}{Nh}. \quad (2.37)$$

Parzen acknowledges that the interval width  $h$  should vary as the number of samples  $N$  increases. Namely,  $h$  should approach 0 as  $N$  approaches  $\infty$ . For small  $h$  and small  $N$ , the number of samples within a region may be zero. A sparse, perhaps spiky, and erroneous estimate of the true density is likely to result. In short, considering the width of interval as a function of  $N$ , or  $h(N)$ , and the number of samples that fall in the interval  $[\alpha, \beta]$  given  $N$  observations as  $k(N)$ , then the following three conditions must be satisfied

$$\lim_{N \rightarrow \infty} h(N) = 0 \quad (2.38)$$

$$\lim_{N \rightarrow \infty} k(N) = \infty \quad (2.39)$$

$$\lim_{N \rightarrow \infty} \frac{k(N)}{N} = 0 \quad (2.40)$$

The first condition guarantees that  $h$  will become arbitrarily small only if a very large number of samples is available. This conclusion is intuitive, for as the number of samples increases, more information about the underlying density function is available. The second condition indicates that as the number of samples  $N$  approach infinity, an infinite number of samples will lie in the region associated with the probability  $P$ . The final condition suggests that for very large  $N$ , the contribution of more samples will contribute very little to the estimate of  $P$ .

Rosenblatt and Parzen introduce an estimate of  $f(x)$  given  $N$  samples with the function

$$f_N(x) = \frac{1}{N} \sum_{j=1}^N \frac{1}{h(N)} k\left(\frac{x - x_j}{h(N)}\right), \quad (2.41)$$

where  $k(x)$  is a window (or weighting) function chosen to satisfy<sup>1</sup> the following

$$k(x) \geq 0 \quad (2.42)$$

$$\int_{-\infty}^{\infty} k(x) dx = 1. \quad (2.43)$$

While several functions satisfy the conditions of the window, define the rectangular window function as

$$k(x) = \begin{cases} 1 & |x| \leq \frac{1}{2} \\ 0 & \text{else} \end{cases}. \quad (2.44)$$

Consider a vector  $\mathbf{s}$  of length  $M$  as a sampled version of the function  $f_N(x)$  such that the  $m^{\text{th}}$  element of  $\mathbf{s}$  is given by

$$[\mathbf{s}]_m = f_N([\mathbf{b}]_m) = \frac{1}{N} \sum_{j=1}^N \frac{1}{h(N)} k\left(\frac{[\mathbf{b}]_m - x_j}{h(N)}\right), \quad m = 0, 1, \dots, M-1 \quad (2.45)$$

where  $\mathbf{b}$  is a vector of length  $M$  of nondecreasing real numbers corresponding to the bin centers. Thus, the vector  $\mathbf{s}$  forms a normalized  $M$ -bin histogram of the probability density function  $f(x)$  when  $k(x)$  is given by Eq. (2.44).

Having described the general formulation of histograms, the discussion continues with the formulation of histograms assuming  $K$  component densities in the finite mixture density  $f(x)$  in Eq. (1.2).

---

<sup>1</sup>The actual conditions given by Parzen are more rigorous. Here the essence captured by [10] is given.

### 2.4.2 Histograms

The objective in this section is to address the estimation of any one-dimensional probability density with a histogram from sample data. Define the sample data set  $\Omega_k = \left\{ X_k^{(n)} \right\}_{n=1}^{S_{\Omega_k}}$  as a collection of  $S_{\Omega_k}$  random samples  $X_k^{(n)}$  with a probability density defined by  $f_k(x)$ . In the end, an  $M$  component vector  $\bar{\mathbf{s}}_k$  will be defined to approximate the probability density function  $f_k(x)$ .

Suppose that the values from class  $k$  lie on the continuous interval of real numbers  $\mathcal{R}_k \in (\alpha_k, \beta_k)$ ,  $\mathcal{R}_k \in \mathbb{R}$ . For  $K$  classes, the overall interval is defined as the union of the intervals for each class, or

$$\mathcal{R} = \bigcup_{k=1}^K \mathcal{R}_k \quad (2.46)$$

and  $(\alpha, \beta)$  defines the interval of the set  $\mathcal{R}$ . If the range of the random variable for a particular class is unbounded, then truncate the set  $\mathcal{R}$  and define

$$\hat{\mathcal{R}} \in [\hat{\alpha}, \hat{\beta}] \quad \text{for } |\hat{\alpha}| < \infty, |\hat{\beta}| < \infty, \hat{\alpha} < \hat{\beta}. \quad (2.47)$$

Any value outside of the set  $\hat{\mathcal{R}}$  will be projected to the nearest bound. Henceforth, assume that the interval is defined by  $\hat{\mathcal{R}}$ .

To form an  $M$ -bin histogram, the set  $\hat{\mathcal{R}}$  must be partitioned into  $M$  intervals. The intervals typically are not uniform. Let the vector  $\mathbf{b}_e \in \mathbb{R}^{(M+1) \times 1}$  define the bounds, or edges, for each bin. The elements of  $\mathbf{b}_e$  must be ordered such that  $[\mathbf{b}_e]_m < [\mathbf{b}_e]_{m+1}$  for  $m = 0, 1, \dots, M$ . The elements of the vector  $\mathbf{b}_c \in \mathbb{R}^{M \times 1}$  define the bin centers, or

$$[\mathbf{b}_c]_m = \frac{1}{2} ([\mathbf{b}_e]_{m+1} + [\mathbf{b}_e]_m), \quad m = 0, 1, \dots, M - 1. \quad (2.48)$$

Let the vector  $\mathbf{b}_w$ , denoting the bin widths, be defined as

$$[\mathbf{b}_w]_m = |[\mathbf{b}_e]_{m+1} - [\mathbf{b}_e]_m|, \quad m = 0, 1, \dots, M - 1. \quad (2.49)$$

We wish to construct  $H_k$  histograms for each class where the data for the  $h^{th}$  histogram corresponds to a subset of samples in  $\Omega_k$ . Denote the  $h^{th}$  histogram with bin centers  $\mathbf{b}_c$  formed from samples in  $\Omega_k$  by the  $M \times 1$  vector  $\mathbf{s}_k^{(h)}$ . This histogram is normalized such that  $\mathbf{b}_w^T \mathbf{s}_k^{(h)} = 1$ . Let  $\mathcal{H}_k = \left\{ \mathbf{s}_k^{(h)} \right\}_{h=1}^{H_k}$  be a set of these histograms, where  $H_k$  is the number of sample histograms for the  $k^{th}$  class. The estimate of the mean  $M$ -bin normalized histogram of the  $k^{th}$  class is given by the vector  $\bar{\mathbf{s}}_k$ . Mathematically,  $\bar{\mathbf{s}}_k$  is the arithmetic average of

the set  $\mathcal{H}_k$  associated with bin centers denoted by the  $M \times 1$  vector  $\mathbf{b}_c$ . The  $m^{th}$  bin of the normalized histogram is related to the actual probability density function by

$$[\bar{\mathbf{s}}_k]_m \approx \int_{[\mathbf{b}_e]_m}^{[\mathbf{b}_e]_{m+1}} f_k(x) dx$$

where equality is established as  $M$  approaches infinity,  $\mathbf{b}_w$  approaches a vector of zeros, and  $H_k$  approaches infinity. In this context, the point  $\bar{\mathbf{s}}_k$  is the sample mean of the collection of points in  $\mathcal{H}_k$  with sample covariance  $\bar{\Sigma}_k$  that respectively approach the class population mean  $\boldsymbol{\mu}_k$  and covariance  $\Sigma_k$  as  $N_k$  approaches infinity. The sample covariance  $\bar{\Sigma}_k$  is unlikely to be strictly diagonal, since the  $m^{th}$  bin of any histogram is often related to its adjacent bins. However, it may be assumed that the sample covariance matrix is diagonally dominant.

In the sense of Eq. (1.2), the histogram of the samples may be written as a linear mixture of the histograms for each class. Let  $\mathbf{s}_k$  be a vector of length  $M$  representing the normalized histogram  $\mathbf{r}$  of the  $N_k$  samples in  $\Omega_k$  according to the bin centers in  $\mathbf{b}_c$ . Then, the normalized histogram of the mixture of samples may be decomposed in terms of the normalized histogram for each class as

$$\mathbf{r} = \sum_{k=1}^K a_k \mathbf{s}_k, \tag{2.50}$$

where  $a_k = \frac{N_k}{N}$  is the proportion of the  $k^{th}$  class.

## Chapter 3

# Estimation Methods

The finite mixture model assuming nonparametric component probability density functions was shown to satisfy the system of equations given in Eq. (1.14) subject to the constraints in Eq. (1.15). The objective is to determine the proportions  $\mathbf{a}$  for an observation vector  $\mathbf{r}$  representing the mixture of  $K$  component densities estimated by the columns of the matrix  $\bar{\mathbf{S}}$ .

This section describes three approaches to determine an estimate of  $\mathbf{a}$ . The general least squares (LS) solution is considered first. The LS solution does not incorporate the constraints found in Eq. (1.15). Furthermore, it is assumed that the perturbations described by  $\Delta\mathbf{S}$  are negligible and the estimates  $\bar{\mathbf{S}}$  of the underlying component densities are precisely known. Thus, the LS solution,  $\hat{\mathbf{a}}_{ls}$ , is found by minimizing the perturbation attributed to the vector  $\boldsymbol{\eta}$ . The shortcomings of this solution are obvious, for it includes very little prior knowledge about a desirable solution. Furthermore, the assumption that the component densities have been correctly estimated is, at the least, naïve. The total least squares (TLS) solution assumes errors in both the measurement of  $\mathbf{r}$  and the estimate  $\bar{\mathbf{S}}$  of the underlying densities, and thus, an elaboration of TLS follows the LS discussion. Again, the constraints in Eq. (1.15) are neglected in the TLS problem. A method that incorporates the constraints in Eq. (1.15) falls into the category of set theoretic estimation [11]. Specifically, projections onto convex sets (POCS) is presented as a means to estimate the proportions  $\mathbf{a}$  subject to the constraints in Eq. (1.15). The set theoretic approach finds a feasible solution among many in a set satisfying the prior knowledge for a particular system. The set of feasible solutions

is formed by finding a point in the intersection of sets that model constraints pertinent to the system. POCS has been shown to be an effective constrained restoration method in both signal and image processing applications. As a result, a set modeled according to the weighted TLS method was developed in [12]. Here, a new, more effective set is presented which generalizes the set introduced in [12]. It is evident from Eq. (1.14) that such a set is appropriate for the finite mixture model problem under consideration. The details of the sets specific to the finite mixture model problem will be presented in Section 4. For this section, the general background for the three estimation methods is developed.

### 3.1 Least Squares

Given the, in general inconsistent, system of linear equations

$$\mathbf{r} \approx \bar{\mathbf{S}}\mathbf{a}, \quad (3.1)$$

the classical least squares method finds the minimum Euclidean norm residual  $\|\boldsymbol{\eta}\|_2 = \|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2$  such that  $\mathbf{r} - \boldsymbol{\eta} \in R(\bar{\mathbf{S}})$ . Thus, we have assumed exact knowledge of the matrix  $\mathbf{S}$ , i.e.  $\bar{\mathbf{S}} = \mathbf{S}$ . The least squares problem generalizes to minimizing the Euclidean norm of the weighted residual. Mathematically, we have

$$\min_{\mathbf{a}} \|\mathbf{D}(\mathbf{r} - \bar{\mathbf{S}}\mathbf{a})\|_2^2, \quad (3.2)$$

where  $\mathbf{D}$  is a nonsingular weighting matrix. Throughout the discussion of the least squares problem it will be assumed that  $\mathbf{D} = \mathbf{I}_M$ , for the modifications with  $\mathbf{D} \neq \mathbf{I}_M$  are obvious.

#### 3.1.1 Least Squares Solution via the Normal Equations

The least squares solution is found from the normal equations. The optimization problem in Eq. (3.2) minimizes

$$\phi_{ls}(\mathbf{a}) = \|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2^2 = \mathbf{r}^T \mathbf{r} - 2\mathbf{r}^T \bar{\mathbf{S}}\mathbf{a} + \mathbf{a}^T \bar{\mathbf{S}}^T \bar{\mathbf{S}}\mathbf{a} \quad (3.3)$$

The normal equations correspond to the first derivative of  $\phi_{ls}(\mathbf{a})$  with respect to  $\mathbf{a}$  set equal to zero.

$$\frac{d\phi_{ls}}{d\mathbf{a}} = -2\bar{\mathbf{S}}^T \mathbf{a} + 2\bar{\mathbf{S}}^T \bar{\mathbf{S}}\mathbf{a} = 0 \Rightarrow \bar{\mathbf{S}}^T \bar{\mathbf{S}}\mathbf{a} = \bar{\mathbf{S}}^T \mathbf{r} \quad (3.4)$$

To solve for  $\mathbf{a}$ , the matrix  $\bar{\mathbf{S}}^T \bar{\mathbf{S}}$  must be nonsingular. Assume  $\bar{\mathbf{S}}$  has full column rank, i.e.  $\text{rank}(\bar{\mathbf{S}}) = K$ , and let  $N(\mathbf{A})$  denote the nullspace of the matrix  $\mathbf{A}$ . Then,

$$\begin{aligned} N(\bar{\mathbf{S}}) = \{0\} &\iff \bar{\mathbf{S}}\mathbf{a} = 0 \text{ only has solution } \mathbf{a} = 0 \\ &\iff \text{for } \bar{\mathbf{S}}\mathbf{a} \neq 0, \mathbf{a}^T \bar{\mathbf{S}}^T \bar{\mathbf{S}}\mathbf{a} > 0, \forall \mathbf{a} \neq 0 \\ &\iff \bar{\mathbf{S}}^T \bar{\mathbf{S}} \text{ is positive definite} \\ &\iff \bar{\mathbf{S}}^T \bar{\mathbf{S}} \text{ is nonsingular since } \bar{\mathbf{S}}^T \bar{\mathbf{S}} \text{ has a Cholesky decomposition } \bar{\mathbf{S}}^T \bar{\mathbf{S}} = \mathbf{L}\mathbf{L}^T \end{aligned} \quad (3.5)$$

Recall, the Cholesky decomposition is unique, and the matrix  $\mathbf{L}$  is lower triangular with strictly positive diagonal elements. Since  $\mathbf{L}$  and  $\mathbf{L}^T$  are nonsingular, the matrix  $\bar{\mathbf{S}}^T \bar{\mathbf{S}}$  is guaranteed to be nonsingular. The solution to the normal equations is

$$\hat{\mathbf{a}}_{ls} = \bar{\mathbf{S}}^\dagger \mathbf{r}, \quad (3.6)$$

where  $\bar{\mathbf{S}}^\dagger = (\bar{\mathbf{S}}^T \bar{\mathbf{S}})^{-1} \bar{\mathbf{S}}^T$  is the left Moore-Penrose inverse of  $\bar{\mathbf{S}}$ . The solution  $\hat{\mathbf{a}}_{ls}$  is unique, since the Moore-Penrose inverse is unique. Furthermore, the solution  $\hat{\mathbf{a}}_{ls}$  also is the minimum Euclidean norm solution to Eq. (3.3).

Showing that the LS solution is the minimum Euclidean norm solution to Eq. (3.3) is straightforward. Consider the following proof. Suppose  $\mathbf{y} \neq \hat{\mathbf{a}}_{ls}$  minimizes  $\phi_{ls}(\mathbf{a})$ . The observed vector  $\mathbf{r}$  has the unique decomposition

$$\mathbf{r} = \mathbf{r}_{\bar{\mathbf{S}}} + \boldsymbol{\eta}, \text{ where } \mathbf{r}_{\bar{\mathbf{S}}} \in R(\bar{\mathbf{S}}), \boldsymbol{\eta} \in R(\bar{\mathbf{S}})^\perp = N(\bar{\mathbf{S}}^T), \quad (3.7)$$

and any vector  $\mathbf{x}$  solving  $\mathbf{r}_{\bar{\mathbf{S}}} = \bar{\mathbf{S}}\mathbf{x}$  is a solution. Regardless of the solution, the residual remains the same since  $\mathbb{R}^M = R(\bar{\mathbf{S}}) \oplus R(\bar{\mathbf{S}})^\perp$ , where  $\oplus$  denotes the direct sum of two vector spaces. Thus, we have

$$\|\boldsymbol{\eta}\|_2^2 = \|\bar{\mathbf{S}}\mathbf{y} - \mathbf{r}\|_2^2 = \|\bar{\mathbf{S}}\mathbf{y} - \bar{\mathbf{S}}\hat{\mathbf{a}}_{ls} - \boldsymbol{\eta}\|_2^2 = \|\bar{\mathbf{S}}(\mathbf{y} - \hat{\mathbf{a}}_{ls})\|_2^2 + \|\boldsymbol{\eta}\|_2^2, \quad (3.8)$$

where it is noted that the residual is in the nullspace of  $\bar{\mathbf{S}}$  and certainly is orthogonal to  $\bar{\mathbf{S}}(\mathbf{y} - \hat{\mathbf{a}}_{ls})$ . Furthermore, we observe

$$\|\bar{\mathbf{S}}(\mathbf{y} - \hat{\mathbf{a}}_{ls})\|_2 = 0 \iff \bar{\mathbf{S}}(\mathbf{y} - \hat{\mathbf{a}}_{ls}) = 0. \quad (3.9)$$

This implies that  $\mathbf{y} - \hat{\mathbf{a}}_{ls} \in N(\bar{\mathbf{S}})$ . However, since by design  $\hat{\mathbf{a}}_{ls} \in R(\bar{\mathbf{S}}^T)$ , it is clear that  $(\mathbf{y} - \hat{\mathbf{a}}_{ls})^T \hat{\mathbf{a}}_{ls} = 0$ . Thus,

$$\|\mathbf{y}\|_2^2 = \|(\mathbf{y} - \hat{\mathbf{a}}_{ls}) + \hat{\mathbf{a}}_{ls}\|_2^2 = \|\mathbf{y} - \hat{\mathbf{a}}_{ls}\|_2^2 + \|\hat{\mathbf{a}}_{ls}\|_2^2, \quad (3.10)$$

The assumption  $\mathbf{y} \neq \hat{\mathbf{a}}_{ls}$  implies that  $\|\mathbf{y} - \hat{\mathbf{a}}_{ls}\|_2 > 0$ . Therefore,  $\|\mathbf{y}\|_2 > \|\hat{\mathbf{a}}_{ls}\|_2$ . This shows that  $\hat{\mathbf{a}}_{ls}$  minimizes  $\phi_{ls}(\mathbf{a})$  and has the smallest Euclidean norm.

### 3.1.2 Least Squares Solution via the SVD

The least squares problem in Eq. (3.2) with  $\mathbf{D} = \mathbf{I}_M$  may be solved using the singular value decomposition (SVD) of  $\bar{\mathbf{S}}$ . Assuming  $\bar{\mathbf{S}}$  has rank  $K$ , let the SVD of  $\bar{\mathbf{S}}$  be defined

$$\begin{aligned} \bar{\mathbf{U}}^T \bar{\mathbf{S}} \bar{\mathbf{V}} &= \bar{\mathbf{\Lambda}} = \begin{bmatrix} \bar{\mathbf{\Lambda}}_K \\ \mathbf{0} \end{bmatrix}_{M \times K} \\ \bar{\mathbf{\Lambda}}_K &= \text{diag}(\bar{\sigma}_1, \bar{\sigma}_2, \dots, \bar{\sigma}_K) \\ \bar{\mathbf{U}} &= [\bar{\mathbf{u}}_1, \dots, \bar{\mathbf{u}}_M], \quad \bar{\mathbf{V}} = [\bar{\mathbf{v}}_1, \dots, \bar{\mathbf{v}}_K], \quad \bar{\mathbf{u}}_i \in \mathbb{R}^M, \bar{\mathbf{v}}_i \in \mathbb{R}^K, \\ &\bar{\sigma}_1 \geq \bar{\sigma}_2 \geq \dots \geq \bar{\sigma}_K > 0, \end{aligned} \quad (3.11)$$

where  $\bar{\mathbf{U}}^T \bar{\mathbf{U}} = \bar{\mathbf{U}} \bar{\mathbf{U}}^T = \mathbf{I}_M$  and  $\bar{\mathbf{V}}^T \bar{\mathbf{V}} = \bar{\mathbf{V}} \bar{\mathbf{V}}^T = \mathbf{I}_K$ . In terms of the the SVD of  $\bar{\mathbf{S}}$ , the norm of the residual  $\boldsymbol{\eta}$  is written [32]

$$\|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2^2 = \|\bar{\mathbf{U}}^T \mathbf{r} - \bar{\mathbf{U}}^T \bar{\mathbf{S}} \bar{\mathbf{V}} (\bar{\mathbf{V}}^T \mathbf{a})\|_2^2 = \|\bar{\mathbf{U}}^T \mathbf{r} - \bar{\mathbf{\Lambda}} \mathbf{g}\|_2^2 = \sum_{i=1}^K (\bar{\mathbf{u}}_i^T \mathbf{r} - \bar{\sigma}_i [\mathbf{g}]_i)^2 + \sum_{i=K+1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2, \quad (3.12)$$

where  $\mathbf{g} = \bar{\mathbf{V}}^T \mathbf{a}$  and  $[\mathbf{g}]_i$  is the  $i^{\text{th}}$  element of  $\mathbf{g}$ . Observe that Eq. (3.12) is minimized for  $[\mathbf{g}]_i = \frac{\bar{\mathbf{u}}_i^T \mathbf{r}}{\bar{\sigma}_i}$  for  $i = 1, \dots, K$ . The LS solution is computed as

$$\mathbf{a}_{ls} = \sum_{i=1}^K \frac{\bar{\mathbf{u}}_i^T \mathbf{r}}{\bar{\sigma}_i} \bar{\mathbf{v}}_i = \bar{\mathbf{V}} \bar{\mathbf{\Lambda}}_K^{-1} \bar{\mathbf{U}}_K^T \mathbf{r}, \quad (3.13)$$

where  $\bar{\mathbf{U}}_K$  are the first  $K$  columns of  $\bar{\mathbf{U}}$ . Furthermore, the residual given the LS solution  $\mathbf{a}_{ls}$  is

$$\|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}_{ls}\|_2^2 = \sum_{i=K+1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2. \quad (3.14)$$

### 3.1.3 Remarks

Given also the presence of uncertainty in the estimate  $\bar{\mathbf{S}}$  of  $\mathbf{S}$ , the least squares solution is not optimal. In other words, the error may not be completely attributed to the observation error  $\boldsymbol{\eta}$ . The problem is then more appropriately modeled as a total least squares (TLS) problem.

Before proceeding with a discussion of the total least squares solution, it is worthwhile to illustrate its relationship to the least squares approach. The simplest explanation

considers the equation of a line in  $\mathbb{R}$ , or

$$r = (\bar{s} + \Delta s)a + \eta. \quad (3.15)$$

Least squares assumes that there is only observation error. Thus, least squares assumes that  $\Delta s = 0$ . Clearly, this means that a solution of  $a$  is found by moving the intercept  $\eta$  as little as possible. Total least squares assumes  $\Delta s \neq 0$ , and thus, it permits perturbations in the slope of the line as well. This leads to a solution that minimizes both sources of error. Furthermore, the classical total least squares problem considers both sources of error with equal weight. Total least squares may be modified to find a solution if the sources of error have different variation.

## 3.2 Total Least Squares

The TLS method minimizes the magnitude of the perturbations,  $\Delta \mathbf{S}$  and  $\boldsymbol{\eta}$ , to find a solution  $\hat{\mathbf{a}}_{tls}$  to Eq. (1.14). This section first reviews the general TLS problem following discussions in [33, 34, 32]. Then, the TLS solution is presented using the singular value decomposition. A closed form solution to the TLS problem is described, which leads to a brief discussion of the relationship between the TLS and LS solutions. Next, a few comments are provided on scaling the perturbations,  $\Delta \mathbf{S}$  and  $\boldsymbol{\eta}$ , to have a common variance. This section concludes with remarks on the TLS solution with regard to the finite mixture model problem. The reader is encouraged to refer to [33] for a more elaborate analysis of the TLS problem.

### 3.2.1 The Total Least Squares Problem

The elements  $\mathbf{r}$  and  $\bar{\mathbf{S}}$  in Eq. (1.14) are precisely known. Thus, a solution to Eq. (1.14) is an approximate solution to

$$\mathbf{r} \approx \bar{\mathbf{S}}\mathbf{a} \Rightarrow \bar{\mathbf{S}}\mathbf{a} - \mathbf{r} \approx 0 \Rightarrow [\bar{\mathbf{S}}|\mathbf{r}] \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} \approx 0, \quad (3.16)$$

where the approximation emphasizes that, in general,  $\mathbf{r} \notin R(\bar{\mathbf{S}})$ . Assuming that  $\mathbf{r}$  is not a linear combination of the columns of  $\bar{\mathbf{S}}$ , the rank of  $[\bar{\mathbf{S}}|\mathbf{r}]$  is  $K + 1$ . This indicates that the nullspace of  $[\bar{\mathbf{S}}|\mathbf{r}]$  is the empty set. Hence, the approximation in (3.16) implies this

inconsistency. Reducing the rank of  $[\bar{\mathbf{S}}|\mathbf{r}]$  will increase the dimension of the nullspace and permit a consistent system of equations. Create the matrix  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}] \approx [\bar{\mathbf{S}}|\mathbf{r}]$  with lower rank by perturbing  $[\bar{\mathbf{S}}|\mathbf{r}]$  as little as possible. The matrix  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}]$  leads to the consistent system of equations

$$[\hat{\mathbf{S}}|\hat{\mathbf{r}}] \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} = 0, \quad (3.17)$$

and  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}] - [\bar{\mathbf{S}}|\mathbf{r}]$  approximates the underlying perturbation  $[\Delta\mathbf{S}|\boldsymbol{\eta}]$ . The classical TLS problem is mathematically written

$$\min_{\Delta\mathbf{S}, \boldsymbol{\eta}} \|[\Delta\mathbf{S}|\boldsymbol{\eta}]\|_F^2 \quad \text{subject to } \mathbf{r} + \boldsymbol{\eta} \in R(\bar{\mathbf{S}} + \Delta\mathbf{S}), \quad (3.18)$$

where  $\|\cdot\|_F$  denotes the Frobenius matrix norm. For a minimizing  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ , any  $\mathbf{a}$  satisfying Eq. (1.14) solves the TLS problem. This formulation of the problem assumes that the perturbation in both  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$  have equal variance. The mixture density problem does not make this assumption. Since the columns of  $\bar{\mathbf{S}}$  reflect the estimates of the mean of a set of histograms, an estimate of the covariance, and therefore the variance, of each column is available. Weighting or equilibrium matrices, similar to those described for the least squares formulation, may be introduced in Eq. (3.18) to standardize the perturbations to a common variance. Eq. (3.18) generalizes to

$$\min_{\Delta\mathbf{S}, \boldsymbol{\eta}} \|\mathbf{D}[\Delta\mathbf{S}|\boldsymbol{\eta}]\mathbf{T}\|_F^2 \quad \text{subject to } \mathbf{r} + \boldsymbol{\eta} \in R(\bar{\mathbf{S}} + \Delta\mathbf{S}), \quad (3.19)$$

where the matrices  $\mathbf{D}$  and  $\mathbf{T}$  are nonsingular matrices conformable with the matrix  $[\Delta\mathbf{S}|\boldsymbol{\eta}]$ . In light of this generalized formulation, Eq. (3.17) is rewritten as

$$(\mathbf{D}[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}) \mathbf{T}^{-1} \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} \approx 0. \quad (3.20)$$

Note that the inclusion of  $\mathbf{D}$  in Eq. (3.20) may seem frivolous. However, since the solution will be found via the singular value decomposition of  $\mathbf{D}[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}$ , its presence affects the outcome. The TLS solution will be presented without the weighting matrices to maintain clarity.

### 3.2.2 TLS solution via the SVD

The singular value decomposition (SVD) can be used to find the matrix  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}]$  closest to  $[\bar{\mathbf{S}}|\mathbf{r}]$ . Assuming  $[\bar{\mathbf{S}}|\mathbf{r}]$  has rank  $K + 1$ , let the SVD of  $[\bar{\mathbf{S}}|\mathbf{r}]$  be defined

$$\mathbf{U}^T[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{V} = \mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}_{K+1} \\ \mathbf{0} \end{bmatrix}_{M \times (K+1)} \quad (3.21)$$

$$\mathbf{\Lambda}_{K+1} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{K+1})$$

$$\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_M], \quad \mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_{K+1}], \quad \mathbf{u}_i \in \mathbb{R}^M, \mathbf{v}_i \in \mathbb{R}^{K+1},$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{K+1} > 0,$$

where  $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}_M$  and  $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}_{K+1}$ .

The Eckart-Young-Mirsky theorem provides a matrix approximation using SVD of Eq. (3.11). The theorem states that the distance from  $[\bar{\mathbf{S}}|\mathbf{r}]$  to the closest matrix of rank  $N$  is

$$\sigma_{K+1} = \min_{\text{rank}([\hat{\mathbf{S}}|\hat{\mathbf{r}}])=K} \left\| [\mathbf{S}|\mathbf{r}] - [\hat{\mathbf{S}}|\hat{\mathbf{r}}] \right\|_F. \quad (3.22)$$

Observe that Eq. (3.11) has the dyadic decomposition

$$[\bar{\mathbf{S}}|\mathbf{r}] = \sum_{i=1}^K \sigma_i \mathbf{u}_i \mathbf{v}_i^T + \sigma_{K+1} \mathbf{u}_{K+1} \mathbf{v}_{K+1}^T. \quad (3.23)$$

The dyadic decomposition represents a matrix of rank  $K$  into the sum of  $K$  rank one matrices [33]. Note that the matrix  $[\bar{\mathbf{S}}|\mathbf{r}]$  has rank  $K + 1$ . The perturbation associated with Eq. (3.22) is defined as

$$[\Delta \hat{\mathbf{S}}|\hat{\mathbf{r}}] = [\mathbf{S}|\mathbf{r}] - [\hat{\mathbf{S}}|\hat{\mathbf{r}}] = \sigma_{K+1} \mathbf{u}_{K+1} \mathbf{v}_{K+1}^T. \quad (3.24)$$

By definition, the Euclidean norm of the vectors  $\mathbf{u}_{K+1}$  and  $\mathbf{v}_{K+1}$  is one. Thus, the square of the Frobenius matrix norm of the right hand side of Eq. (3.24) is

$$\|\sigma_{K+1} \mathbf{u}_{K+1} \mathbf{v}_{K+1}^T\|_F^2 = \sigma_{K+1}^2 \text{tr}(\mathbf{v}_{K+1} \mathbf{u}_{K+1}^T \mathbf{u}_{K+1} \mathbf{v}_{K+1}^T) = \sigma_{K+1}^2 \text{tr}(\mathbf{v}_{K+1} \mathbf{v}_{K+1}^T) = \sigma_{K+1}^2, \quad (3.25)$$

where  $\text{tr}(\mathbf{A})$  denotes the trace of the matrix  $\mathbf{A} \in \mathbb{C}^{N \times N}$ . Clearly, this establishes the relationship defined in Eq. (3.22).

The SVD is a special case of a larger class of orthogonal matrix factorizations [35]. In general, a matrix  $\mathbf{A} \in \mathbb{C}^{M \times K}$  of rank  $r$  may be factored as

$$\mathbf{A} = \mathbf{U}\mathbf{R}\mathbf{V}^* = \mathbf{U} \begin{bmatrix} \mathbf{C}_{r \times r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}_{M \times K} \mathbf{V}^*,$$

where  $\mathbf{C}$  is nonsingular,  $\mathbf{U}_{M \times M}$  is unitary, and  $\mathbf{V}_{K \times K}$  is unitary. Moreover, the first  $r$  columns of  $\mathbf{U}$  represent an orthonormal basis for  $R(\mathbf{A})$ , and the last  $M - r$  columns of  $\mathbf{U}$  represent an orthonormal basis for  $N(\mathbf{A}^T)$ . Similarly, the first  $r$  columns of  $\mathbf{V}$  represent an orthonormal basis for  $R(\mathbf{A}^T)$ , and the last  $K - r$  columns of  $\mathbf{V}$  represent an orthonormal basis for  $N(\mathbf{A})$ .

For the finite mixture model problem, Eqs. (3.23) and (3.22) indicate that the nullspace of  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}]$  has dimension one, as desired, and the vector  $[\mathbf{a}^T | -1]^T$  in Eq. (3.20) must lie in that subspace. Therefore, the last column of  $\mathbf{V}$ ,  $\mathbf{v}_{K+1} \in N([\hat{\mathbf{S}}|\hat{\mathbf{r}}])$ , contains the TLS solution. Partition the vector  $\mathbf{v}_{K+1}$  such that  $\mathbf{v}_{K+1} = [\mathbf{y}^T | \alpha]^T$ .

Observe from Eq. (3.17) that the vector  $[\hat{\mathbf{a}}^T | -1]^T$  in the nullspace of  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}]$  is sought. By construction,  $\mathbf{v}_{K+1}$  lies in the nullspace of  $[\hat{\mathbf{S}}|\hat{\mathbf{r}}]$ . The TLS solution is found by equating the vector  $[\mathbf{a}^T | -1]^T$  with  $\mathbf{v}_{K+1}$  and solving for  $\mathbf{a}$ . Mathematically,

$$\begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} = \mathbf{v}_{K+1} \Rightarrow \hat{\mathbf{a}}_{tls} = \frac{-\mathbf{y}}{\alpha}, \quad (3.26)$$

and  $\hat{\mathbf{a}}_{tls}$  is the TLS solution.

### 3.2.3 Closed Form TLS Solution

A closed form of the unique TLS solution exists if the smallest singular value of  $\bar{\mathbf{S}}$ ,  $\bar{\sigma}_K$ , is strictly greater than  $\sigma_{K+1}$ . First, observe that the  $i^{th}$  right singular vector of  $[\bar{\mathbf{S}}|\mathbf{r}]$  is an eigenvector of  $[\bar{\mathbf{S}}|\mathbf{r}]^T[\bar{\mathbf{S}}|\mathbf{r}]$  and  $\sigma_i^2$  is the associated eigenvalue for  $i = 1, \dots, K$ . Then,  $\hat{\mathbf{a}}_{tls}$  satisfies the eigenvector equations corresponding to the eigenvalue  $\sigma_{K+1}^2$ , since Eq. (3.26) concludes that  $[\mathbf{a}^T | -1]^T$  lies along the space spanned by  $\mathbf{v}_{K+1}$ .

The eigenvector equations associated with eigenvalue  $\sigma_{K+1}^2$  are

$$[\bar{\mathbf{S}}|\mathbf{r}]^T[\bar{\mathbf{S}}|\mathbf{r}] \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{S}}^T \bar{\mathbf{S}} & \bar{\mathbf{S}}^T \mathbf{r} \\ \mathbf{r}^T \bar{\mathbf{S}} & \mathbf{r}^T \mathbf{r} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} = \sigma_{K+1}^2 \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix}.$$

This leads to the two systems of equations

$$\bar{\mathbf{S}}^T \bar{\mathbf{S}} \mathbf{a} - \bar{\mathbf{S}}^T \mathbf{r} = \sigma_{K+1}^2 \mathbf{a} \quad (3.27)$$

$$\mathbf{r}^T \bar{\mathbf{S}} \mathbf{a} - \mathbf{r}^T \mathbf{r} = -\sigma_{K+1}^2, \quad (3.28)$$

and the TLS solution is found from Eq. (3.27) to be

$$\hat{\mathbf{a}}_{tls} = (\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \bar{\mathbf{S}}^T \mathbf{r}. \quad (3.29)$$

Making use of the SVD of  $\bar{\mathbf{S}}$ , we find that the TLS solution is given by

$$\hat{\mathbf{a}}_{tls} = (\bar{\mathbf{V}} \bar{\mathbf{\Lambda}}^T \bar{\mathbf{\Lambda}} \bar{\mathbf{V}}^T - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \bar{\mathbf{V}} \bar{\mathbf{\Lambda}}^T \bar{\mathbf{U}}^T \mathbf{r} \quad (3.30)$$

$$= (\bar{\mathbf{V}} \bar{\mathbf{\Lambda}}_K^2 \bar{\mathbf{V}}^T - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \bar{\mathbf{V}} \bar{\mathbf{\Lambda}}^T \bar{\mathbf{U}}^T \mathbf{r} \quad (3.31)$$

$$= \bar{\mathbf{V}} (\bar{\mathbf{\Lambda}}_K^2 - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \bar{\mathbf{\Lambda}}_K \bar{\mathbf{U}}_K^T \mathbf{r} \quad (3.32)$$

$$= \sum_{i=1}^K \frac{\bar{\sigma}_i \bar{\mathbf{u}}_i^T \mathbf{r}}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} \bar{\mathbf{v}}_i \quad (3.33)$$

From this formulation of the TLS solution, it is clear that if  $\bar{\sigma}_K < \sigma_{K+1}$ , then the matrix  $\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \sigma_{K+1}^2 \mathbf{I}_M$  is not invertible. As  $\sigma_{K+1}$  approaches zero the TLS solution approaches the LS solution in Eq. (3.13). As a result, the TLS solution is equal to the LS solution only if the assumption of inconsistency is false and  $\mathbf{r} \in R(\bar{\mathbf{S}})$ .

Substituting  $\hat{\mathbf{a}}_{tls}$  into Eq. (3.28) shows that the choice of  $\sigma_{K+1}$  indicates that  $\hat{\mathbf{a}}_{tls}$  also satisfies the Euclidean norm of the LS residual  $\|\mathbf{r} - \bar{\mathbf{S}} \mathbf{a}\|_2^2$ , defined by the right hand side of Eq. (3.14). Using the SVD of  $\bar{\mathbf{S}}$  defined in Eq. (3.11) and substituting  $\mathbf{a}_{tls}$  from Eq. (3.29) into Eq. (3.28) leads to the following

$$\mathbf{r}^T \bar{\mathbf{U}} \bar{\mathbf{\Lambda}} \bar{\mathbf{V}}^T (\bar{\mathbf{V}} \bar{\mathbf{\Lambda}}_K^2 \bar{\mathbf{V}}^T - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \bar{\mathbf{V}} \bar{\mathbf{\Lambda}}^T \bar{\mathbf{U}}^T \mathbf{r} - \mathbf{r}^T \mathbf{r} = -\sigma_{K+1}^2, \quad (3.34)$$

which reduces to

$$\sigma_{K+1}^2 + \mathbf{z}^T (\bar{\mathbf{\Lambda}}_K^2 - \sigma_{K+1}^2 \mathbf{I}_K)^{-1} \mathbf{z} = \mathbf{r}^T \mathbf{r}, \quad (3.35)$$

where  $\mathbf{z} = \bar{\mathbf{\Lambda}}^T \bar{\mathbf{U}}^T \mathbf{r}$ . This may be written as

$$\sigma_{K+1}^2 + \sum_{i=1}^K \frac{\bar{\sigma}_i^2 (\bar{\mathbf{u}}_i^T \mathbf{r})^2}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} = \sum_{i=1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2, \quad (3.36)$$

where  $\bar{\sigma}_i^2 (\bar{\mathbf{u}}_i^T \mathbf{r})^2 = [\mathbf{z}^T \mathbf{z}]_i$  and the right hand side uses  $\mathbf{r}^T \mathbf{r} = \mathbf{r}^T \bar{\mathbf{U}} \bar{\mathbf{U}}^T \mathbf{r}$ . Further simplification of Eq. (3.36) follows.

$$\sigma_{K+1}^2 + \sum_{i=1}^K \frac{\bar{\sigma}_i^2 (\bar{\mathbf{u}}_i^T \mathbf{r})^2}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} - \sum_{i=1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2 = 0 \iff \quad (3.37)$$

$$\sigma_{K+1}^2 + \sum_{i=1}^K \frac{\bar{\sigma}_i^2 (\bar{\mathbf{u}}_i^T \mathbf{r})^2 - (\bar{\mathbf{u}}_i^T \mathbf{r})^2 (\bar{\sigma}_i^2 - \sigma_{K+1}^2)}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} - \sum_{i=K+1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2 = 0 \iff \quad (3.38)$$

$$\sigma_{K+1}^2 + \sum_{i=1}^K \frac{\sigma_{K+1}^2 (\bar{\mathbf{u}}_i^T \mathbf{r})^2}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} = \sum_{i=K+1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2 \iff \quad (3.39)$$

$$\sigma_{K+1}^2 \left( 1 + \sum_{i=1}^K \frac{(\bar{\mathbf{u}}_i^T \mathbf{r})^2}{\bar{\sigma}_i^2 - \sigma_{K+1}^2} \right) = \sum_{i=K+1}^M (\bar{\mathbf{u}}_i^T \mathbf{r})^2 \quad (3.40)$$

First, notice that the right side of Eq. (3.40) is equivalent to the right side of Eq. (3.14). Indeed, this indicates that  $\sigma_{K+1}$  minimizes Euclidean norm of the residual  $\|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2^2$ .

### 3.2.4 Relationship Between the TLS and LS Solutions

The relationship between the LS solution  $\mathbf{a}_{ls}$  and the TLS solution  $\mathbf{a}_{tls}$  is worth comment. Since  $\mathbf{a}_{ls}$  satisfies the normal equations in Eq. (3.4), the TLS solution is given in terms of the LS solution by

$$\mathbf{a}_{tls} = (\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \sigma_{K+1}^2 \mathbf{I}_M)^{-1} \bar{\mathbf{S}}^T \bar{\mathbf{S}} \mathbf{a}_{ls} \quad (3.41)$$

$$= \left( \mathbf{I}_M - \sigma_{K+1}^2 (\bar{\mathbf{S}}^T \bar{\mathbf{S}})^{-1} \right)^{-1} \mathbf{a}_{ls} \quad (3.42)$$

$$= \left( \mathbf{I}_M + \sigma_{K+1}^2 (\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \sigma_{K+1}^2 \mathbf{I}_M)^{-1} \right) \mathbf{a}_{ls}, \quad (3.43)$$

where the last equality is given in Corollary 6.1 of [33]. It is clear from this relationship that  $\sigma_{K+1}$  distinguishes the relationship of the TLS and LS solutions. As  $\sigma_{K+1}$  increases, the two solutions become significantly different.

### 3.2.5 TLS solution with Weighting Matrices

The TLS solution with weighting matrices is straightforward if the SVD in Eq. (3.21) is instead computed for  $\mathbf{D}[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}$  from Eq. (3.20). Let the matrix  $\mathbf{T}$  be partitioned as

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{t}_{12} \\ \mathbf{t}_{21}^T & t_{K+1} \end{bmatrix},$$

where  $\mathbf{t}_{12}$  and  $\mathbf{t}_{21}$  are  $K \times 1$  vectors and  $t_{K+1}$  is a scalar. Then, the generalized TLS solution is given by

$$\mathbf{T}^{-1} \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix} = \hat{\mathbf{v}}_{K+1} \Rightarrow \hat{\mathbf{a}}_{gtls} = \frac{-\mathbf{T}_{11} \hat{\mathbf{y}}}{\hat{\alpha} t_{K+1}}, \quad (3.44)$$

where  $\hat{\mathbf{v}}_{K+1}$  is the right singular vector associated with the smallest singular value of  $\mathbf{D}[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}$  and  $\hat{\mathbf{v}}_{K+1} = [\hat{\mathbf{y}}^T|\hat{\alpha}]^T$ .

Observe that the matrix  $\mathbf{D}$  in Eq. (3.20) does not directly appear in the definition of the estimate of  $\mathbf{a}$  in Eq. (3.44). However, one must consider that the SVD is computed on the matrix  $\mathbf{D}[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}$  instead of the matrix  $[\bar{\mathbf{S}}|\mathbf{r}]\mathbf{T}$ . The two decompositions will produce a vector in their respective nullspaces, but it is unlikely that they are equal unless  $\mathbf{D} = \mathbf{I}_M$ .

### 3.2.6 Remarks

It has been assumed throughout this discussion of the LS and TLS problems that the matrix  $\bar{\mathbf{S}}$  has rank  $K$ . However,  $\bar{\sigma}_K$  could be approximately zero. If  $\bar{\sigma}_K \approx 0$ , then the matrix  $\bar{\mathbf{S}}$  is said to be rank-deficient. Furthermore, the condition number  $\kappa(\bar{\mathbf{S}})$  of  $\bar{\mathbf{S}}$  becomes very large as  $\bar{\sigma}_K$  approaches zero. Recall that the Euclidean norm condition number  $\kappa(\bar{\mathbf{S}})$  of the matrix  $\bar{\mathbf{S}}$  is given, in general, by

$$\kappa(\bar{\mathbf{S}}) = \frac{\bar{\sigma}_1}{\bar{\sigma}_r}, \quad (3.45)$$

where  $r$  is the rank of  $\bar{\mathbf{S}}$ . The condition number indicates the sensitivity of the matrix  $\bar{\mathbf{S}}$ . Large condition numbers suggest very small  $\bar{\sigma}_K$ . In such cases, small relative changes in the matrix  $\bar{\mathbf{S}}$  cause large relative changes in the Moore-Penrose inverse of  $\bar{\mathbf{S}}$ .

When  $\kappa(\bar{\mathbf{S}})$  is very large, Eq. (3.13) indicates that the LS solution becomes very sensitive to small changes in the  $\bar{\mathbf{S}}$ . Consequently, very inaccurate LS solutions are likely to result. For the TLS solution, the perturbations,  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ , and likewise  $\bar{\sigma}_{K+1}$  must be very small when  $\bar{\sigma}_K \approx 0$ . The condition number of the matrix clearly is important when evaluating the LS and TLS solutions.

While the TLS method accounts for both the signal independent and signal dependent perturbations, the nonnegativity and sum-to-one constraints are not satisfied. Set theoretic estimation can incorporate these additional constraints as well as the TLS constraint.

## 3.3 Set Theoretic Estimation

The LS and TLS methods provide unconstrained solutions to Eq. (1.14), however the finite mixture model assumes that the proportions are nonnegative and sum-to-one.

Set theoretic estimation may be designed to find a solution satisfying these additional constraints while solving either the LS or TLS problem. This section provides an overview of set theoretic estimation with an emphasis on projections onto convex sets (POCS). Much of the subsequent discussion of set theoretic estimation will follow [11] and emphasize the aspects pertinent to the finite mixture problem. Further analysis of set theoretic estimation may be found in [36].

### 3.3.1 Method Overview

Set theoretic estimation finds a solution in the intersection of sets describing prior knowledge about the system under investigation. The motivation for adopting this method over classical estimation methods that solve an optimization problem rests in the variability in the interpretation of a problem. Unsurprisingly, different interpretations of a system produce different optimization problems and, thus, different solutions. Any of these solutions may be feasible. Thus, one solution should not be emphasized over another. Since set theoretic estimation offers a set of feasible solutions to a problem, any solution in this set satisfies all the prior knowledge about the system. In contrast to classical optimization problems, set theoretic estimation does not guarantee a unique solution.

To be precise, set theoretic estimation incorporates three types of information to determine the acceptability of a candidate solution. The first type is any known information describing a desirable estimate. The constraint that the resulting vector is nonnegative is frequently encountered in image processing applications [12]. In fact, the finite mixture density problem requires such a constraint. The second type of information to consider pertains to the system. Perhaps, it is known that an observation  $\mathbf{r} \in \mathbb{R}^M$  is related to the true solution  $\mathbf{a} \in \mathbb{R}^K$  by  $\mathbf{r} = T(\mathbf{a})$ , for some operator  $T(\cdot) : \mathbb{R}^K \rightarrow \mathbb{R}^M$ . Then, a feasible estimate  $\hat{\mathbf{a}}$  must satisfy  $\mathbf{r} = T(\hat{\mathbf{a}})$ . The third type of knowledge may be related to external sources, such as measurement noise. For example, if a noise source  $\boldsymbol{\eta}$  is present when observing  $\mathbf{r}$ , then the model previously considered becomes  $\mathbf{r} = T(\mathbf{a}) + \boldsymbol{\eta}$ . It would be desirable to choose an estimation  $\hat{\mathbf{a}}$  such that the residual  $\mathbf{r} - T(\hat{\mathbf{a}})$  is consistent with some of the noise characteristics of  $\boldsymbol{\eta}$ . These three types of information may be described by sets. An example illustrates the construction of a set consistent with a given constraint.

Consider the case of finding an estimate  $\hat{\mathbf{a}} \in \mathbb{R}^K$  such that the residual  $\mathbf{r} - T(\hat{\mathbf{a}})$  satisfies the variance of  $\boldsymbol{\eta} \in \mathbb{R}^M$ . Suppose that  $\boldsymbol{\eta}$  is a zero mean, independent and identically

distributed random variable with covariance  $\Sigma_\eta = \sigma_\eta^2 \mathbf{I}$ . A set that is consistent with the variance constraint could have the form

$$S_\eta = \{\mathbf{a} \in \mathbb{R}^K \mid \|\mathbf{r} - T(\mathbf{a})\|^2 \leq M\sigma_\eta^2\}. \quad (3.46)$$

This set describes all vectors  $\mathbf{a}$  that yields a variance of the residual that is at or below the threshold  $M\sigma_\eta^2$ .

Generally, a problem imposes several constraints on a solution. Suppose there are  $C$  known pieces of information about a desirable solution among the three types previously described. Let  $S_c$  be a set describing the  $c^{\text{th}}$  piece of knowledge. Given the sets  $\{S_c\}_{c=1}^C$ , the set theoretic estimate will be any element of  $S$ , the intersection of the  $C$  sets, or

$$S = \bigcap_{c=1}^C S_c. \quad (3.47)$$

Any estimate in the set  $S$  is considered a feasible solution, for it possesses the properties described by the  $C$  sets.

Now, having considered the formulation of the  $C$  sets, a method is needed to find a point in  $S$ . The method of successive projections onto convex sets (POCS) is an iterative algorithm to find a point in  $S$ . For POCS, the sets must be closed and convex. Furthermore, it is assumed that the sets have a common intersection. The update  $\hat{\mathbf{a}}_{i+1}$  at iteration  $i + 1$  is found by sequentially projecting onto the  $C$  sets. Let  $P_c(\mathbf{a})$  denote the projection of  $\mathbf{a}$  onto the set  $S_c$ , i.e.

$$P_c(\mathbf{a}) = \begin{cases} \mathbf{a} & \mathbf{a} \in S_c \\ \hat{\mathbf{a}} & \mathbf{a} \notin S_c \end{cases}, \quad (3.48)$$

where  $\hat{\mathbf{a}}$  satisfies

$$\min_{\hat{\mathbf{a}}} d(\mathbf{a}, \hat{\mathbf{a}}) \quad \text{subject to } \hat{\mathbf{a}} \in S_c, \quad (3.49)$$

for some metric  $d(\cdot, \cdot)$ . Then, the updated estimate  $\hat{\mathbf{a}}_{i+1}$  at iteration  $i + 1$  is given by

$$\hat{\mathbf{a}}_{i+1} = P_C (P_{C-1} (\cdots P_2 (P_1 (\hat{\mathbf{a}}_i))). \quad (3.50)$$

Though the POCS formulation guarantees convergence to a point in  $S$ , the number of iterations is not necessarily finite. Therefore, several criteria for convergence may be used to terminate a POCS iteration. Obviously, if the estimate at iteration  $i$  is in  $S$ , then a feasible solution has been found. An alternative approach is to enlarge the sets by a small

amount  $\epsilon_c > 0$ . Then, an estimate  $\mathbf{a}_i$  is feasible if it is within  $\epsilon_c$  of the sets. Moreover, the relaxation  $\epsilon_c$  may vary for each set  $S_c$ . Convergence may also be achieved if

$$\|\hat{\mathbf{a}}_{i+1} - \hat{\mathbf{a}}_i\| < \epsilon_p, \quad (3.51)$$

for some threshold  $\epsilon_p > 0$ . This suggests that if the projections for iteration  $i + 1$  perturb the estimate  $\hat{\mathbf{a}}_i$  less than  $\epsilon_p$ , then the algorithm has converged. However, this criterion could prematurely terminate the POCS iteration. As the number of sets  $C$  increases, the geometry of  $S$  becomes quite complex, and the update at iteration  $i + 1$  could satisfy Eq. (3.51) but not lie in  $S$ . Alternatively, the iteration could be terminate when the change with regard to a particular constraint satisfies a certain threshold. Consider the constraint described the set  $S_\eta$  in Eq. (3.46). Convergence may be achieved when

$$\left| (\|r - T(\hat{\mathbf{a}}_{i+1})\|^2 - M\sigma_\eta^2) - (\|r - T(\hat{\mathbf{a}}_i)\|^2 - M\sigma_\eta^2) \right| = \left| \|r - T(\hat{\mathbf{a}}_{i+1})\|^2 - \|r - T(\hat{\mathbf{a}}_i)\|^2 \right| < \epsilon_\eta, \quad (3.52)$$

for some  $\epsilon_\eta > 0$ . Again, this criterion could be met and the current estimate may not lie in  $S$ . Thus, it is imperative to take caution when establishing a criterion to stop the iteration.

### 3.3.2 Closed and Convex Sets

The restriction with POCS is the need for sets that are both closed and convex. Closed sets include boundary points, and thus, a projection onto a set boundary is also a projection to an element in that set. For example, the set defined in Eq. (3.46) is a closed ball in  $\mathbb{R}^M$  with radius  $M\sigma_\eta^2$ .

The restriction to convex sets guarantees that a point  $\mathbf{a} \notin S_c$  has a unique projection onto the set  $S_c$ . A set is convex if for  $\alpha \in [0, 1]$  and the points  $\mathbf{a}_1, \mathbf{a}_2 \in S_c$ , then the point  $\mathbf{a}_3 = \alpha\mathbf{a}_1 + (1 - \alpha)\mathbf{a}_2$  is in  $S_c$ . A simple closed and convex set is

$$S_c = \{\mathbf{a} \mid \|\mathbf{a}\| \leq \delta\}. \quad (3.53)$$

For this set, closure is obvious, and convexity is easily verified by

$$\|\mathbf{a}_3\| = \|\alpha\mathbf{a}_1 + (1 - \alpha)\mathbf{a}_2\| \leq \alpha\|\mathbf{a}_1\| + (1 - \alpha)\|\mathbf{a}_2\| \leq \delta. \quad (3.54)$$

Nonconvex sets may be incorporated. Coverage of the use of nonconvex sets is found in [36].

### 3.3.3 Projection Methods

According to [11], two common projection methods used for set theoretic estimation algorithms are serial [37] and parallel [38]. Either of these methods may be incorporated to find the updated estimate  $\hat{\mathbf{a}}_{i+1}$  at iteration  $i+1$ . However, the sets specified may encourage the implementation of one method over the other. A brief overview of the two methods adapted from [11] is presented here. See [11] and the references therein for a more thorough discussion.

The serial projection method is attributed to Kaczmarz [37]. This method performs successive projections onto each of the  $C$  sets at each iteration. For  $C$  sets, define

$$P(\hat{\mathbf{a}}_i) = P_C (P_{C-1} (\cdots P_2 (P_1 (\hat{\mathbf{a}}_i)))) , \quad (3.55)$$

where  $P_c(\mathbf{a})$  is defined by Eq. (3.48). The update at iteration  $i+1$  is given by

$$\hat{\mathbf{a}}_{i+1} = P(\hat{\mathbf{a}}_i). \quad (3.56)$$

An illustration of this sequence is shown in Figure 3.1 for two sets defined by the hyperplanes  $S_1$  and  $S_2$ . When the sets are defined as half-spaces, such as Eq. (3.53), the serial method may be extended to the relaxed method of Agmon-Motzkin-Schoenberg [11]. This method incorporates a relaxation parameter  $\lambda \in (0, 2]$ , and the update at iteration  $i+1$  is given by

$$\hat{\mathbf{a}}_{i+1} = \hat{\mathbf{a}}_i + \lambda_i (P(\hat{\mathbf{a}}_i) - \hat{\mathbf{a}}_i). \quad (3.57)$$

For a set  $S_1$  the affect of  $\lambda$  is illustrated in Figure 3.2. As shown,  $\lambda = 2$  corresponds to a reflection,  $\lambda \in (1, 2)$  to an overprojection,  $\lambda = 1$  to a projection, and  $\lambda \in (0, 1)$  to an underprojection. Of course,  $\lambda$  may be defined for each set at each iteration. The unrelaxed update  $\lambda = 1$  is frequently implemented.

The parallel projection method is based on the reflection method by Cimmino [38]. The method first computes the reflection of the current estimate for each of the  $C$  sets. Then, the average of the  $C$  reflections is the update at iteration  $i+1$ . Figure 3.3 illustrates the parallel projection scheme for the hyperplanes  $S_1$  and  $S_2$ . Mathematically, the update at iteration  $i+1$  is given by

$$\hat{\mathbf{a}}_{i+1} = \left[ \frac{2}{C} \sum_{c=1}^C P_c(\hat{\mathbf{a}}_i) \right] - \hat{\mathbf{a}}_i. \quad (3.58)$$

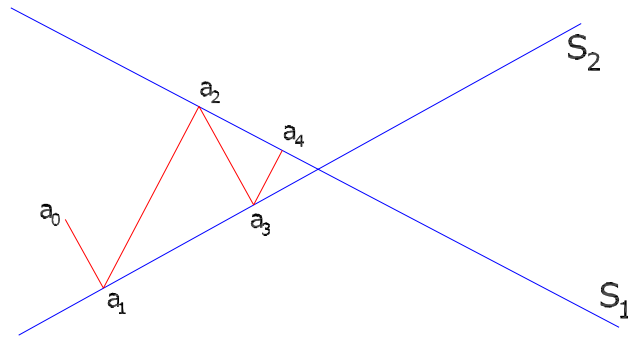


Figure 3.1: Illustration of Kaczmarz's Projection Method

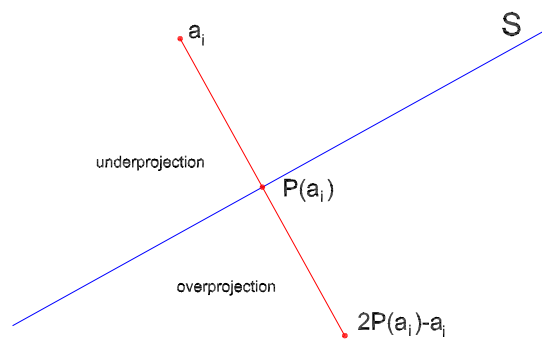


Figure 3.2: Illustration of the Effect of Relaxation Parameter  $\lambda$  on Projections

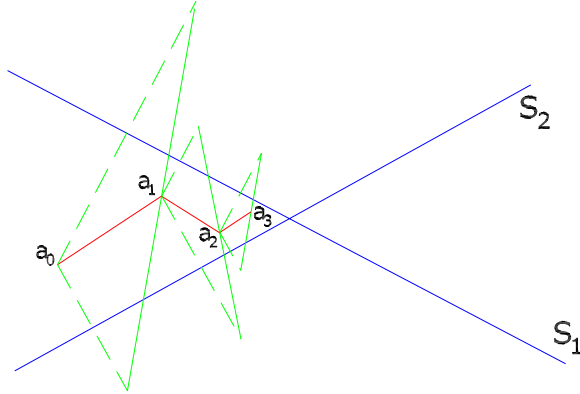


Figure 3.3: Illustration of Cimmino's Projection Method

The  $C$  reflections may be weighted to alter the rate of convergence [11, 35]. An important consequence of the parallel method is illustrated in Figure 3.3. At iteration  $i$ , the current update is not guaranteed to satisfy any of the constraints. However, the parallel projection method was shown to provide faster convergence to a feasible solution in [39]. This behavior is illustrated in Figures 3.1 and 3.3, where both projection methods begin with the same initial estimate. Notice that the result of Cimmino's projection method is closer to the intersection of the two hyperplanes after three projections.

### 3.3.4 Effect of Initial Estimate and the Order of Projection

The choice of the initial estimate and the order of projection affects the performance of the POCS method [40, 41]. In general, optimization methods exhibit some sensitivity to the initial estimate, especially when several local extrema points exist in the function to be optimized. A similar phenomenon is observed with the POCS method.

Figure 3.4, based on a similar illustration in [40], demonstrates the consequence of the initial estimate and the order of projection using Kaczmarz's sequential projection method. Three convex sets are denoted by the rectangle (1), the triangle (2), and the circle (3). The shaded region indicates the intersection of the three sets. When  $\mathbf{x}_0$  is the initial estimate and the order of projection is 1, 2, 3, then the feasible estimate  $\mathbf{z}_1$  is determined in two iterations. Changing the order of projection to 3, 2, 1 and starting with the same point,  $\mathbf{x}_0$  leads to the estimate  $\mathbf{z}_2$  in one projection. Following the original order of projection, 1, 2, 3,

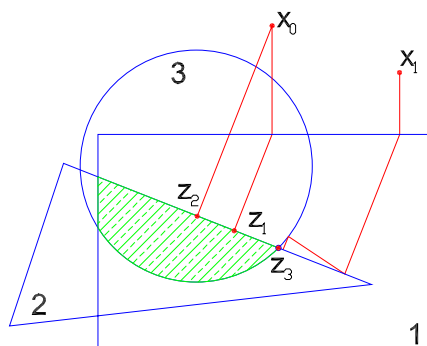


Figure 3.4: Illustration of the Effect of Initial Estimate and the Order of Projection

the initial estimate  $\mathbf{x}_1$  converges to the limit point  $\mathbf{z}_3$ . This highlights an important aspect of the POCS method: convergence to a point in a nonempty intersection is guaranteed, but the number of projections need not be finite. In fact, the situation illustrated motivates the alternative convergence criteria mentioned in Section 3.3.1. Thus, the initial estimate and the order of projection significantly impact the performance of the POCS method.

## Chapter 4

# Estimation of Finite Mixture

## Models with Nonparametric

## Components

The estimation of finite mixture models using approximations of the component densities has been presented as a constrained optimization problem solving Eq. (1.14) subject to the constraints in Eq. (1.15). The straightforward estimates provided by classical LS and TLS methods fail to satisfy the known properties of the proportions  $\mathbf{a}$ . The flexibility of set theoretic estimation is appealing, for adding a constraint simply requires the inclusion of an appropriate set. Then, with POCS an estimate is found in the intersection of the sets. This section describes the POCS formulation for the finite mixture model problem. The formulation first discusses the formation of the appropriate sets given the known constraints. Then, the projections onto these sets are presented.

## 4.1 Sets for Finite Mixture Density Estimation

Three constraints must be considered for the finite mixture model. First, an estimate must satisfy Eq. (1.14). This constraint is represented by the noise variance set. This set is ultimately constructed from the general total least squares formulation. The second set defines the set of vectors whose elements sum to one. Finally, the third set, which actually leads to  $K$  sets, one for each element of  $\mathbf{a}$ , defines the set of nonnegative vectors.

### 4.1.1 Finite Mixture Model Set

The general finite mixture model considers errors in both  $\boldsymbol{\eta}$  and  $\Delta\mathbf{S}$ . In the LS sense, the errors are attributed only to the residual  $\boldsymbol{\eta}$ , whereas TLS accommodates both sources of error. A set consistent with the LS solution introduces the development of a new set related to the TLS problem.

#### Residual Variance Set $S_v$ Motivated by Least Squares Optimization

The residual variance set based on the constrained LS solution is considered in [41]. Accordingly, the set ignores the perturbations attributed to  $\Delta\mathbf{S}$ . The set simply assumes that the variance of the residual  $\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}$  must not exceed the variance of the noise  $\boldsymbol{\eta}$ . Assuming the noise  $\boldsymbol{\eta}$  is independent and identically distributed, the set is defined in the context of the finite mixture model formulation as

$$S_v = \{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 \leq \delta_\eta \}, \quad (4.1)$$

where  $\delta_\eta = M\sigma_\eta^2$  is the variance of the Euclidean norm of the residual  $\|\boldsymbol{\eta}\|_2$ . This set implicitly assumes that the measurement errors have equal variance. Excluding the presence of the the perturbation  $\Delta\mathbf{S}$ , this set assumes the static model, i.e.  $\bar{\mathbf{S}} = \mathbf{S}$ , is appropriate. Following the nomenclature introduced in [42], a POCS formulation incorporating  $S_v$  is called a static POCS (SPOCS) scheme.

#### Dynamic Residual Set $S_{v'}$

Since  $\mathbf{S}$  is estimated from samples, the statistics of the error  $\Delta\mathbf{S}$  should be incorporated into the noise variance set. The perturbation  $\Delta\mathbf{S}$  is considered in [42] by writing

the finite mixture model in Eq. (1.14) as

$$\mathbf{r} = \bar{\mathbf{S}}\mathbf{a} + \boldsymbol{\eta} + \Delta\mathbf{S}\mathbf{a}. \quad (4.2)$$

Inspired by the constrained LS solution, the set assumes that the variance of the residual  $\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}$  must not exceed  $E\|\boldsymbol{\eta} + \Delta\mathbf{S}\mathbf{a}\|_2^2$ . The expected value of the norm of this residual follows as

$$E\|\boldsymbol{\eta} + \Delta\mathbf{S}\mathbf{a}\|_2^2 = E\|\boldsymbol{\eta}\|_2^2 + E\|\Delta\mathbf{S}\mathbf{a}\|_2^2,$$

where the cross term  $2E\|\boldsymbol{\eta}\Delta\mathbf{S}\mathbf{a}\|$  is zero, assuming the noise sources are uncorrelated with zero mean. The authors of [42] acknowledge the lack of prior knowledge to estimate  $E\|\Delta\mathbf{S}\mathbf{a}\|$ , and rewrite the bound using the Cauchy-Bunyakovskii-Schwartz<sup>1</sup> where  $E\|\Delta\mathbf{S}\mathbf{a}\|_2^2 \leq \|\mathbf{a}\|_2^2 E\|\Delta\mathbf{S}\|_2^2$ . The term  $E\|\Delta\mathbf{S}\|_2^2$  is defined to be the largest eigenvalue of  $\Delta\mathbf{S}^T\Delta\mathbf{S}$ , the induced matrix 2-norm of  $\Delta\mathbf{S}$  [35].

The approximation of  $E\|\Delta\mathbf{S}\|_2^2$  requires a closer examination of the matrix  $\Delta\mathbf{S}^T\Delta\mathbf{S}$ . For  $K$  classes, the matrix  $\Delta\mathbf{S}^T\Delta\mathbf{S}$  has the form

$$\Delta\mathbf{S}^T\Delta\mathbf{S} = \begin{bmatrix} \Delta\mathbf{s}_1^T\Delta\mathbf{s}_1 & \Delta\mathbf{s}_1^T\Delta\mathbf{s}_2 & \Delta\mathbf{s}_1^T\Delta\mathbf{s}_3 & \cdots & \Delta\mathbf{s}_1^T\Delta\mathbf{s}_{K-1} & \Delta\mathbf{s}_1^T\Delta\mathbf{s}_K \\ \Delta\mathbf{s}_2^T\Delta\mathbf{s}_1 & \Delta\mathbf{s}_2^T\Delta\mathbf{s}_2 & \Delta\mathbf{s}_2^T\Delta\mathbf{s}_3 & \cdots & \Delta\mathbf{s}_2^T\Delta\mathbf{s}_{K-1} & \Delta\mathbf{s}_2^T\Delta\mathbf{s}_K \\ \vdots & \ddots & \ddots & \cdots & \ddots & \vdots \\ \Delta\mathbf{s}_K^T\Delta\mathbf{s}_1 & \Delta\mathbf{s}_K^T\Delta\mathbf{s}_2 & \Delta\mathbf{s}_K^T\Delta\mathbf{s}_3 & \cdots & \Delta\mathbf{s}_K^T\Delta\mathbf{s}_{K-1} & \Delta\mathbf{s}_K^T\Delta\mathbf{s}_K \end{bmatrix}, \quad (4.3)$$

where  $\Delta\mathbf{s}_k$  denotes the  $k^{\text{th}}$  column vector of the matrix  $\Delta\mathbf{S}$ . For  $j \neq k$ , the perturbations  $\Delta\mathbf{s}_j$ , associated with class  $j$ , and  $\Delta\mathbf{s}_k$ , associated with class  $k$  are uncorrelated. Therefore, the expected value of  $\Delta\mathbf{S}^T\Delta\mathbf{S}$  is a diagonal matrix, and the diagonal elements also correspond to the eigenvalues of  $E\{\Delta\mathbf{S}^T\Delta\mathbf{S}\}$ . The expected value of the  $k^{\text{th}}$  element on the diagonal is given by

$$E\{\Delta\mathbf{s}_k^T\Delta\mathbf{s}_k\} = E\sum_{m=1}^M[\Delta\mathbf{s}_k]_m^2 = \sum_{m=1}^M E[\Delta\mathbf{s}_k]_m^2 = \sum_{m=1}^M \sigma_{k,m}^2, \quad (4.4)$$

where  $\sigma_{k,m}^2$  denotes the variance of the  $m^{\text{th}}$  element of class  $k$ . The diagonals of the sample autocovariance matrix  $\mathbf{C}_k$  of the  $k^{\text{th}}$  class contains estimates of  $\sigma_{k,m}^2$ . Thus,  $E\{\Delta\mathbf{s}_k^T\Delta\mathbf{s}_k\}$  may be approximated by the trace of  $\mathbf{C}_k$  and

$$\rho = \max_k \{\text{tr}(\mathbf{C}_k)\}, \quad (4.5)$$

---

<sup>1</sup>This is commonly referred to as the Cauchy-Schwartz inequality, however it was noted in [35] on page 271 that Bunyakovskii's contribution was overlooked for many years.

and  $\rho$  approximates  $E\|\Delta\mathbf{S}\|_2^2$ .

Let  $\delta_{\mathbf{a}}$  denote the largest possible value of  $\|\mathbf{a}\|_2^2$ . The set  $S_v$  may be enlarged to incorporate the sources of error  $\Delta\mathbf{S}\mathbf{a}$ . This modified set is given by

$$S_{v'} = \{ \mathbf{a} \mid \|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2^2 \leq \delta'_v \}, \quad (4.6)$$

where  $\delta'_v = \delta_\eta + \delta_{\mathbf{a}}\rho$ . Since this set takes into account the perturbations  $\Delta\mathbf{S}$ , [42] calls a POCS formulation incorporating this set a dynamic POCS (DPOCS) technique. The choice of the name “dynamic” can be misleading. Notice that the set  $S_{v'}$  has the same form as the set  $S_v$  in Eq. (4.1). The difference in the definitions of the sets is the boundary,  $\delta_v$  or  $\delta'_v$ . Neither of these parameters change during the POCS iteration. Thus, the size of the sets remain constant for the POCS iteration. Though somewhat of a misnomer, the authors of [42] refer to the inclusion of the perturbations  $\Delta\mathbf{S}$  as the “dynamic” aspect of  $S_{v'}$ .

For the finite mixture model problem, the largest possible value of  $\|\mathbf{a}\|_2^2$  is one. It was suggested in [43] that  $\delta_a$  could be approximated by

$$\delta_a = \frac{\|\mathbf{r}\|_2^2}{\|\bar{\mathbf{S}}\|_2^2}. \quad (4.7)$$

This approximation could be very inaccurate, since  $\bar{\mathbf{S}}$  is estimated from samples. Furthermore, this approximation could be too restrictive and reduce the size of the set such that the actual solution is not contained in the set.

It is noted that the set  $S_{v'}$  has a larger diameter than the set  $S_v$  and implies a larger set of possible solutions. The set of feasible solutions is clearly smaller for the set  $S_v$ , and, as a consequence, the true solution may not be in the set. However, the intent of the enlarged set  $S_{v'}$  is to include the actual solution among the set of feasible solutions. As the perturbations  $\Delta\mathbf{S}$  diminish, the sets  $S_{v'}$  and  $S_v$  become equivalent. Refer to [42] for a more elaborate discussion.

### **Error-in-Variables Set $\Gamma$ Motivated by Total Least Squares Optimization**

The set  $S_{v'}$  accommodates the perturbations  $\Delta\mathbf{S}$  at the cost of creating a much larger set. Therefore, a set motivated by the weighted TLS optimization was introduced by [12]. Section 3.2.5 indicates that the weighted TLS solution permits perturbations  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$  with different magnitudes. With this in mind, the set  $S_v$  in Eq. (4.1) is revisited in [12],

and the authors note that the set  $S_v$  may be equivalently written as

$$S_v = \{ \mathbf{a} \mid \exists \boldsymbol{\eta} \ni \mathbf{S}\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \|\boldsymbol{\eta}\|_2^2 \leq \delta_\eta \} \quad (4.8)$$

by arguing that the set is motivated by the LS optimization problem, which solves the optimization

$$\min_{\mathbf{a}} \|\boldsymbol{\eta}\|_2^2 \quad \text{subject to } \mathbf{S}\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}. \quad (4.9)$$

The classical TLS problem solves the optimization

$$\min_{\mathbf{a}} \|[\Delta\mathbf{S}|\boldsymbol{\eta}]\|_F^2 \quad \text{subject to } (\bar{\mathbf{S}} + \Delta\mathbf{S})\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \quad (4.10)$$

where [12] indicates that the weighted TLS problem solves the optimization

$$\min_{\mathbf{a}} \tau \|\Delta\mathbf{S}\|_F^2 + \|\boldsymbol{\eta}\|_2^2 \quad \text{subject to } (\bar{\mathbf{S}} + \Delta\mathbf{S})\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \quad (4.11)$$

with a positive weight  $\tau$  determined by the relationship of the statistics of both  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ . For  $\tau = 1$ , Eq. (4.11) is equivalent to Eq. (4.10). Recall, the square of the Frobenius norm of a matrix  $\mathbf{A}$ , denoted  $\|\mathbf{A}\|_F^2$ , is defined as the trace of  $\mathbf{A}^T\mathbf{A}$ . The authors introduce a convex set based on Eq. (4.11) as

$$S_{TLS} = \left\{ \mathbf{a} \mid \exists \{ \Delta\mathbf{S}, \boldsymbol{\eta} \} \ni (\bar{\mathbf{S}} + \Delta\mathbf{S})\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \tau \|\Delta\mathbf{S}\|_F^2 + \|\boldsymbol{\eta}\|_2^2 \leq \nu \right\}, \quad (4.12)$$

where the parameters  $\tau$  and  $\nu$  are determined by statistical properties of  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ . It was shown in [12] that  $S_{TLS}$  may also be defined by

$$\Gamma = \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \frac{\nu}{\tau} \|\mathbf{a}\|_2^2 - \nu \leq 0 \right\}, \quad (4.13)$$

where  $\nu$  is fixed and chosen to satisfy

$$\nu = \tau E \|\Delta\mathbf{S}\|^2 + E \|\boldsymbol{\eta}\|_2^2, \quad (4.14)$$

where the parameter  $\tau$  is defined as

$$\tau \geq \frac{E \|\boldsymbol{\eta}\|_2^2}{\sigma_K^2(\bar{\mathbf{S}}) - E \|\Delta\mathbf{S}\|^2}, \quad (4.15)$$

where  $\sigma_K(\bar{\mathbf{S}})$  denotes the smallest singular value of  $\bar{\mathbf{S}}$ . The subscripts defining the matrix norm in the definition of  $\nu$  and  $\tau$  have been purposefully removed, since the definitions are not restricted to the Frobenius matrix norm. Note that the definitions of  $\nu$  and  $\tau$  lead to

the implicit constraint that  $E\|\Delta\mathbf{S}\| \leq \sqrt{\frac{\nu}{\tau}} \leq \sigma_K(\bar{\mathbf{S}})$ . To satisfy this implicit and restrictive condition, the induced matrix 2-norm is used to approximate  $E\|\Delta\mathbf{S}\|^2$ . The equivalence of the sets  $S_{TLS}$  and  $\Gamma$  is reproduced from [12] in Section B.1. Refer to Section B.2 for a derivation of  $\tau$ .

The POCS formulation is referred to as the error in variables POCS (EVPOCS) method if  $\Gamma$  describes the noise properties of the model. The nomenclature was introduced by [12], since the set theoretic method does not minimize the TLS optimization problem in Eq. (4.10).

The set  $\Gamma$  utilizes the relationship between the perturbations  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ . With this in mind, the set  $\Gamma$  is worth examining in further detail. Let  $\alpha \geq 1$  and define  $\hat{\tau}$  as the lower bound given in Eq. (4.15). Then,  $\tau$  can be written as  $\tau = \alpha\hat{\tau}$ . First, observe

$$\begin{aligned} \frac{\nu}{\tau} &= \frac{\tau E\|\Delta\mathbf{S}\|^2 + E\|\boldsymbol{\eta}\|_2^2}{\tau} = E\|\Delta\mathbf{S}\|^2 + \frac{E\|\boldsymbol{\eta}\|_2^2}{\tau} = E\|\Delta\mathbf{S}\|^2 + \frac{\sigma_K^2(\bar{\mathbf{S}}) - E\|\Delta\mathbf{S}\|^2}{\alpha} \\ &= \rho + \frac{\sigma_K^2(\bar{\mathbf{S}}) - \rho}{\alpha}, \end{aligned} \quad (4.16)$$

where  $\rho = E\|\Delta\mathbf{S}\|^2$ . Also, consider

$$\nu = \alpha\hat{\tau}E\|\Delta\mathbf{S}\|^2 + E\|\boldsymbol{\eta}\|_2^2 = \left( \frac{\alpha\rho}{\sigma_K^2(\bar{\mathbf{S}}) - \rho} + 1 \right) \delta_\eta, \quad (4.17)$$

where  $\delta_\eta = E\|\boldsymbol{\eta}\|_2^2$ . Substituting Eqs. (4.16) and (4.17) into the set constraint for  $\Gamma$  leads to

$$\begin{aligned} \Gamma &= \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \left( \rho + \frac{\sigma_K^2(\bar{\mathbf{S}}) - \rho}{\alpha} \right) \|\mathbf{a}\|_2^2 - \left( \frac{\alpha\rho}{\sigma_K^2(\bar{\mathbf{S}}) - \rho} + 1 \right) \delta_\eta \leq 0 \right\} \\ &= \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \left( \rho + \frac{\sigma_K^2(\bar{\mathbf{S}}) - \rho}{\alpha} \right) \|\mathbf{a}\|_2^2 \leq \left( \frac{\alpha\rho}{\sigma_K^2(\bar{\mathbf{S}}) - \rho} + 1 \right) \delta_\eta \right\}. \end{aligned} \quad (4.18)$$

The coefficient of the term  $\|\mathbf{a}\|_2^2$  in the set constraint for  $\Gamma$  approaches  $\rho$  as  $\alpha$  approaches infinity. As  $\alpha$  increases, the coefficient of the fixed term  $\delta_\eta$  increases, and, as expected, the set becomes unbounded. This indicates that increasing  $\tau$  increases the convexity of  $\Gamma$  as noted in [12]. When  $\alpha = 1$ , the set  $\Gamma$  is given by

$$\Gamma = \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \sigma_K^2(\bar{\mathbf{S}})\|\mathbf{a}\|_2^2 \leq \left( \frac{\rho}{\sigma_K^2(\bar{\mathbf{S}}) - \rho} + 1 \right) \delta_\eta \right\}. \quad (4.19)$$

Now, consider the effect of  $\rho = E\|\Delta\mathbf{S}\|^2$ . As the magnitude of the perturbations  $\Delta\mathbf{S}$  increase, while satisfying  $\sigma_K^2(\bar{\mathbf{S}}) > \rho$ , the fixed bound of the set increases. The set becomes larger.

### Generalized Error-in-Variables Set $\Gamma_g$

A limitation of the set  $\Gamma$  is the inherent coupling of the fixed boundary  $\nu$  and the coefficient  $\frac{\nu}{\tau}$ . In [12],  $\frac{\nu}{\tau}$  is required to be less than the smallest singular value of  $\bar{\mathbf{S}}$  to ensure that  $\Gamma$  is a convex set. A new, more effective set is formed by removing the coupling of the magnitudes of the perturbations  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$ . Consider the formulation of the set  $S_{\nu'}$ . Recall that this set assumes that the variance of the residual  $\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}$  must not exceed  $E\|\boldsymbol{\eta}\|_2^2 + \|\mathbf{a}\|_2^2 E\|\Delta\mathbf{S}\|_2^2$ . In light of the set  $\Gamma$ , a natural modification to the set  $S_{\nu'}$  is to vary the size of the set with respect to  $\|\mathbf{a}\|_2^2$ . The new set  $\Gamma_g$  is defined as

$$\Gamma_g = \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \rho\|\mathbf{a}\|_2^2 \leq \delta_\eta \right\}. \quad (4.20)$$

To ensure convexity, the parameter  $\rho$ , like  $\frac{\nu}{\tau}$ , must not exceed  $\sigma_K^2(\bar{\mathbf{S}})$  and the parameter  $\delta_\eta$  must be nonnegative. Choose  $\rho = E\|\Delta\mathbf{S}\|_2^2$  and  $\delta_\eta = E\|\boldsymbol{\eta}\|_2^2$ .

Observe the relationship between the sets  $\Gamma$  and  $\Gamma_g$ . For comparison, consider the set  $\Gamma$  as expanded in Eq. (4.18). Suppose  $\mathbf{a} \in \Gamma_g$ . Since  $\sigma_K^2(\bar{\mathbf{S}}) > \rho \geq 0$ , then according to Eq. (4.16)  $\frac{\nu}{\tau} > \rho \geq 0$ . It is clear that  $\mathbf{a} \in \Gamma$ , since  $\nu > \delta_\eta$ . On the other hand, suppose  $\mathbf{a} \in \Gamma$ . Again, note that  $\sigma_K^2(\bar{\mathbf{S}}) > \rho \geq 0$  and  $\frac{\nu}{\tau} > \rho \geq 0$ . Thus,  $\mathbf{a}$  is not necessarily in  $\Gamma_g$ . This indicates that the set  $\Gamma_g$  is smaller than the set  $\Gamma$ .

The set  $\Gamma_g$  generalizes the formulation for the set  $\Gamma$ . Accordingly, a POCS scheme incorporating the noise set  $\Gamma_g$  is called a generalized error-in-variables POCS (GEVPOCS) method.

#### 4.1.2 Sets Describing Properties of Feasible Contribution Estimates

In addition to satisfying the finite mixture model defined by Eq. (1.14), the contributions  $\mathbf{a}$  must also be nonnegative and sum-to-one as defined in Eq. (1.15). The finite mixture model sets presented do not guarantee these additional constraints. It is assumed that imposing these signal constraints will produce a nonempty set of feasible solutions. To include these constraints, the set  $S_a$  defined in Eq. (1.15) is decomposed into two appropriate sets. Let the set  $S_\Sigma$  define the sum-to-one constraint of the vector elements. Thus,  $S_\Sigma$  is given as the set

$$S_\Sigma = \left\{ \mathbf{a} \in \mathbb{R}^K \mid \sum_{k=1}^K [\mathbf{a}]_k = 1 \right\}. \quad (4.21)$$

The nonnegativity constraint may be decomposed into  $K$  sets, where the  $k^{\text{th}}$  set is given by

$$S_k = \{\mathbf{a} \in \mathbb{R}^K \mid [\mathbf{a}]_k \geq 0\} \quad \text{for } k = 1, 2, \dots, K. \quad (4.22)$$

## 4.2 Projection onto Sets

The projections onto the sets were determined according to Eq. (3.49), where  $d(\mathbf{a}, \hat{\mathbf{a}}) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2$  is given by the square of the Euclidean norm, and is found using Lagrange multipliers. The projections onto the noise sets,  $S_v$ ,  $S_{v'}$ ,  $\Gamma$ , and  $\Gamma_g$ , have very similar forms. However, for clarity the derivations of the all the projections are provided.

Suppose that  $\mathbf{a} \notin S_v$ , the static residual set given by Eq. (4.1). To find the closest point  $\hat{\mathbf{a}} \in S_v$  to  $\mathbf{a}$ , consider formulating the constrained function using the method of Lagrange multipliers as

$$\phi_v(\hat{\mathbf{a}}, \lambda_v) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2 + \lambda_v (\|\bar{\mathbf{S}}\hat{\mathbf{a}} - \mathbf{r}\|_2^2 - \delta_\eta). \quad (4.23)$$

The minimizing  $\hat{\mathbf{a}}_0 \in S_v$  is found with differential calculus. Thus,  $\hat{\mathbf{a}}_0$  must satisfy

$$\mathbf{D}_{\hat{\mathbf{a}}} [\phi_v(\hat{\mathbf{a}}, \lambda_v)]|_{\hat{\mathbf{a}}=\hat{\mathbf{a}}_0} = 2\hat{\mathbf{a}}_0 - 2\mathbf{a} + 2\lambda_v \bar{\mathbf{S}}^T \bar{\mathbf{S}}\hat{\mathbf{a}}_0 - 2\lambda_v \bar{\mathbf{S}}^T \mathbf{r} = 0, \quad (4.24)$$

and solving for  $\hat{\mathbf{a}}_0$  yields

$$\hat{\mathbf{a}}_0 = (\mathbf{I}_K + \lambda_v \bar{\mathbf{S}}^T \bar{\mathbf{S}})^{-1} (\mathbf{a} + \lambda_v \bar{\mathbf{S}}^T \mathbf{r}), \quad (4.25)$$

the projection onto the set  $S_v$ . The Lagrange multiplier  $\lambda_v \geq 0$  must satisfy

$$\|\bar{\mathbf{S}}\hat{\mathbf{a}}_0 - \mathbf{r}\|_2^2 - \delta_\eta = 0, \quad (4.26)$$

which is the derivative of Eq. (4.23) with respect to  $\lambda_v$ .

Suppose that  $\mathbf{a} \notin S_{v'}$ , the dynamic residual set defined in Eq. (4.6). The closest point  $\hat{\mathbf{a}} \in S_{v'}$  to  $\mathbf{a}$  is found by minimizing the function

$$\phi_{v'}(\hat{\mathbf{a}}, \lambda_{v'}) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2 + \lambda_{v'} (\|\bar{\mathbf{S}}\hat{\mathbf{a}} - \mathbf{r}\|_2^2 - \delta'_{v'}), \quad (4.27)$$

with respect to  $\hat{\mathbf{a}}$ . Note that the minimizing  $\hat{\mathbf{a}}_0$  satisfies an equation similar to Eq. (4.23) replacing  $\lambda_{v'}$  with  $\lambda_v$  and  $M\sigma_\eta^2$  with  $\delta'_{v'}$ . The projection onto the set is

$$\hat{\mathbf{a}}_0 = (\mathbf{I}_K + \lambda_{v'} \bar{\mathbf{S}}^T \bar{\mathbf{S}})^{-1} (\mathbf{a} + \lambda_{v'} \bar{\mathbf{S}}^T \mathbf{r}), \quad (4.28)$$

and the parameter  $\lambda_{v'} \geq 0$  must satisfy

$$\|\bar{\mathbf{S}}\hat{\mathbf{a}} - \mathbf{r}\|_2^2 - \delta'_v. \quad (4.29)$$

To obtain the projection to the error-in-variables residual set  $\Gamma$  for a vector  $\mathbf{a} \notin \Gamma$ , minimize the function

$$\phi_{t_{ls}}(\hat{\mathbf{a}}, \lambda_{t_{ls}}) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2 + \lambda_{t_{ls}} \left( \|\bar{\mathbf{S}}\hat{\mathbf{a}} - \mathbf{r}\|^2 - \frac{\nu}{\tau} \|\hat{\mathbf{a}}\|^2 - \nu \right) \quad (4.30)$$

with respect to  $\hat{\mathbf{a}}$ . The minimizing vector  $\hat{\mathbf{a}}_0$  satisfies

$$\mathbf{D}_{\hat{\mathbf{a}}} [\phi_{t_{ls}}(\hat{\mathbf{a}}, \lambda_{t_{ls}})]|_{\hat{\mathbf{a}}=\hat{\mathbf{a}}_0} = 2\hat{\mathbf{a}}_0 - 2\mathbf{a} + 2\lambda_{t_{ls}}\bar{\mathbf{S}}^T\bar{\mathbf{S}}\hat{\mathbf{a}}_0 - 2\lambda_{t_{ls}}\bar{\mathbf{S}}^T\mathbf{r} - 2\lambda_{t_{ls}}\frac{\nu}{\tau}\hat{\mathbf{a}}_0 = 0, \quad (4.31)$$

and is given by

$$\hat{\mathbf{a}}_0 = \left[ \mathbf{I} + \lambda_{t_{ls}} \left( \bar{\mathbf{S}}^T\bar{\mathbf{S}} - \frac{\nu}{\tau}\mathbf{I} \right) \right]^{-1} (\mathbf{a} + \lambda_{t_{ls}}\bar{\mathbf{S}}^T\mathbf{r}), \quad (4.32)$$

where the parameter  $\lambda_{t_{ls}} \geq 0$  is chosen to satisfy

$$\|\bar{\mathbf{S}}\hat{\mathbf{a}}_0 - \mathbf{r}\|^2 - \frac{\nu}{\tau} \|\hat{\mathbf{a}}_0\|^2 - \nu = 0. \quad (4.33)$$

The projection of  $\mathbf{a} \notin \Gamma_g$  to the generalized error-in-variables residual set  $\Gamma_g$  minimizes the function

$$\phi_g(\hat{\mathbf{a}}, \lambda_g) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2 + \lambda_g (\|\bar{\mathbf{S}}\hat{\mathbf{a}} - \mathbf{r}\|^2 - \rho \|\hat{\mathbf{a}}\|^2 - \delta_\eta) \quad (4.34)$$

with respect to  $\hat{\mathbf{a}}$ . Note that the minimizing  $\hat{\mathbf{a}}_0$  satisfies an equation similar to Eq. (4.30) replacing  $\lambda_{t_{ls}}$  with  $\lambda_g$ ,  $\frac{\nu}{\tau}$  with  $\rho$ , and  $\nu$  with  $\delta_\eta$ . The projection onto the set is

$$\hat{\mathbf{a}}_0 = [\mathbf{I}_K + \lambda_g (\bar{\mathbf{S}}^T\bar{\mathbf{S}} - \rho\mathbf{I}_K)]^{-1} (\mathbf{a} + \lambda_g\bar{\mathbf{S}}^T\mathbf{r}). \quad (4.35)$$

The parameter  $\lambda_g \geq 0$  must satisfy

$$\|\bar{\mathbf{S}}\hat{\mathbf{a}}_0 - \mathbf{r}\|_2^2 - \rho \|\hat{\mathbf{a}}_0\|_2^2 - \delta_\eta = 0. \quad (4.36)$$

Suppose that  $\mathbf{a} \notin S_\Sigma$ , the summation to one set. The point  $\hat{\mathbf{a}} \in S_\Sigma$  closest to  $\mathbf{a}$  minimizes

$$\phi_\Sigma(\hat{\mathbf{a}}, \lambda_\Sigma) = \|\mathbf{a} - \hat{\mathbf{a}}\|_2^2 + \lambda_\Sigma (\hat{\mathbf{a}}^T \mathbf{1}_K - 1) \quad (4.37)$$

where  $\mathbf{1}_K$  is a  $K \times 1$  column vector of ones. Thus,  $\hat{\mathbf{a}}_0$  and  $\lambda_\Sigma$  must satisfy the following equations

$$\mathbf{D}_{\hat{\mathbf{a}}} [\phi_\Sigma(\hat{\mathbf{a}}, \lambda_\Sigma)]|_{\hat{\mathbf{a}}=\hat{\mathbf{a}}_0} = 2\mathbf{a} - 2\hat{\mathbf{a}}_0 + \lambda_\Sigma \mathbf{1}_K = 0 \quad (4.38)$$

$$\mathbf{D}_{\lambda_\Sigma} [\phi_\Sigma(\hat{\mathbf{a}}, \lambda_\Sigma)]|_{\hat{\mathbf{a}}=\hat{\mathbf{a}}_0} = \hat{\mathbf{a}}_0^T \mathbf{1}_K - 1, \quad (4.39)$$

which leads to the projection

$$\hat{\mathbf{a}}_0 = \mathbf{a} - \frac{1}{K} (\mathbf{a}^T \mathbf{1} - 1) \mathbf{1}_{K \times 1}. \quad (4.40)$$

The projection onto the nonnegativity set  $S_k$  is performed by replacing negative elements of  $\mathbf{a} \notin S_k$  with zeros to form  $\hat{\mathbf{a}}_0 \in S_k$  for  $k = 1, 2, \dots, K$ .

## Chapter 5

# Results

Several different simulations were designed to evaluate the performance of the set theoretic methods described to estimate the contributions of the class densities described by finite mixture models. Simulations modeled finite mixtures of either probability densities or spectral densities as described in Section 1.1. The contributions of  $K$  components were estimated from an observed mixture density by least squares (LS), total least squares (TLS), and various set theoretic formulations. Neither the LS solution nor the TLS solution is guaranteed to satisfy both the sum-to-one and nonnegativity constraints; however, set theoretic estimates satisfy both the sum-to-one and nonnegativity constraints. Estimates were found via the modified EM algorithm, where the parameters of the class densities are estimated from samples, to establish a benchmark when the observed finite mixture density is known to be composed of  $K$  parametric component probability densities.

The set theoretic methods find a feasible solution by projecting sequentially onto three sets: the finite mixture model set, the sum-to-one set, and the nonnegativity set. The iteration terminates when the estimate lies in all three sets. The four finite mixture model sets described in Section 4.1 were tested: the residual variance set given in Eq. (4.1), the dynamic residual set given in Eq. (4.6), the error-in-variables set given in Eq. (4.13), and the generalized error-in-variables set given in Eq. (4.20). The parameters chosen for the noise sets depend upon the available prior knowledge of the simulated finite mixture.

Since the initial estimate is known to affect the resulting set theoretic estimate [41], three reasonable initial estimates were considered to start the set theoretic iteration: the

LS solution, the TLS solution, and the uniform contribution estimate<sup>1</sup> of the  $K$  component densities.

While the choice of the initial estimate is somewhat subjective, it is reasonable to use an estimate that is compatible with the chosen noise set. An estimate would be compatible with a noise set if the estimate is contained in the noise set. For example, the TLS solution assumes a model with variation in the discrete estimates of the densities. The residual variance set  $S_v$  does not incorporate this variation. In general, a TLS solution is not likely to satisfy the set  $S_v$  and would require a projection to this set. The optimal initial estimate should minimize the number of iterations necessary to find a solution satisfying all the problem constraints. For instance, suppose that the TLS solution serves as the initial estimate and the dynamic residual set  $S_{v'}$  is selected. The first iteration would not require a projection to the dynamic residual set, if containment of the TLS solution in the set  $S_{v'}$  is tested first. The sum-to-one and nonnegativity sets would find the closest vectors, with regard to the Euclidean distance, to the TLS solution. Thus, the set  $S_{v'}$  prevents these other projections from diverging from the actual model. Since the noise sets were modeled from the LS and TLS interpretations of the finite mixture model, it is appropriate to use either the LS solution or TLS solution as an initial estimate. For the residual variance set, the LS solution is tested as an initial estimate. The TLS solution provides a reasonable initial estimate to test with the dynamic residual set, the error-in-variables residual set, and the generalized error-in-variables set.

The uniform estimate is appropriate for any of the noise sets, but unfortunately, the estimate is at best projected to the noise set boundary if any projection is necessary. The sum-to-one and nonnegativity sets will improve the solution only if the uniform estimate does not satisfy the noise set. Thus, when the uniform estimate lies in the noise set, then the set theoretic method will not change the estimate. The incorporation of the perturbation  $\Delta\mathbf{S}$  increases the size of the sets  $S_{v'}$ ,  $\Gamma$ , and  $\Gamma_g$ . The increase in size indicates that more points lie in these sets which obviously increases the likelihood of the set containing the uniform estimate or the actual contributions. The residual variance set  $S_v$  ignores perturbations due to  $\Delta\mathbf{S}$ , defining a smaller set less likely to contain the uniform estimate or the actual contributions.

As stated, set theoretic estimates were computed by projecting sequentially onto a

---

<sup>1</sup>In the case of finite mixtures of probability densities, this corresponds to the uniform, or equal, proportion estimate.

chosen finite mixture model set, the sum-to-one set,  $S_\Sigma$ , and the nonnegativity set,  $\{S_k\}_{k=1}^K$ . Two tests for convergence were utilized. When either criteria for convergence is met, the POCS iteration is terminated. The first criterion determines if the estimate lies in all three sets. The second test for convergence is similar to the criterion defined in Eq. (3.52) and is best explained via a simple example. Let  $\hat{\mathbf{a}}_i$  denote the set theoretic estimate at iteration  $i$ . Similarly,  $\hat{\mathbf{a}}_{i+1}$  refers to the estimate after the next iteration. In addition, suppose the finite mixture model set is  $\Gamma_g$ . The POCS iteration is considered to have converged if the following inequality is true

$$\left| \left( \|\bar{\mathbf{S}}\hat{\mathbf{a}}_i - \mathbf{r}\|_2^2 - \rho\|\hat{\mathbf{a}}_i\|_2^2 \right) - \left( \|\bar{\mathbf{S}}\hat{\mathbf{a}}_{i+1} - \mathbf{r}\|_2^2 - \rho\|\hat{\mathbf{a}}_{i+1}\|_2^2 \right) \right| < \epsilon, \quad (5.1)$$

for  $\epsilon = 1 \times 10^{-10}$ . Similar convergence tests were implemented for the other finite mixture model sets, and all convergence tests utilized the same value for  $\epsilon$ .

The chosen finite mixture model set will describe the POCS method implemented. The static POCS (SPOCS) method indicates that the residual variance set  $S_v$  was selected. Likewise dynamic POCS (DPOCS), error-in-variables (EVPOCS), and generalized error-in-variables (GEVPOCS) will refer to the use of the respective sets  $S_{v'}$ ,  $\Gamma$ , and  $\Gamma_g$ . Furthermore, SPOCS (u) indicates that the uniform contribution estimate was chosen to start the SPOCS iteration. The choice of the LS solution and TLS solution as the initial estimate is represented by (ls) and (tls), respectively.

Relative mean-squared estimation error (MSEE) statistics were obtained from simulations to compare the estimates obtained from the various methods. The relative MSEE is given in decibels (dB) by

$$e_{mmse}(\mathbf{a}, \hat{\mathbf{a}}) = 10 \log_{10} \left( \frac{E\{\|\mathbf{a} - \hat{\mathbf{a}}\|^2\}}{\|\mathbf{a}\|^2} \right), \quad (5.2)$$

where  $\mathbf{a}$  and  $\hat{\mathbf{a}}$  denote the actual and estimated class proportions, respectively. Absolute estimation error statistics were computed to evaluate the effectiveness of the estimates with regard to a particular class. The absolute estimation error for class  $k$  is given in dB by

$$e_k([\mathbf{a}]_k, [\hat{\mathbf{a}}]_k) = 10 \log_{10} (E\{|[\mathbf{a}]_k - [\hat{\mathbf{a}}]_k|\}), \quad (5.3)$$

where  $[\mathbf{a}]_k$  and  $[\hat{\mathbf{a}}]_k$  denote the  $k^{th}$  class of the actual and estimate proportions, respectively.

## 5.1 Simulations of Finite Mixtures of Probability Densities

Finite mixtures of probability densities were formed with either parametric or nonparametric class densities. Mixtures of nonparametric class densities may include parametric class densities, but at least one of the classes must have a nonparametric density. The simulations of the two types of mixture densities will be discussed separately, beginning with mixtures described by parametric class densities. First, a few remarks regarding the discrete estimates of the class densities are appropriate.

Discrete estimates of the class density for class  $k$  are obtained from  $H_k S_{\Omega_k}$  samples. Recall,  $H_k$  is the number of sample histograms, and  $S_{\Omega_k}$  is the total number of samples (scalars) in each histogram. Estimates of the autocovariance are obtained from the  $H_k$  discrete samples of the class densities. To do this, the number of samples,  $S_{\Omega_k}$ , is fixed for all  $k$  and approximates the number of samples from each class in an observed sample mixture. This will keep the behavior of the system under analysis somewhat consistent. In other words, a sample histogram from class  $k$  will contain  $S_{\Omega_k} = N$  samples. A histogram representing a mixture of the  $K$  classes will contain  $KN$  samples. Under this formulation, each class is expected to contribute  $N$  samples to the mixture, since no additional knowledge about the mixtures is available. Of course, in an actual mixture, there is variability associated with the proportion contributed by each class. To illustrate this, suppose there are two classes, i.e.  $K = 2$ . An observed mixture would contain  $2N$  samples, but the distribution of the samples depends on the proportion of each class present. In a mixture of these two classes, suppose 70% of the samples belong to class 1. The error of the normalized histogram for class 1 with respect to the discrete estimate of the class density for class 1 would be smaller than the same error computed for class 2. This is expected, since the histogram for class 2 is generated from fewer samples than the histogram for class 1.

The number of samples,  $N$ , also affects the resolution of the proportions,  $\mathbf{a}$ , and the perturbation attributed to  $\Delta\mathbf{S}$ . If  $N$  is small, then the true proportions will exhibit a “quantized” resolution. This “quantization” behavior could be utilized to improve the estimate of the proportions if  $N$  is known beforehand. In some applications, only the (normalized) histograms are observed, and the number of samples,  $N$ , is unavailable. For  $N$  samples, the resolution on the proportions,  $\mathbf{a}$ , is  $\frac{1}{N}$ . Increasing  $N$  will increase the resolution of the proportions, while simultaneously diminishes the variability attributed to  $\Delta\mathbf{S}$ . At some point, it will suffice to use  $\bar{\mathbf{S}}$  for  $\mathbf{S}$  and ignore the errors  $\Delta\mathbf{S}$ . Preferably, estimates of  $\bar{\mathbf{S}}$

should be based on a very large number of samples (scalars) and/or histograms. Although, in practice many samples may not be available.

Given the histogram approximations  $\bar{\mathbf{s}}_k$  of the probability densities for each class according the bin centers  $\mathbf{b}_c$  described in Section 2.4.2, consider a mixture of new samples from the various classes. Define a normalized histogram of the set of  $N$  new samples  $\Omega_{new} = \bigcup_{k=1}^K \Omega_k$  by the  $M \times 1$  vector  $\mathbf{r}$  with respect to the bin centers  $\mathbf{b}_c$ . To create a mixture of the  $K$  classes, form a random vector  $\boldsymbol{\beta}$  whose components are distributed on the interval  $[0, 1]$ . Normalize this vector according to the sum of the elements in  $\boldsymbol{\beta}$  to define the mixture contributions  $\hat{\boldsymbol{\beta}}$  as

$$\hat{\boldsymbol{\beta}} = \frac{\boldsymbol{\beta}}{\sum_{k=1}^K [\boldsymbol{\beta}]_k}. \quad (5.4)$$

Since the true mixture proportions are given by the number of samples from each class, first define the number samples from each class, denoted  $N_k$ , according to  $\hat{\boldsymbol{\beta}}$ . The number of samples from a class must be an integer. Thus,  $N_k$  is given by

$$N_k = \begin{cases} \text{round} \{N[\hat{\boldsymbol{\beta}}]_k\} & 1 \leq k < K \\ N - \sum_{i=1}^{K-1} \text{round} \{N[\hat{\boldsymbol{\beta}}]_i\} & k = K \end{cases}, \quad (5.5)$$

where the function  $\text{round} \{\alpha\}$  rounds the scalar  $\alpha$  to the nearest integer. Let the set  $\mathcal{X}$  denote the collection of  $N$  samples from the  $K$  classes. In the set  $\mathcal{X}$ , there are  $N_k$  samples from the  $k^{\text{th}}$  class. The true mixture contributions given  $N$  samples with  $N_k$  samples from the  $k^{\text{th}}$  class are given by the vector  $\mathbf{a}$  whose  $k^{\text{th}}$  component is given by

$$[\mathbf{a}]_k = \frac{N_k}{N}. \quad (5.6)$$

Thus, the  $K \times 1$  vector  $\mathbf{a}$  defines the true contribution of each class to the mixture. Observe that  $a_k$  has a resolution given by  $\frac{1}{N}$ .

A histogram with  $M$  bins is created from the  $N$  samples in the set  $\mathcal{X}$ . Let the bin centers of the histogram be given by the  $M \times 1$  vector  $\mathbf{b}_c$  defined in Eq. (2.48) with bin widths  $\mathbf{b}_w$  defined in Eq. (2.49). Let the  $M$ -bin histogram formed from the samples in  $\mathcal{X}$  be denoted by the  $M \times 1$  vector  $\hat{\mathbf{r}}$ . The observed mixture  $\mathbf{r}$  is given by

$$\mathbf{r} = \frac{\hat{\mathbf{r}}}{\hat{\mathbf{r}}^T \mathbf{b}_w}. \quad (5.7)$$

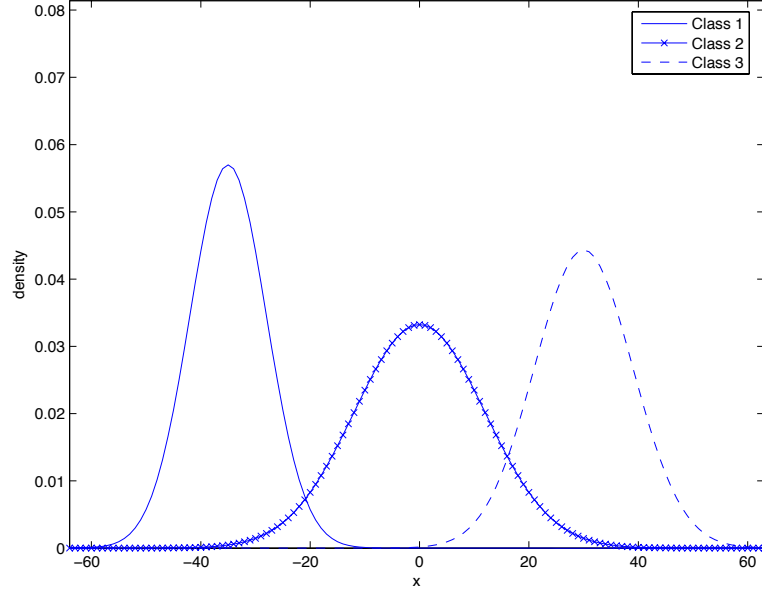


Figure 5.1: Actual class probability densities used for simulations of parametric mixture densities.

### 5.1.1 Finite Mixtures of Parametric Densities

In this simulation, mixture densities with parametric class densities were composed of  $K = 3$  Gaussian densities with unique means and standard deviations. Let the function  $f_{\mathcal{N}}(x; \mu, \sigma)$  denote a Gaussian probability density with mean  $\mu$  and standard deviation  $\sigma$ . The probability density function  $f_{\mathcal{N}}(x; \mu_k, \sigma_k)$  of the  $k^{\text{th}}$  class is given by

$$f_{\mathcal{N}}(x; \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}} \quad \forall x \in \mathbb{R}. \quad (5.8)$$

Thus, the  $K = 3$  class finite mixture probability density defined in Eq. (1.5) is defined as

$$f(x; \boldsymbol{\theta}) = \sum_{k=1}^3 a_k \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}} \quad \forall x \in \mathbb{R}, \quad (5.9)$$

where  $\boldsymbol{\theta} = [a_1, a_2, a_3, \mu_1, \sigma_1, \mu_2, \sigma_2, \mu_3, \sigma_3]^T$ .

For simulation, the parameters of the class densities were defined as  $\boldsymbol{\theta} = [a_1, a_2, a_3, -35, 7, 0, 12, 30, 9]^T$ . The mixture proportions  $\mathbf{a} = [a_1, a_2, a_3]$  were allowed to vary randomly while satisfying the sum-to-one and nonnegativity constraints. The actual class probability densities are shown in Figure 5.1.

Discrete estimates of the class probability densities, given by the columns of the  $M \times K$  matrix  $\bar{\mathbf{S}}$ , were formed in accordance with the procedure outlined in Section 2.4.2.

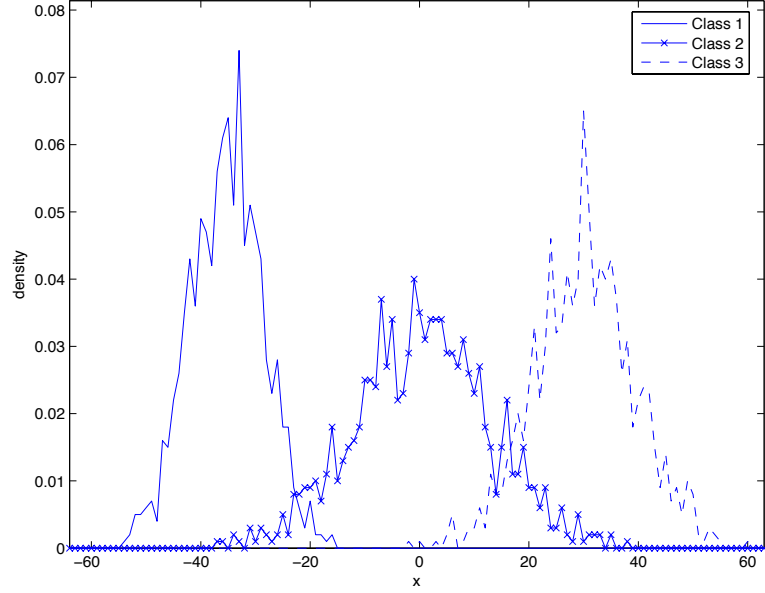


Figure 5.2: Discrete estimates of class probability densities for the simulations of parametric mixture densities. The estimates were formed from  $H_k = 10$  sample histograms, where  $N = 100$  samples were used to form each histogram.

For the  $k^{\text{th}}$  class, sample histograms with  $M = 128$  bins were created with  $N = 100$  samples from a Gaussian distribution with mean  $\mu_k$  and standard deviation  $\sigma_k$ . The bins of the histogram were equally distributed such that the first bin center is located at  $x = -64$  and the last bin center is located at  $x = 63$ . The estimate of each class probability density was computed from the arithmetic average of  $H_k = 10$  sample histograms for  $k = 1, \dots, K$ . The discrete estimates of the class probability densities are shown in Figure 5.2. Estimates of the autocovariance of the discrete class probability densities were determined from the sample histograms for each class. For class  $k$ , the sample autocovariance,  $\mathbf{C}_k$ , approximates the actual autocovariance of the  $k^{\text{th}}$  discrete class probability density. The sample autocovariance is given by

$$\mathbf{C}_k = \frac{1}{H_k - 1} \sum_{h=1}^{H_k} \mathbf{s}_{k,h} \mathbf{s}_{k,h}^T, \quad (5.10)$$

where  $\mathbf{s}_{k,h}$  denotes the  $h^{\text{th}}$  sample histogram from class  $k$ . As the number of samples,  $S_{\Omega_k}$ , or histograms,  $H_k$ , increases, the sample autocovariance converges to the actual autocovariance. The sample autocovariance is used to approximate  $E\|\Delta\mathbf{S}\|_2^2$ .

Given an observed discrete mixture density, estimates of the proportions of the  $K$

components contributing to the mixture were found using LS, SPOCS, DPOCS, GEVPOCS, the EM algorithm, and the modified EM algorithm. The use of LS is justified if the perturbation  $\Delta\mathbf{S}$  is represented as the signal dependent noise source  $\Delta\mathbf{S}\mathbf{a}$ . As mentioned, the parameters and initial estimates for the set theoretic methods varied.

The SPOCS scheme considers only the presence of signal independent noise. The observation is a histogram, normalized to sum-to-one with respect to the bin widths. The histogram is formed from a collection of  $N$  samples. It is assumed that there is no error in the count for each bin; hence, the signal independent noise theoretically is zero. Accordingly, the parameter  $\delta_\eta$  was set to zero. Therefore, the SPOCS estimate merely minimizes the residual while satisfying the sum-to-one and nonnegativity constraints. Two different initial estimates were considered for the SPOCS scheme: the uniform proportion estimate and the LS solution.

The DPOCS scheme combines the signal dependent and signal independent noise sources into the parameter  $\delta'_v = \delta_\eta + \delta_{\mathbf{a}}\rho$ . Again, the parameter  $\delta_\eta$  was set to zero. The parameter  $\delta_{\mathbf{a}}$  approximates  $\|\mathbf{a}\|_2^2$ , and two different approximations were considered. For the finite mixture model problem,  $\|\mathbf{a}\|_2^2$  must not exceed one. The first approximation considers this upper bound on  $\|\mathbf{a}\|_2^2$  and sets the parameter to the maximum value of  $\|\mathbf{a}\|_2^2$ . Note that if the Euclidean norm of the actual proportions is one, then only one class is present in the observed histogram. The problem of interest is mixtures of classes, and it is expected that this choice of  $\delta_{\mathbf{a}}$  will typically be too large. As a result, the parameter  $\delta_{\mathbf{a}}$  was also tested for the approximation given in Eq. (4.7) proposed by [43]. Thus, the label DPOCS (u) indicates that the maximum value of  $\|\mathbf{a}\|_2^2$  is used for  $\delta_{\mathbf{a}}$ , and the label DPOCS (u)\* will indicate that  $\delta_{\mathbf{a}}$  is given by Eq. (4.7). For 1000 simulations, the average value of  $\delta_{\mathbf{a}}$  using the approximation in Eq. (4.7) is 0.4292 and the average actual  $\|\mathbf{a}\|_2^2$  is 0.4091. This suggests that the value defined by Eq. (4.7) is adequate.

To find the parameter  $\rho$  that approximates  $E\|\Delta\mathbf{S}\|_2^2$ , consider either of the following two approaches. From the discussion in Section 4.1.1, the parameter  $\rho$  is directly computed from the  $K$  sample autocovariance matrices according to Eq. (4.5). Alternatively, it was noted in Section 2.4.1 that the  $m^{th}$  bin of the normalized histogram for class  $k$  may be modeled as a binomial random variable with probability  $P_{k,m}$ . Since the histograms have been normalized to sum-to-one and the bin width is one, the variance for the  $m^{th}$  bin

for class  $k$  is given by

$$\text{Var}([\mathbf{s}_k]_m) = \frac{P_{k,m}(1 - P_{k,m})}{N}, \quad (5.11)$$

where  $N$  is the number of samples used to form the normalized histogram,  $\mathbf{s}_k$ . The matrix  $\bar{\mathbf{S}}$  provides discrete approximations of the probabilities  $P_{k,m}$ , so the variance may be approximated by

$$\text{Var}([\mathbf{s}_k]_m) \approx \frac{[\bar{\mathbf{s}}_k]_m(1 - [\bar{\mathbf{s}}_k]_m)}{N} = \frac{1}{N} \left( [\bar{\mathbf{s}}_k]_m - [\bar{\mathbf{s}}_k]_m^2 \right), \quad (5.12)$$

where  $\bar{\mathbf{s}}_k$  is the  $k^{\text{th}}$  column of  $\bar{\mathbf{S}}$ . According to the right hand side of Eq. (4.4), the parameter  $\rho$  is given by

$$\rho = \max_k \left\{ \frac{1}{N} \left( \sum_{m=1}^M [\bar{\mathbf{s}}_k]_m - [\bar{\mathbf{s}}_k]_m^2 \right) \right\} = \max_k \left\{ \frac{1}{N} \left( 1 - \sum_{m=1}^M [\bar{\mathbf{s}}_k]_m^2 \right) \right\}. \quad (5.13)$$

Recall that the  $K$  histograms in  $\bar{\mathbf{S}}$  have been normalized such that  $\sum_{m=1}^M [\bar{\mathbf{s}}_k]_m = 1$ , since the bin width of histograms are uniform and equal one. Either method to approximate  $\rho$  is appropriate. The DPOCS schemes were tested using both the uniform proportion estimate and the LS solution as the initial estimate. The LS solution is compatible with the DPOCS method. Since  $\delta_\eta = 0$ , the residual is only attributed to the signal dependent perturbations. The parameter  $\rho$  was computed directly from the sample autocovariance matrices according to Eq. (4.5) and was set to 0.0098.

The GEVPOCS method assumes both signal independent and signal dependent noise; however, the set does not fix the value of  $\|\mathbf{a}\|_2^2$ . The value  $\|\mathbf{a}\|_2^2$  is allowed to vary, changing the size of the set  $\Gamma_g$ . The parameters  $\delta_\eta$  and  $\rho$  chosen for the DPOCS scheme were used. The parameter  $\rho$  must be less than the square of the smallest singular value of  $\bar{\mathbf{S}}$  to ensure that  $\Gamma_g$  is a convex set. The singular values of  $\bar{\mathbf{S}}$  are 0.2080, 0.1841, and 0.1484. The square of the smallest singular value is 0.0220, and this value is clearly greater than  $\rho = 0.0098$ . The GEVPOCS iteration was tested using both the uniform proportion estimate and the LS solution as the initial estimate. As for the DPOCS scheme, the LS solution is compatible with the GEVPOCS method since  $\delta_\eta = 0$ .

In contrast to the other methods, the EM algorithm uses the  $KN$  samples that form the observed mixture. Furthermore, the EM algorithm estimates both the proportions and the parameters of the class probability densities. The uniform proportion estimate was used to initialize the proportions for the EM algorithm. The initial estimate of the class means were generated from a Gaussian distribution whose mean is given by the

Table 5.1: Estimates of Class Proportions for Sample Mixture of Gaussian Probability Densities in Figure 5.3

Method	Class 1	Class 2	Class 3	Relative MSE (dB)	Iterations
Actual	0.5300	0.1300	0.3400	n/a	n/a
LS	0.5307	0.0983	0.3579	-24.9363	n/a
SPOCS (u)	0.5351	0.1026	0.3622	-25.1292	2
SPOCS (ls)	0.5351	0.1026	0.3622	-25.1292	2
DPOCS (u)	0.3333	0.3333	0.3333	-7.1292	0
DPOCS (ls)	0.5351	0.1026	0.3622	-25.1292	1
DPOCS (u)*	0.4489	0.2187	0.3324	-14.5486	1
DPOCS (ls)*	0.5351	0.1026	0.3622	-25.1292	1
GEVPOCS (u)	0.5431	0.1073	0.3496	-27.2397	56
GEVPOCS (ls)	0.5371	0.1003	0.3626	-24.5822	1
EM Alg.	0.5310	0.1945	0.2745	-16.8944	17
EM Alg. (mod.)	0.5379	0.1172	0.3449	-32.1983	5
Uniform	0.3333	0.3333	0.3333	-7.1292	n/a

sample mean of the  $KN$  observed samples and whose variance is the sample variance of the  $KN$  samples. The initial estimate of the component variances were set to the sample variance of the  $KN$  samples. Simulations revealed that the EM algorithm performed very poorly when few samples form the observed mixture. The implementation of the modified EM algorithm provides the EM algorithm with greater knowledge about the underlying class probability densities. For the modified implementation of the EM algorithm, estimates of the mean and standard deviation for each class were fixed in the algorithm. These estimates were determined from the  $H_k N$  samples (scalars) used to construct  $\bar{\mathbf{S}}$ . The estimates of the mean and standard deviation of the class probability densities were fixed, and thus, the modified EM algorithm estimated only the proportions of the classes. The estimates of the parameters of the class probability densities for the modified EM algorithm were  $[\hat{\mu}_1, \hat{\sigma}_1, \hat{\mu}_2, \hat{\sigma}_2, \hat{\mu}_3, \hat{\sigma}_3] = [-35.3016, 6.6044, 0.5465, 12.3751, 30.1595, 9.2230]$ . Both implementations of the EM algorithm were empirically determined to have converged when the difference  $\log L(\boldsymbol{\theta}^{i+1}) - \log L(\boldsymbol{\theta}^i)$  is less than  $2 \times 10^{-3}$ .

A sample discrete mixture density obtained from  $KN = 300$  samples is shown in Figure 5.3. The estimates of the class proportions of this mixture are provided in Table 5.1. The results for this sample mixture illustrate the behavior of the various estimation methods.

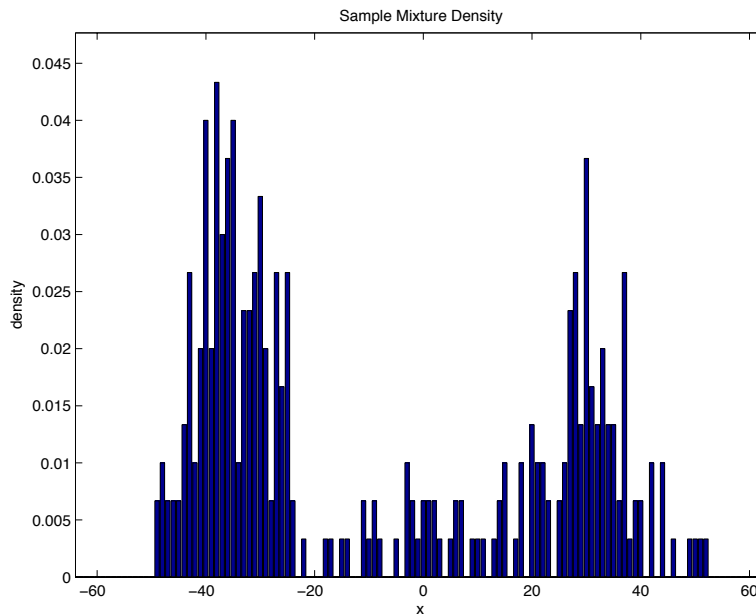


Figure 5.3: Sample discrete mixture density obtained from  $KN = 300$  samples. Refer to Table 5.1 for estimates of the class proportions.

Of all the methods, the modified EM algorithm performed the best. This is expected, for the algorithm essentially fits a  $K = 3$  class Gaussian mixture density to  $KN = 300$  samples from the mixture. Furthermore, only the class proportions were estimate, since the mean and standard deviation of each class were approximated well by the  $S_{\Omega_k} H_k = 1000$  class samples (scalars). The EM algorithm, which also estimates the class probability density parameters, shows a much poorer relative mean square error (MSE) for the class proportions. The estimates of the class probability density parameters are given in Table 5.2. The variable  $c$  in Table 5.2 is used to emphasize that the EM algorithm does not necessarily produce class probability densities with parameters corresponding to the classes defined by Eq. (5.9). Notice that the first ( $k = 1$ ) class probability density parameters were significantly more accurate than the parameters for the other classes. Furthermore, note that the proportion of this class is very accurately estimated by the EM algorithm, even better than the modified EM algorithm. This relationship is intuitive. Since the first class contributes the most samples to the mixture, the parameters corresponding to the first class should be more accurate. In fact, most of the methods accurately estimate the proportion of the first class.

A pitfall of the EM algorithm is that it does not constrain the parameters of the

class probability densities. Those parameters are allowed to vary freely in order to fit a  $K$  class Gaussian mixture density to the  $KN$  observed samples. This can lead to parameters of the Gaussian components that differ from the actual components when the number of samples is small. Consequently, the proportions found via the EM algorithm may be associated with the Gaussian density components with parameters different from the actual parameters. Proportion errors have a tendency to increase in these circumstances, since the estimated proportions deviate from the actual proportions and correspond with densities having parameters that best fit the samples. Table 5.2 includes a column indicating the association of the EM algorithm components with the actual classes based on the proportion estimate. These associations are reasonable given the estimates of the Gaussian density components. However, the comparatively poor relative MSE found for the EM algorithm estimate is justified, since the proportions are based on a finite mixture model with different component density parameters.

Figure 5.4 compares the mixtures based on the parameters from the EM algorithm and the modified EM algorithm to the normalized histogram of the observed samples. Both algorithms produce plausible mixtures, and it is noted that the mixture using the EM algorithm estimates appear to provide a better fit. Recall that the EM algorithm is designed to find parameters of the finite mixture model to fit the observed. Notice that the EM algorithm defined a class probability density with mean 8.4301 and standard deviation 18.6402. Table 5.2 shows that these parameters are associated with class  $k = 2$ , but class  $k = 2$  actually has mean zero and standard deviation 12. This discrepancy weakens the association of the classes described by the EM algorithm estimate. Proportions found via the modified EM algorithm are much more appropriate for comparisons, since the proportions are associated with class probability densities whose parameters are very close to the actual values.

The EM algorithm performs poorly and converges very slowly when compared to the number of iterations for the other methods. Clearly, the limited number of observed samples (scalars) impairs the ability of the EM algorithm to accurately estimate the proportions and the components.

The remaining methods estimate only the proportions from the sample discrete mixture density of the  $KN = 300$  samples. The LS solution is quite accurate; however, the LS solution is not guaranteed the sum-to-one and nonnegativity constraints. For this sample mixture, the sum of the LS solution is 0.9869. Again, the proportion of the first

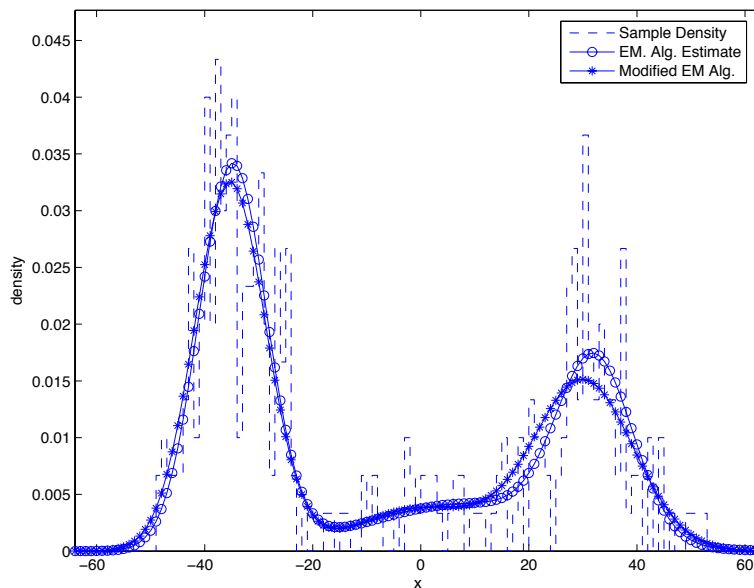


Figure 5.4: Comparison of observed mixture  $\mathbf{r}$  from Figure 5.3 and the reconstructed mixture based on the standard and modified EM algorithm estimates in Table 5.1. The EM algorithm estimates were substituted into Eq. (1.5) to produce reconstructed mixture density.

Table 5.2: Results for EM Algorithm for Gaussian Mixture Density in Figure 5.3

EM Alg. Component $c$	Est. $\hat{\alpha}_c$	Est. $\mu_c$	Est. $\sigma_c$	Associated Class, $k$
1	0.2745	32.1965	7.0234	3
2	0.5310	-34.8192	6.2458	1
3	0.1945	8.4301	18.6402	2

class is the most accurate.

The set theoretic estimates guarantee that a solution will satisfy the finite mixture model set as well as the sum-to-one and nonnegativity constraints. The SPOCS estimate is not affected by the initial estimates tested. The SPOCS implementations were observed to have converged when deviation of the Euclidean norm of the residual  $\|\mathbf{r} - \bar{\mathbf{S}}\hat{\mathbf{a}}\|_2$  fell below the threshold  $1 \times 10^{-10}$  for two consecutive estimates. This criterion is analogous to the example described by Eq. (5.1). This condition forces convergence without necessarily finding a feasible solution of the estimate. The convergence of the SPOCS scheme in two iterations according to Table 5.3 indicates that the estimate  $\hat{\mathbf{a}}$  does not lie in the set  $S_v$ . Recall that the parameter  $\delta_\eta$  is set to zero. In order for  $\mathbf{a}$  to lie in  $S_v$ , it is necessary that  $\mathbf{r} = \bar{\mathbf{S}}\mathbf{a}$ . This is very unlikely, since  $\bar{\mathbf{S}}$  is estimated from samples. Thus, the SPOCS estimates minimize the norm of the residual while satisfying the sum-to-one and nonnegativity sets. Furthermore, this result, regardless of the initial estimate used, indicates that the set  $S_v$  projects to the LS solution when  $\delta_\eta = 0$ , even though the LS solution does not lie in  $S_v$  when  $\delta_\eta = 0$ .

The DPOCS estimate is sensitive to both the initial estimate and the choice of  $\delta_{\mathbf{a}}$ . When the uniform proportion estimate is used as the initial estimate and  $\delta_{\mathbf{a}} = 1$ , the DPOCS estimate remains unchanged. Since the uniform proportion estimate satisfies the sum-to-one and nonnegativity constraints, zero iterations for DPOCS (u) indicates that the uniform proportion estimate lies in the set  $S_{v'}$ . Using the alternative definition for  $\delta_{\mathbf{a}}$  given by Eq. (4.7), the DPOCS scheme estimate significantly improves. Decreasing the value of the parameter  $\delta'_{v'}$  reduces the size of the set  $S_{v'}$ , clearly eliminating the uniform proportion estimate as a feasible solution. The DPOCS estimate is observed to be identical to the SPOCS estimates when the LS solution is the initial estimate. However, the DPOCS scheme converges in one iteration rather than two. This indicates that the projection<sup>2</sup> of the LS solution to the sum-to-one set,  $S_\Sigma$ , lies in the set  $S_{v'}$ . In contrast to the SPOCS method, the DPOCS scheme provides an estimate contained in the intersection of the sets defining a feasible estimate.

As an illustration, Figure 5.5 compares the reconstructed mixture based on the DPOCS (ls)\* estimate to the observed mixture  $\mathbf{r}$ . The reconstructed mixture is visually observed to fit well to the observed mixture  $\mathbf{r}$ .

GEVPOCS is also affected by the initial estimate. Among the set theoretic

---

<sup>2</sup>Projections to the nonnegativity sets were unnecessary for this specific mixture density.

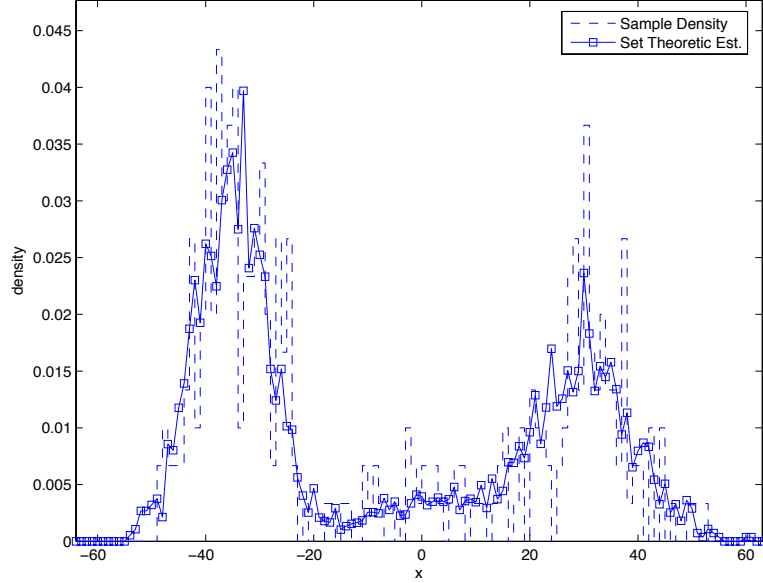


Figure 5.5: Comparison of observed mixture  $\mathbf{r}$  from Figure 5.3 and the reconstructed mixture based on the DPOCS (ls)\* estimate in Table 5.1.  $\bar{\mathbf{S}}$  is used reconstruct the mixture for the DPOCS (ls)\* estimate.

schemes, the GEVPOCS estimate using the uniform contribution estimate as the initial estimate performs the best. However, the improved accuracy of the estimate required many more iterations to converge. It is important to recall that the POCS theorem guarantees convergence but not necessarily in a finite number of iterations. When the LS solution is the initial estimate, the GEVPOCS scheme converges in one iteration. Note that the converged estimate is not identical to the SPOCS and DPOCS methods using the same initial estimate. Thus, the GEVPOCS scheme does project to the set  $\Gamma_g$ .

The underlying construction of the various methods differ and, consequently, the interpretation of the number of iterations to achieve convergence as listed in Table 5.1 differ. To give meaning to the number of iterations, the time elapsed to compute the estimate of the class proportions is listed in Table 5.3. The simulations were performed in MATLAB 7 on an Apple PowerBook 1.5GHz G4 processor with 512MB of RAM. For each method, the time elapsed is determined from the average time of 1000 identical simulations of the estimate of the proportions listed in Table 5.1. The noise source is kept constant by seeding the random number generator with the same value. The redundant simulations provide a more accurate estimate of the elapsed time, since several unexpected background processes affect the allocation of the processor to MATLAB. Table 5.3 illustrates that most of the estimates

Table 5.3: Comparison of the Number of Iterations and the Average Elapsed Time for the Estimation Methods for Sample Mixture of Gaussian Probability Densities in Figure 5.3

Method	Relative MSE (dB)	Iterations	Avg. Elapsed Time (sec)
LS	-24.9363	n/a	0.00021
SPOCS (u)	-25.1292	2	0.13106
SPOCS (ls)	-25.1292	2	0.08491
DPOCS (u)	-7.1292	0	0.01132
DPOCS (ls)	-25.1292	1	0.01251
DPOCS (u)*	-14.5486	1	0.03494
DPOCS (ls)*	-25.1292	1	0.01275
GEVPOCS (u)	-27.2397	56	1.60652
GEVPOCS (ls)	-24.5822	1	0.03357
EM Alg.	-16.8944	17	0.28219
EM Alg. (mod.)	-32.1983	5	0.08129

obtained via the POCS methods are computed much faster than the EM Algorithm.

To obtain mean-squared estimation error (MSEE) statistics for the various estimation methods, 1000 Monte Carlo simulations were performed. In addition, these simulations verify the results observed for the sample mixture density considered before. The EM algorithm is not included in this simulation per the conclusions from the sample mixture density. Each discrete mixture density is generated according to the same procedure used to create the sample discrete mixture density discussed before.

The estimated relative MSEE and average number of iterations for the methods implemented to estimate the proportions from the simulated mixtures are provided in Table 5.4. Also, the number of converged iterations for the set theoretic methods are included in the table. The modified EM algorithm converges based on the deviation in the log-likelihood function after successive iterations. Thus, an upper bound on the number of iterations was not imposed. For the set theoretic methods, the estimated relative MSEE and average number of iterations are computed from only the converged estimates. If a method exceeded 100 iterations, then it was said to have not converged.

As observed for the sample mixture density, the modified EM algorithm performs the best among all the methods tested. The remaining methods estimate the proportions from the discrete mixture densities of the  $KN = 300$  samples. While the LS solution is quite accurate, the sum-to-one and nonnegativity constraints are not guaranteed.

The SPOCS estimate is nearly unaffected by the initial estimate. The difference in

Table 5.4: Results for Finite Mixtures with Gaussian Probability Densities from 1000 Simulations

Method	Est. Relative MSE (dB)	Avg. Iter.	Converged Iter.
LS	-24.0865	n/a	n/a
SPOCS (u)	-25.8341	2.0000	100%
SPOCS (ls)	-25.8341	1.9990	100%
DPOCS (u)	-7.4017	0.0680	100%
DPOCS (ls)	-25.8358	1.0230	100%
DPOCS (u)*	-14.0742	1.0410	100%
DPOCS (ls)*	-25.8355	1.1840	100%
GEVPOCS (u)	-15.1976	29.5268	83.9%
GEVPOCS (ls)	-21.5902	10.4579	87.8%
EM Alg. (mod.)	-28.7373	4.6980	n/a
Uniform	-7.1055	n/a	n/a

the average number of iterations when the LS solution serves as the initial estimate is due to one of the LS solutions satisfying the sum-to-one and nonnegativity constraints. Thus, no projections were necessary; the LS solution was a feasible solution in that particular case.

The DPOCS scheme exhibits the same behavior noted for the sample discrete mixture density. From Table 5.4, it is apparent that the accuracy of the GEVPOCS (u) estimate for the sample mixture is unusual. The simulations indicate that the GEVPOCS (u) estimate is slightly more accurate than the DPOCS (u)\* estimate. However, the GEVPOCS (u) scheme requires many more iterations on average and may not even converge in less than 100 iterations. The GEVPOCS (ls) produces less accurate estimates of the parameters than the other set theoretic schemes that also use the LS solution as an initial estimate.

Having established the performance of the set theoretic methods when the class probability densities are parametric, the application of the set theoretic methods is considered for mixture of classes described by nonparametric densities.

### 5.1.2 Finite Mixtures of Nonparametric Densities

For this simulation, the nonparametric mixture densities were composed of  $K = 3$  nonstandard probability densities. Nonstandard probability densities were defined as a mixture of parametric probability densities. Specifically, four parametric probability densities

were used to form the  $K = 3$  classes with nonparametric probability densities. Let the function  $f_e(x; \lambda)$  denote an exponential probability density with mean  $\frac{1}{\lambda}$ . Let  $f_R(x; \omega)$  denote a Rayleigh probability density with mean  $\sqrt{\frac{\pi}{2}}\omega$ . Let the probability density function for the each of the  $k$  classes have the form

$$f(x; \boldsymbol{\xi}_k) = \gamma_1^{(k)} f_{\mathcal{N}}(x; \hat{\mu}_1^{(k)}, \hat{\sigma}_1^{(k)}) + \gamma_2^{(k)} f_{\mathcal{N}}(x; \hat{\mu}_2^{(k)}, \hat{\sigma}_2^{(k)}) + \gamma_3^{(k)} f_e(x; \lambda^{(k)}) + \gamma_4^{(k)} f_R(x; \omega^{(k)}), \quad (5.14)$$

where  $\boldsymbol{\xi}_k = [\gamma_1^{(k)}, \gamma_2^{(k)}, \gamma_3^{(k)}, \gamma_4^{(k)}, \hat{\mu}_1^{(k)}, \hat{\sigma}_1^{(k)}, \hat{\mu}_2^{(k)}, \hat{\sigma}_2^{(k)}, \lambda^{(k)}, \omega^{(k)}]^T$  are fixed parameters for a particular model. Note that, in general,  $\boldsymbol{\xi}_k \neq \boldsymbol{\xi}_j$  for  $k \neq j$ . Thus, the  $K = 3$  class finite mixture probability density defined in Eq. (1.5) is given by

$$f(x; \boldsymbol{\theta}) = \sum_{k=1}^3 a_k f(x; \boldsymbol{\xi}_k), \quad \forall x \in \mathbb{R}, \quad (5.15)$$

where  $\boldsymbol{\theta} = [a_1, a_2, a_3, \boldsymbol{\xi}_1^T, \boldsymbol{\xi}_2^T, \boldsymbol{\xi}_3^T]^T$  and  $f(x; \boldsymbol{\xi}_k)$  is given by Eq. (5.14). The EM algorithm may be constructed to determine all of the parameters, but such a formulation of the algorithm is certainly complex. Furthermore, it would need to be known beforehand that the finite mixture model has this composition. It is assumed that no analysis has been performed to describe the classes in terms of a mixture of parametric densities as given by Eq. (5.14). Thus,  $f(x; \boldsymbol{\theta})$  is a nonparametric density, since its composition, though parametric, is unknown. Defining the classes in this manner provides a convenient approach to simulate the  $K$  classes.

For simulation, the parameters of the probability densities of the  $K = 3$  classes were defined as

$$\boldsymbol{\xi}_1 = [0.5, 0, 0.5, 0, 100, 7, 0, 1, 5, 1]^T, \quad (5.16)$$

$$\boldsymbol{\xi}_2 = [0.5, 0, 0, 0.5, 46, 13, 0, 1, 1, 22]^T, \quad (5.17)$$

$$\boldsymbol{\xi}_3 = [0.6, 0.3, 0.1, 0, 77, 8, 60, 5, 20, 1]^T. \quad (5.18)$$

The true class probability densities are shown in Figure 5.6.

Discrete estimates of the class probability densities, given by the columns of the  $M \times K$  matrix  $\bar{\mathbf{S}}$ , were formed in accordance with the algorithm described in Section 2.4.2. However, the creation of sample histograms from each class is a little more elaborate. For the  $k^{\text{th}}$  class, sample histograms with  $M = 128$  bins were created with  $N = 100$  samples with a probability density described by  $f_k(x; \boldsymbol{\xi}_k)$  in Eq. (1.5). The bins of the histogram were

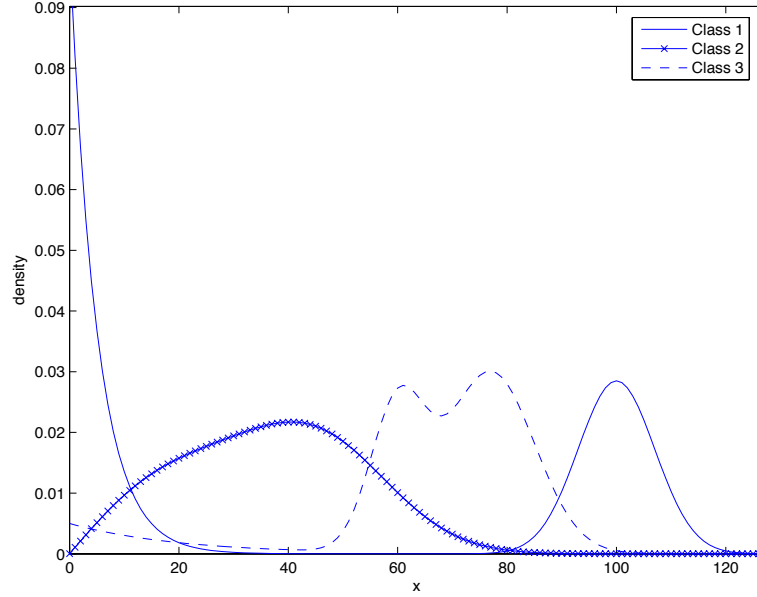


Figure 5.6: Actual class probability densities for the simulations of nonparametric mixture densities.

equally distributed such that the first bin center is located at  $x = 0$  and the last bin center is located at  $x = 127$ . The estimate of each class probability density was computed as the arithmetic average of  $H_k = 10$  sample histograms for  $k = 1, \dots, K$ . The discrete estimates of the class probability densities are shown in Figure 5.7. Estimates of the autocovariance matrix from the discrete class densities were determined from the sample histograms for each class as defined by Eq. (5.10). Let  $\mathbf{C}_k$  denote the sample autocovariance matrix of the  $k^{\text{th}}$  discrete class density. As for the parametric case, the sample autocovariance matrix converges to the actual autocovariance matrix as the number of samples,  $S_{\Omega_k}$ , or histograms,  $H_k$  increases.

Given an observed discrete mixture density, estimates of the proportions of the  $K$  classes contributing to the mixture were found using LS, SPOCS, DPOCS, and GEVPOCS. The EM algorithm cannot be applied to nonparametric models, since the underlying classes are assumed to have no parametric form. All of the set theoretic schemes were tested with the initial estimates used for parametric mixture densities. The parameter  $\delta_\eta$  was set to zero. For the DPOCS and GEVPOCS, the parameter  $\rho$  was determined from the sample autocovariance matrices via the same method described for the parametric case and was set to 0.0110. The singular values of  $\bar{\mathbf{S}}$  are 0.1872, 0.1564, and 0.1175, and the square of

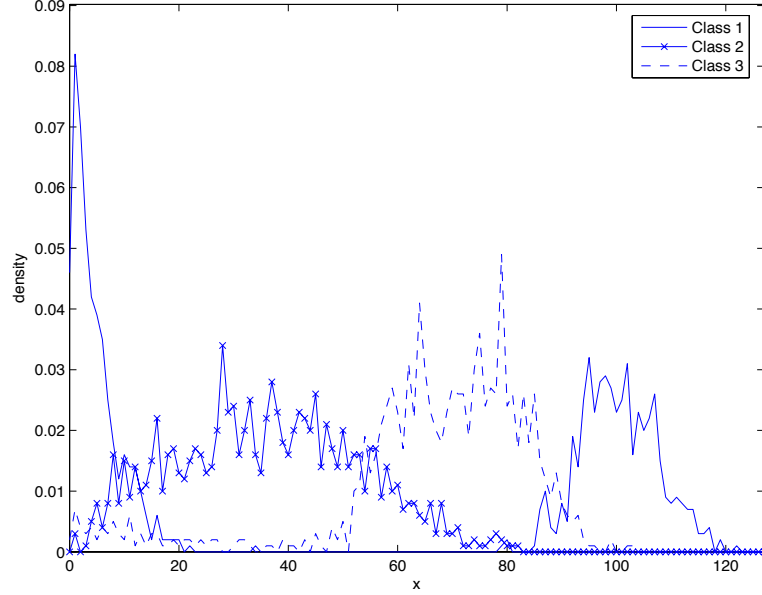


Figure 5.7: Discrete estimates of component probability densities for the simulations of nonparametric mixture densities. The estimates were formed from 10 sample histograms, where  $S_{\Omega_k} = 100$  samples were used to form each histogram.

the smallest singular value is 0.0138, which is greater than  $\rho$ . For the DPOCS method, the parameter  $\delta_{\mathbf{a}}$  was tested for the maximum value of  $\|\mathbf{a}\|_2^2$  and the approximation proposed by [43].

A sample discrete mixture density obtained from  $N = 100$  samples is shown in Figure 5.8. The estimates of the class proportions of this mixture are provided in Table 5.5. Since histograms approximate the class probability densities, LS formulation and the various set theoretic methods do not impose restrictions on the actual form of the class probability densities. The results for this sample discrete mixture density illustrate the flexibility of the proposed methods to estimate the class proportions, for the results resemble the results for parametric finite mixtures.

The LS solution is not as accurate as the set theoretic solutions, and it also does not satisfy the sum-to-one constraint. The sum of the LS solution is 0.9529. Projecting the LS solution to the set  $S_{\Sigma}$  yields the component estimates shown for the SPOCS scheme. As with the example with parametric component densities, the SPOCS scheme projects to  $S_{\Sigma}$  and converges since the change in the Euclidean norm of the residual  $\|\mathbf{r} - \bar{\mathbf{S}}\mathbf{a}\|_2$  does not exceed  $1 \times 10^{-10}$  after two, consecutive iterations. Regardless of the selection of  $\delta_{\mathbf{a}}$ , the

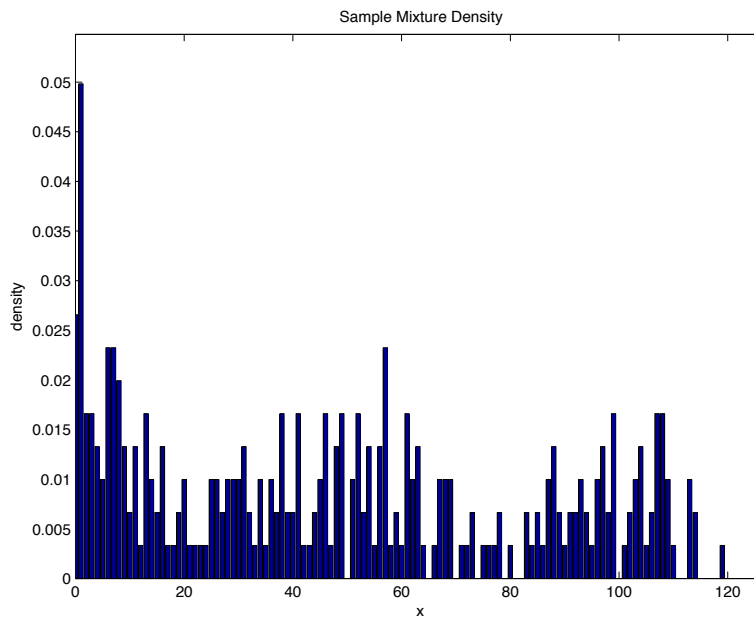


Figure 5.8: Sample discrete mixture density obtained from  $KN = 300$  samples. Refer to Table 5.5 for estimates of class proportions.

Table 5.5: Estimates of Component Proportions for Sample Mixture of Nonparametric Probability Densities in Figure 5.8

Method	Class 1	Class 2	Class 3	Relative MSE (dB)	Iterations
Actual	0.4133	0.4433	0.1433	n/a	n/a
LS	0.4007	0.4089	0.1433	-24.6133	n/a
SPOCS (u)	0.4164	0.4246	0.1589	-28.0846	2
SPOCS (ls)	0.4164	0.4246	0.1589	-28.0846	2
DPOCS (u)	0.3333	0.3333	0.3333	-8.5156	0
DPOCS (ls)	0.4164	0.4246	0.1589	-28.0846	1
DPOCS (u)*	0.3333	0.3333	0.3333	-8.5156	0
DPOCS (ls)*	0.4164	0.4246	0.1589	-28.0846	1
GEVPOCS (u)	0.3928	0.3768	0.2305	-14.9388	55
GEVPOCS (ls)	0.4164	0.4246	0.1589	-28.0846	1
Uniform	0.3333	0.3333	0.3333	-8.5156	n/a

Table 5.6: Comparison of the Number of Iterations and the Average Elapsed Time for the Estimation Methods for Sample Mixture of Nonparametric Probability Densities in Figure 5.8

Method	Relative MSE (dB)	Iterations	Avg. Elapsed Time (sec)
LS	-24.6133	n/a	0.00023
SPOCS (u)	-28.0846	2	0.13109
SPOCS (ls)	-28.0846	2	0.08709
DPOCS (u)	-8.5156	0	0.01128
DPOCS (ls)	-28.0846	1	0.01249
DPOCS (u)*	-8.5156	0	0.01157
DPOCS (ls)*	-28.0846	1	0.01275
GEVPOCS (u)	-14.9388	55	1.49452
GEVPOCS (ls)	-28.0846	1	0.01246

DPOCS (u) methods do not modify the uniform proportion estimate. Thus, the uniform proportion estimate must lie in the set  $S_{v'}$ . The other DPOCS methods converge once the LS solution has been projected to the sum-to-one set,  $S_{\Sigma}$ . Furthermore, the GEVPOCS (ls) estimate follows the same procedure to achieve convergence. These results indicate that the LS solution must lie in the sets  $S_{v'}$  and  $\Gamma_g$ . Finally, it is observed that the GEVPOCS (u) is quite accurate when compared with the DPOCS (u) and DPOCS (u)\* estimates, but 55 iterations were necessary to establish convergence. Furthermore, the GEVPOCS (u) estimate reveals that the uniform proportion estimate is not contained in the set  $\Gamma_g$ .

The averaged elapsed time from 1000 identical simulations of the estimate of the proportions by each method were computed in the same manner described for the sample parametric mixture. The comparison of the number of iterations to the elapsed time for each estimation method is listed in Table 5.6. Notice that the elapsed times for the various methods are nearly the same as those computed for the sample parametric mixture. As expected, these similarities indicate that the underlying component densities have very little impact on the computation time.

Mean-squared estimation error (MSEE) statistics for the various estimation methods were obtained from 1000 Monte Carlo simulations. As before, these simulations verify the results observed for the sample mixture density. The discrete mixture densities are formed according to the procedure used to create the sample discrete mixture density discussed before.

The estimated relative MSEE and average number of iterations for the methods implemented to estimate the proportions from the simulated mixtures are provided in Table 5.7. The table also includes the number of converged iterations for the set theoretic methods. For the set theoretic methods, the estimated relative MSEE and average number of iterations are computed from only the converged estimates.

Table 5.7: Results for Finite Mixtures with Nonparametric Components from 1000 Simulations

Method	Est. Relative MSEE (dB)	Avg. Iter.	Converged Iter.
LS	-22.7938	n/a	n/a
SPOCS (u)	-23.7875	2.0000	100%
SPOCS (ls)	-23.7875	2.0000	100%
DPOCS (u)	-7.2220	0.0130	100%
DPOCS (ls)	-23.7881	1.0110	100%
DPOCS (u)*	-9.2234	0.4330	100%
DPOCS (ls)*	-23.7881	1.0160	100%
GEVPOCS (u)	-12.1259	30.9567	85.4%
GEVPOCS (ls)	-23.0609	2.6098	89.7%
Uniform	-7.1831	n/a	n/a

The results for the nonparametric mixture models are expected to be similar parametric mixture models, since the estimation methods use histogram estimates of the class probability densities. Thus, the mixture model form, either parameter or nonparametric, of the contributing probability densities does not significantly alter the behavior of the set theoretic methods.

### 5.1.3 Discussion

A noted shortcoming of either of the EM algorithm implementations is the prior knowledge about the number of classes. The EM algorithm assumes a fixed number of classes appear in the mixture of samples. For a fixed number of classes, the algorithm finds the maximum likelihood estimate of the parameters of the chosen probability densities given the samples. When one class contributions very little, or not at all, the EM algorithm can begin to produce erroneous estimates of the parameters of the class probability density as well as the class proportion. The set theoretic methods do not require that all the known classes contribution to an observed mixture. This allows accurate estimates of the class

contributing the least, which may be the class of greatest interest. On the other hand, if more than  $K$  classes appear in the observed mixture, the set theoretic methods will produce only estimates of the contributions based on the known  $K$  classes.

The set theoretic methods performed well regardless of the form of the class probability densities. Unsurprisingly, the methods would show less accurate results when the class probability densities overlap. Of course, the EM algorithm would encounter the same problems, since overlapping class densities indicates ambiguous labels for the scalar samples. Such class probability densities do not produce a problem when only the tails of the class probability densities overlap. Distinctions between the classes is established in the regions where the probability is greater. When the regions having higher probability overlap, then any of the methods are expected to generate poor results.

The simulations of finite mixtures of probability densities illustrated the performance of the set theoretic methods. Simulations based on the application to target detection using spectral densities follow.

## 5.2 Simulations of Finite Mixtures of Spectral Densities

Finite mixtures of spectral densities were considered in the context of the hyperspectral target detection application described in Section 1.1. Two types of simulations were designed to evaluate the effectiveness of the set theoretic methods developed to estimate the contributions of the classes in an observed spectral density. The first simulation demonstrates the general performance of the estimation methods for various contributions of a specified class, or target. The second simulation models a simple hyperspectral camera that records an image containing pixels of mixed spectra. The image possesses mixtures of spectral densities resulting from the natural blurring of neighboring objects.

A set of  $K = 4$  spectral classes were constructed from the collection of  $M = 31$  band spectral data of 170 objects in [44]. To create the classes, first the collection of 170 object spectra was clustered into 25 clusters using the k-means clustering algorithm in MATLAB. The measure of similarity for the algorithm was defined by the correlation of samples. For two measurements,  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , the similarity is computed as

$$d(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{(\mathbf{x}_i - \bar{\mathbf{x}}_i)^T (\mathbf{x}_j - \bar{\mathbf{x}}_j)}{\|\mathbf{x}_i - \bar{\mathbf{x}}_i\|_2 \|\mathbf{x}_j - \bar{\mathbf{x}}_j\|_2}, \quad (5.19)$$

where

$$\bar{\mathbf{x}}_i = \left( \frac{1}{M} \sum_{m=1}^M [\mathbf{x}_i]_m \right) \mathbf{1}_{M \times 1} \quad (5.20)$$

$$\bar{\mathbf{x}}_j = \left( \frac{1}{M} \sum_{m=1}^M [\mathbf{x}_j]_m \right) \mathbf{1}_{M \times 1}, \quad (5.21)$$

and  $\mathbf{1}_{M \times 1}$  is an  $M \times 1$  vector of ones. Four clusters were selected from the 25 formed by the MATLAB `kmeans` routine. The samples in the chosen clusters provided a set of basis spectra for the respective classes. Figures 5.9(a)-5.9(d) show the samples from the four clusters created using the k-means clustering algorithm.

For each set, a class sample is formed from a random mixture of the class basis spectra. The random mixture is found by first generating  $B_k$  random numbers from a uniform distribution defined on the interval  $[0, 1]$ , where  $B_k$  is the number of basis spectra for class  $k$ . The  $B_k$  numbers were normalized such that they sum-to-one. Let the  $B_k \times 1$  vector  $\boldsymbol{\alpha}$  denote the normalized  $B_k$  random numbers. A class sample is given by the matrix-vector product

$$\mathbf{s}_k = \tilde{\mathbf{S}}_k \boldsymbol{\alpha}, \quad (5.22)$$

where the  $M \times B_k$  matrix  $\tilde{\mathbf{S}}_k$  is the basis spectra for class  $k$

Estimates of the mean spectral signatures were obtained from the arithmetic mean of  $H_k = 10$  class samples for each set according to Eq. (5.22). The estimates of the mean spectral signatures obtained for the set of spectral signatures are shown in Figure 5.10.

Given an observed spectral density, estimates of the contributions of the  $K$  classes were found using LS, TLS, SPOCS, DPOCS, EVPOCS and GEVPOCS. The LS solution combines the signal dependent and signal independent noise sources as  $\Delta \mathbf{S} \mathbf{a} + \boldsymbol{\eta}$ . The parameters and initial estimates for the set theoretic formulations vary, and the variations are discussed below. The accuracy of the estimates were compared to the uniform contribution estimate for the observed mixture.

The SPOCS scheme overlooks the perturbations  $\Delta \mathbf{S}$  and only considers the signal independent noise  $\boldsymbol{\eta}$  in the model. In this application, the signal independent noise accounts for the system measurement error. The signal independent noise is assumed to be known to achieve a signal-to-noise (SNR) of  $SNR_{dB}$  in decibels (dB). The SNR in dB is defined as

$$SNR_{dB} = 10 \log_{10} \left( \frac{\|\mathbf{S} \mathbf{a}\|_2^2}{M \sigma_\eta^2} \right), \quad (5.23)$$

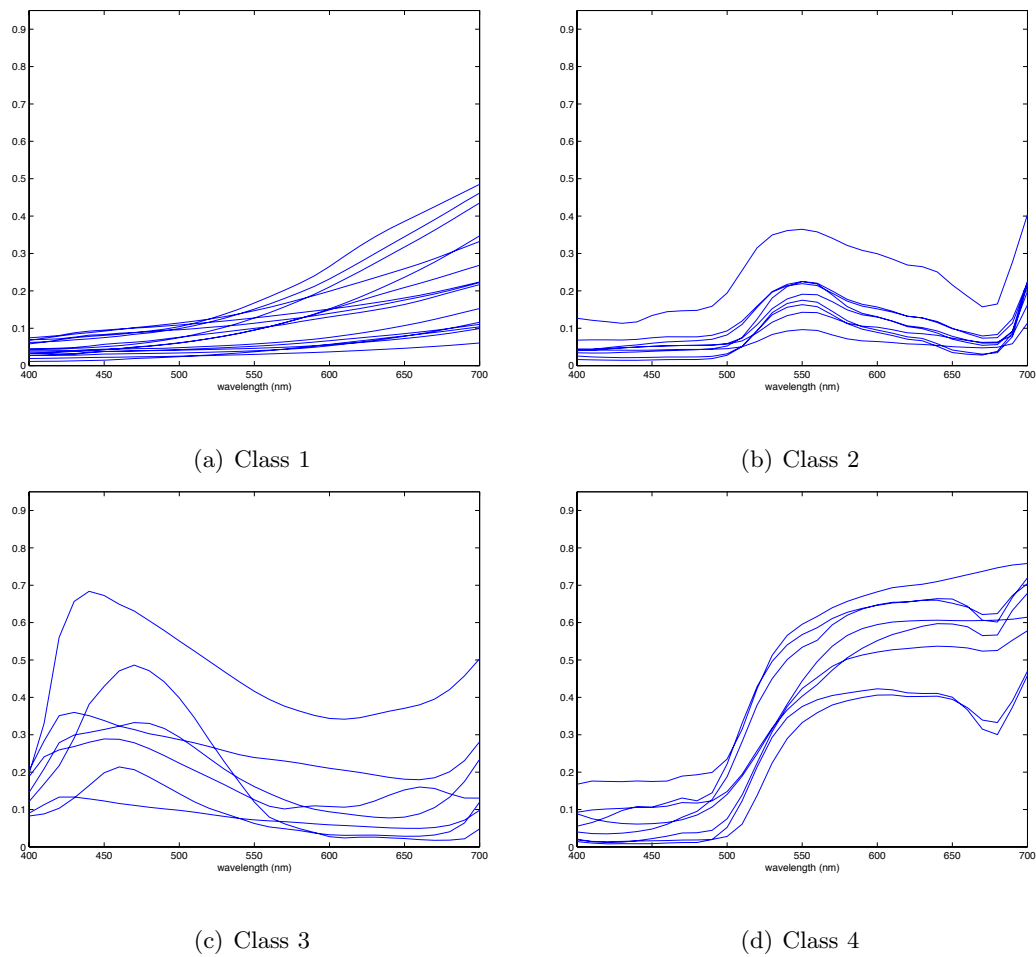


Figure 5.9: Class Basis Spectra for the First Set of Spectral Signatures

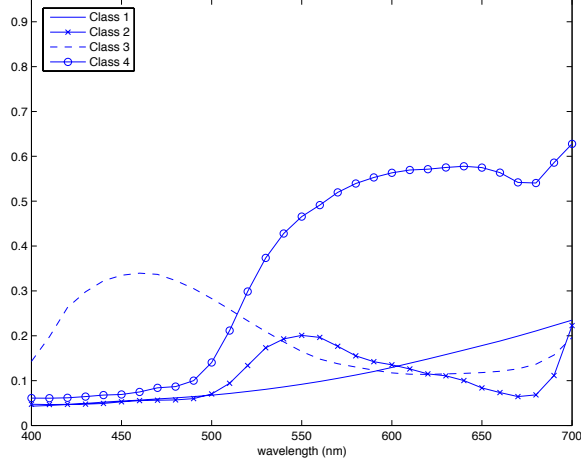


Figure 5.10: Estimates of the Class Spectral Signatures

where  $\sigma_\eta^2$  denotes the variance of the noise for all wavelengths sensed by the hyperspectral camera. Thus, it is assumed that the variance of the measurement errors are the same for every hyperspectral band. For a specific SNR,  $SNR_{dB}$ , the variance  $\sigma_\eta^2$  is defined as

$$\sigma_\eta^2 = \frac{\|\mathbf{S}\mathbf{a}\|_2^2}{M10^{\frac{SNR_{dB}}{10}}}. \quad (5.24)$$

Thus, for an observed spectral density,  $\mathbf{r}$ , the parameter  $\delta_\eta$  is given by

$$\delta_\eta = \frac{\|\mathbf{r}\|_2^2}{10^{\frac{SNR_{dB}}{10}}}. \quad (5.25)$$

The DPOCS method combines the signal independent and signal dependent noise sources into the parameter  $\delta'_v = \delta_\eta + \rho\delta_\mathbf{a}$ . The parameter  $\delta_\eta$  was set to the value defined in Eq. (5.25). The parameter  $\delta_\mathbf{a}$  approximates  $\|\mathbf{a}\|_2^2$  and was tested for estimate defined in Eq. (4.7), since it was determined that this produced more accurate results for the application of finite mixtures of probability densities. Recall that the parameter  $\rho$  approximates  $E\|\Delta\mathbf{S}\|_2^2$  and is computed from the sample autocovariance matrices  $\mathbf{C}_k$ , for  $k = 1 \cdots 4$ , according to Eq. (4.5).

The EVPOCS scheme also accommodates the perturbations  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$  via the positive parameters  $\nu$  and  $\tau$ , respectively defined in Eq. (4.14) and Eq. (4.15). The parameter  $\delta_\eta$  is given by Eq. (5.25). To ensure  $\tau > 0$ ,  $E\|\Delta\mathbf{S}\|_2^2$  must be strictly less than  $\sigma_K^2(\bar{\mathbf{S}})$ , where  $\sigma_K(\bar{\mathbf{S}})$  is the smallest singular value of  $\bar{\mathbf{S}}$ . The singular values of  $\bar{\mathbf{S}}$  defined by

the mean spectral densities in Figure 5.10 are 2.6321, 0.8975, 0.2551, and 0.0981. Following Eq. (4.5),  $\rho$  is computed to be 0.0165, which is greater than  $\sigma_K^2(\bar{\mathbf{S}}) = 0.0096$ . Thus, the EVPOCS method may not be implemented, since the magnitude of the noise exceeds  $\sigma_K^2(\bar{\mathbf{S}})$ . Recall that to guarantee that the set  $\Gamma$  is convex the condition  $\frac{\nu}{\tau} < \sigma_K^2(\bar{\mathbf{S}})$  must be satisfied. Note that the DPOCS formulation does not impose constraints on  $\rho$ , for convexity is guaranteed as long as  $\delta'_v$  is nonnegative. However, as  $\rho$  increases the size of the DPOCS set,  $S_{v'}$ , increases.

To evaluate the performance of the EVPOCS scheme, the basis spectra were re-defined by removing spectra from the clusters to reduce the variations of the classes. This reduces the power of the noise  $\Delta\mathbf{S}$ . The modified set of class basis spectra are shown in Figures 5.11(a) - 5.11(d). It is apparent that the variation of the classes has been reduced. The estimates of the mean spectral densities, henceforth denoted by the matrix  $\bar{\mathbf{S}}_a$ , for each class using the modified set of basis spectra are shown in Figure 5.12. The singular values of  $\bar{\mathbf{S}}_a$  for this new set are 2.7814, 0.8628, 0.3418, and 0.1324. For this alternate set of basis spectra, the parameter  $\rho_a$ , using Eq. (4.5), is given by 0.0057, which is less than the square of the smallest singular value of  $\bar{\mathbf{S}}_a$ ,  $\sigma_K^2(\bar{\mathbf{S}}_a) = 0.0175$ . The trace calculated for the four sample autocovariance matrices,  $\mathbf{C}_k$  corresponding to the classes 1 through 4 are 0.0026, 0.0023, 0.0054, and 0.0057, respectively. Notice that the variance for class 4 is the greatest, and the variance for classes 3 and 4 is well represented by the value chosen for  $\rho_a$ . The other two classes do not exhibit the variation indicated by  $\rho_a$ . The parameter  $\tau$  was tested for the lower bound defined by Eq.(4.15). The parameter  $\nu$  is set to  $\tau\rho_a + \delta_\eta$ , where  $\delta_\eta$  is given by Eq. (5.25).

The GEVPOCS method incorporates both signal independent and signal dependent noise. Like the EVPOCS scheme,  $\|\Delta\mathbf{S}\|_2^2$  must not exceed  $\sigma_k^2(\bar{\mathbf{S}})$ , so the GEVPOCS method is only appropriate when the basis spectra correspond to  $\bar{\mathbf{S}}_a$ . Thus, the parameter  $\rho_a$  defined for the EVPOCS scheme also estimates  $\|\Delta\mathbf{S}\|_2^2$  for GEVPOCS.

The LS solution was tested as the initial estimate for the SPOCS scheme. The DPOCS, EVPOCS, and GEVPOCS formulations were tested using the TLS solution as the initial estimate. The results from the finite mixtures of probability densities revealed that when the uniform estimate is the initial estimate the set theoretic methods produced poor estimates. Furthermore, it was observed that for some of the set theoretic methods the finite mixture model set contained the uniform contribution estimate when the true solution was not actually the uniform contribution estimate. Simulations of spectral densities revealed

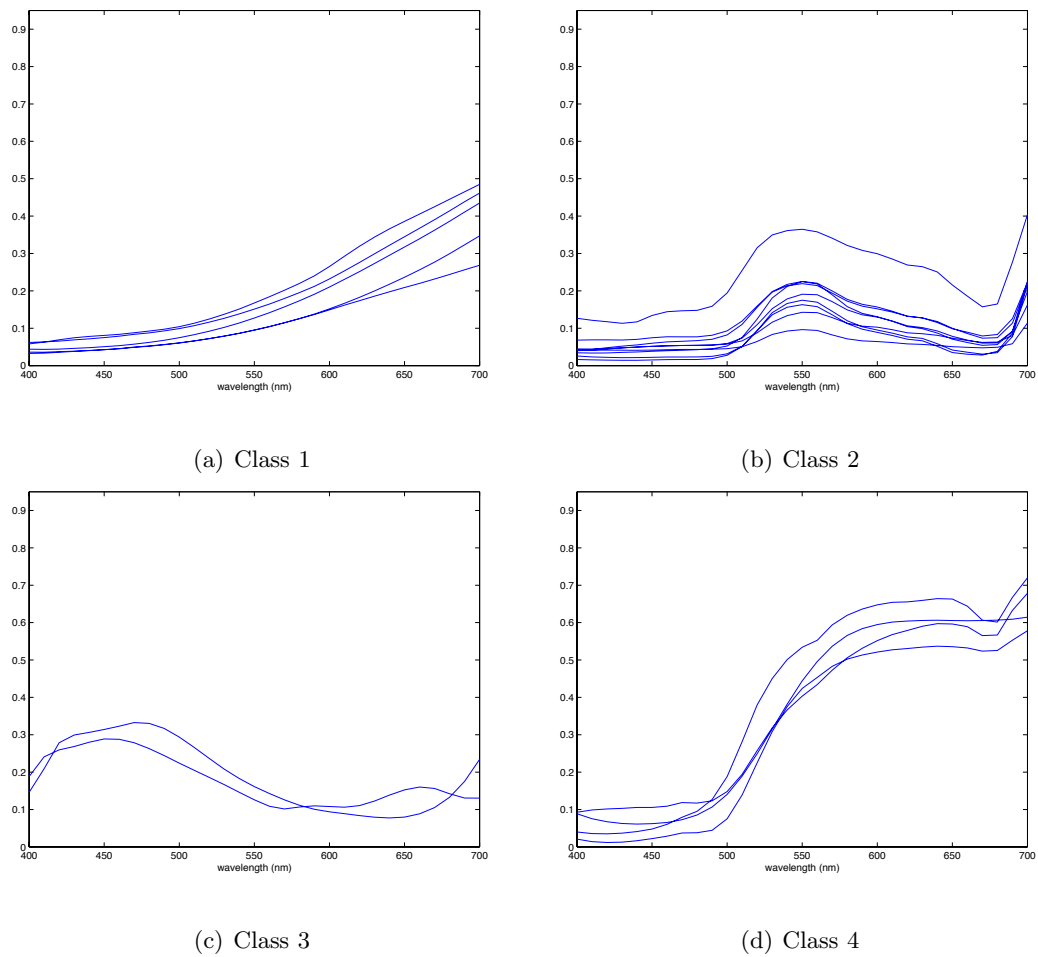


Figure 5.11: Modified Class Basis Spectra for the First Set of Spectral Signatures

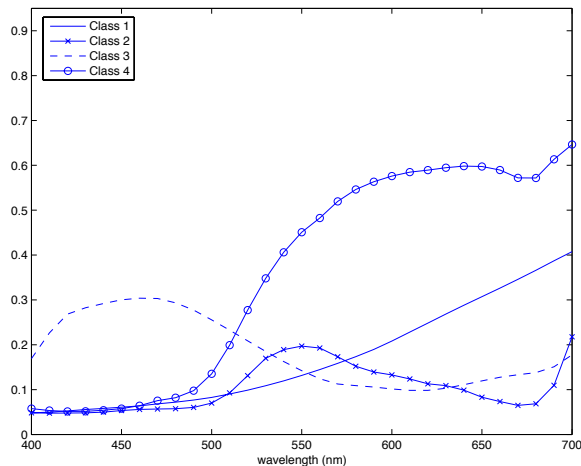


Figure 5.12: Estimates of the Class Spectral Signatures

that the using the uniform contribution estimate did not lead to accurate estimates and converged very slowly. Thus, the results when the uniform estimate was tested as the initial estimate are not included for the spectral mixture application.

### 5.2.1 Target Contribution Simulation

Determining the presence of a chosen target class in an observed spectral density is the essence of the target detection problem. The performance of a method relates to the ability to detect the presence of the target class when its contribution to the observed mixture density is relatively small. The performance is also related to the spectral density of the target class relative to the other spectral classes. The contribution of a target class that is “similar” to one or more the other classes may be difficult to accurately resolve.

To test the estimation methods, spectral mixtures simulating hyperspectral pixels were constructed with specified contributions of the target class under consideration. A hyperspectral pixel was generated according to Eq. (5.22) using the basis spectra corresponding to  $\bar{S}_a$ . One of the spectral classes was selected as the target class. The contributions of the remaining  $K - 1 = 3$  classes varied randomly in the observed mixture such that the contributions from the  $K$  classes sum-to-one. The effect of the contributions from the target class were evaluated from 0 to 0.3 at 0.05 increments, 0.3 to 0.7 at 0.1 increments, and 0.7 to 1 at 0.05 increments. Each of the 17 target contributions was simulated for 500

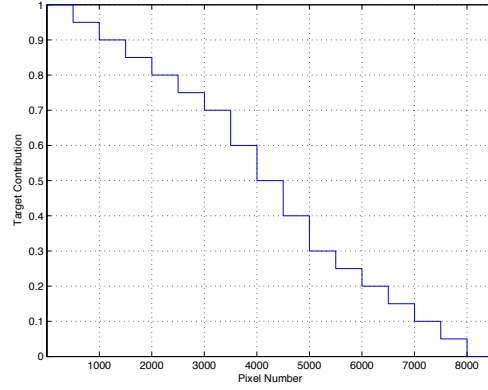


Figure 5.13: Actual target contributions for the target contribution simulation

hyperspectral pixels, leading to a total of 8500 pixels simulated overall. The smaller increments near 0 and 1 capture the behavior of the sum-to-one and nonnegativity constraints on the estimate of the target contribution. A plot of the actual target contributions is shown in Figure 5.13. Zero mean Gaussian noise was added to each pixel to simulate an SNR of  $SNR_{dB} = 30$  dB.

The basis spectra corresponded to the mean spectral densities in  $\bar{\mathbf{S}}_a$ , and estimates of the contributions were found using LS, TLS, SPOCS, DPOCS, EVPOCS and GEVPOCS. Note that the parameter  $\rho_a$  is used to estimate  $E\|\Delta\mathbf{S}\|_2^2$  for the DPOCS scheme, since the mean spectral densities  $\bar{\mathbf{S}}_a$  are used. The restriction that  $\|\Delta\mathbf{S}\|_2$  is less than the smallest singular value of the estimates of the mean spectral densities is satisfied and permits tests for the EVPOCS and GEVPOCS methods. While the other estimation methods do not impose restrictions on  $\|\Delta\mathbf{S}\|_2^2$ , this simulation seeks to illustrate and compare the performance of all the methods.

The performance of the estimation methods was assessed by estimating the relative MSEE statistics from the estimates of the contributions as defined by Eq. (5.2). Absolute estimation error statistics were calculated from the estimate of the target class contribution. Both error statistics were computed for each of the 17 different contributions of the target class to illustrate the behavior of the estimate methods with regard to the target class contribution. Figures 5.14-5.17 show the estimated absolute estimation error and the estimated relative MSEE for a particular spectral class selected as the target class.

Consider Figures 5.14(a) and 5.14(b), which show the estimated absolute estimation error and the estimated relative MSEE when class 1 is the target class. The LS and TLS solutions produce significantly worse estimates compared to the set theoretic estimates. Of course, neither the LS nor TLS solution satisfy the sum-to-one and nonnegativity constraints, so the dichotomy between the errors observed for the LS and TLS solutions and the set theoretic estimates is unsurprising.

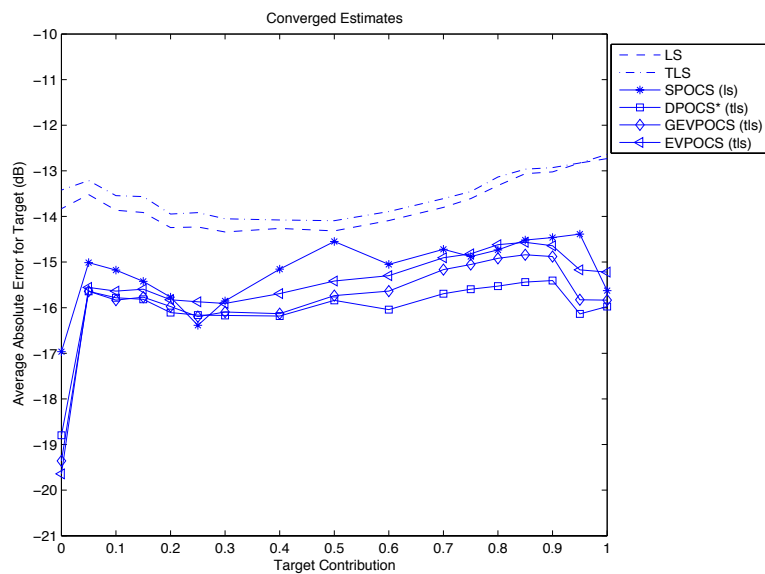
The average absolute errors for the target class from the SPOCS (ls) estimates follow a different trend as compared with the other set theoretic estimates. Nevertheless, the average absolute errors for the SPOCS (ls) estimates are similar to the average absolute errors for the other set theoretic estimates. However, as the contribution of the target class decreases, the estimated relative MSEE shows that the overall SPOCS (ls) estimate is much worse. This suggests that the SPOCS (ls) contribution estimates for the other classes become less accurate as the contribution of the target class decreases.

When estimating the contribution of the target class for smaller contributions of the target class, both the DPOCS (tls)\* and GEVPOCS (tls) methods exhibit very similar behaviors. As the contribution of the target class increases, the DPOCS (tls)\* method performs a little better than the GEVPOCS (tls) scheme. The EVPOCS (tls) method generates slightly worse estimates of the target contribution but significantly worse overall estimates of the class contributions.

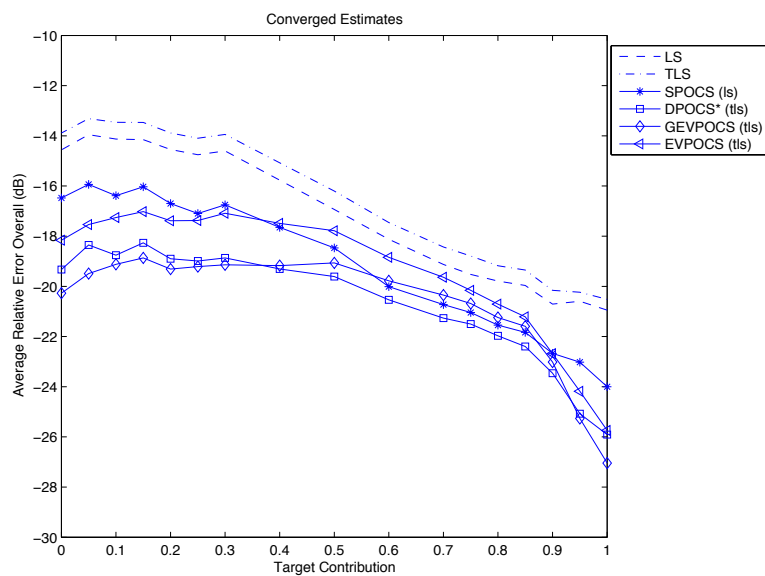
It is noted that the absolute errors of the DPOCS (tls), EVPOCS (tls), and GEVPOCS (tls) methods follow a common trend with respect to the contribution of the target class. For the relative MSEE, the SPOCS (ls) and DPOCS (tls)\* schemes follow similar trends while the EVPOCS (tls) and GEVPOCS (tls) follow another trend. The presence of these two trends reflects the similar construction of the finite mixture model sets, where the significant difference is the radius of the sets, i.e.  $\delta_v$ ,  $\delta_{v'}$ , and  $\nu$ .

The errors when class 2 is selected as the target class, shown in Figures 5.15(a) and 5.15(b), reveal behaviors similar to those observed with class 1 as the target class. It is noted that the estimated absolute estimation errors show greater separation among the various estimation methods. In fact, the DPOCS (tls)\* estimate tends to generate much better estimates of the target class as the target class contribution increases. On the other hand, the estimated relative MSEE for all the estimation methods are very similar to that observed for class 1 as the target class.

As observed in Figures 5.16(a) and 5.16(b), the errors associated with the LS and

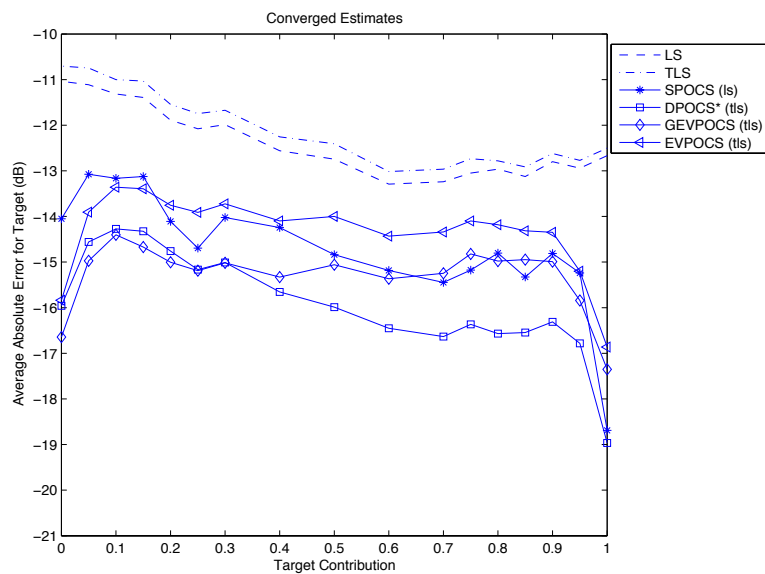


(a) Average Absolute Errors of the Target Class

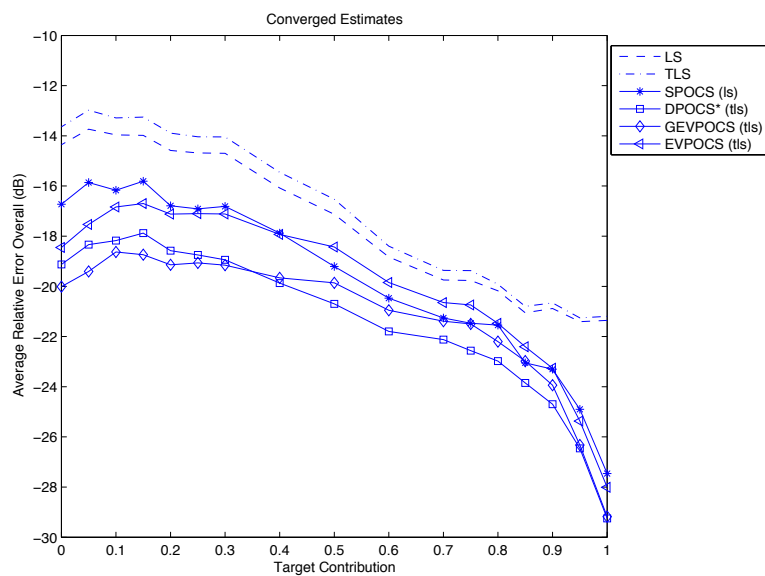


(b) Average Relative Errors Overall Classes

Figure 5.14: Errors with respect to the Target Proportion Using Class 1 as Target Class



(a) Average Absolute Errors of the Target Class



(b) Average Relative Errors Overall Classes

Figure 5.15: Errors with respect to the Target Proportion Using Class 2 as Target Class

TLS solutions when class 3 is selected as the target class appear significantly different. The estimated absolute estimation error for the target contribution is much more accurate for the LS and TLS solutions. To understand these errors, each class was analyzed to determine how well the other estimate class densities could approximate estimated class density from that class. Consider the linear system of equations

$$\bar{\mathbf{s}}_k = \hat{\mathbf{S}}_k \mathbf{x}, \quad (5.26)$$

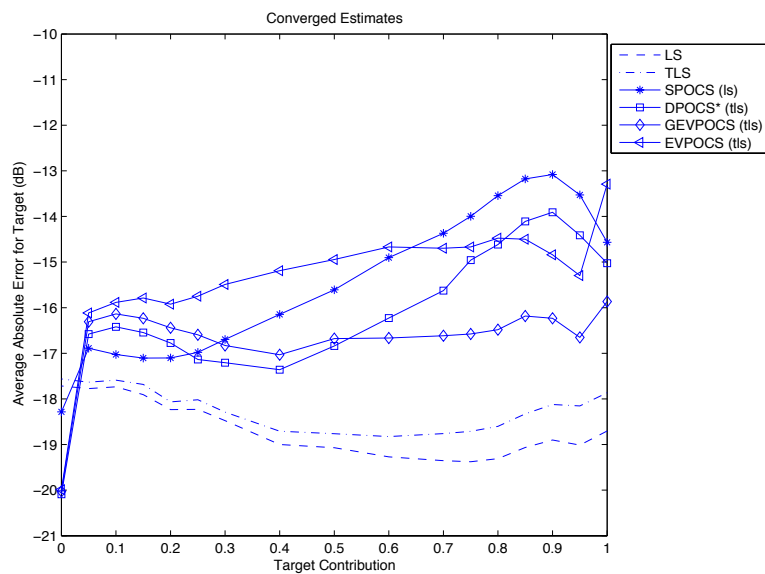
where  $\bar{\mathbf{s}}_k$  denotes the mean spectral signature for class  $k$  and the mean spectral signatures of the remaining three classes are given by the columns of the matrix  $\hat{\mathbf{S}}_k$ . The least squares solution of Eq. (5.26) is given as  $\hat{\mathbf{x}}_{ls} = \left( \hat{\mathbf{S}}_k^T \hat{\mathbf{S}}_k \right)^{-1} \hat{\mathbf{S}}_k^T \bar{\mathbf{s}}_k$ . A simple approach to quantify this relationship between the classes is to compute the Euclidean norm of the residual  $\|\bar{\mathbf{s}}_k - \hat{\mathbf{S}}_k \hat{\mathbf{x}}_{ls}\|_2$ . Distinct classes would lead to residuals with larger magnitudes. Table 5.8 lists the applicable elements of  $\mathbf{x}_{ls}$  for each class considered and the Euclidean norm of the residual  $\hat{\mathbf{x}}_{ls} = \left( \hat{\mathbf{S}}_k^T \hat{\mathbf{S}}_k \right)^{-1} \hat{\mathbf{S}}_k^T \bar{\mathbf{s}}_k$ . From Table 5.8, class 3, where  $\|\bar{\mathbf{s}}_3 - \hat{\mathbf{S}}_3 \hat{\mathbf{x}}_{ls}\|_2 = 0.5510$

Table 5.8: Comparison of the Representation of Spectral Classes by the other Classes

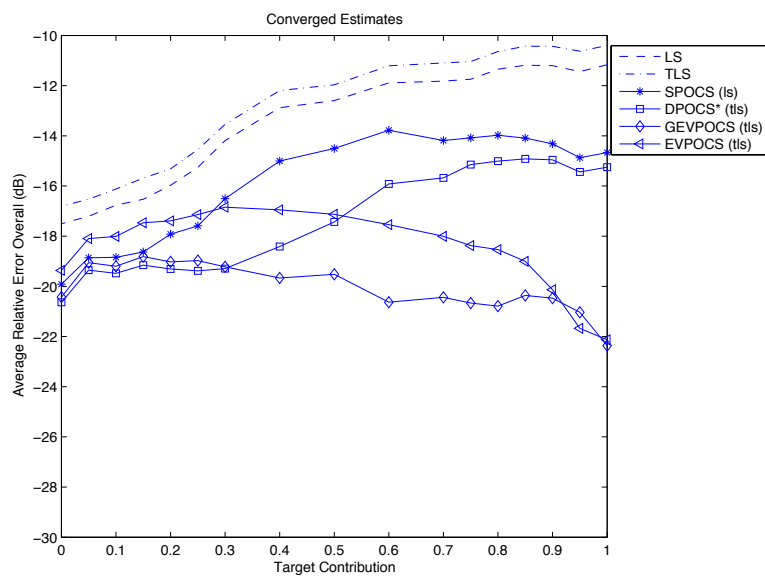
$\bar{\mathbf{s}}_k$	Class 1	Class 2	Class 3	Class 4	$\ \bar{\mathbf{s}}_k - \hat{\mathbf{S}}_k \hat{\mathbf{x}}_{ls}\ _2$
1	n/a	0.6030	0.2602	-0.8764	0.2189
2	0.4242	n/a	0.2369	-0.5323	0.1706
3	1.6483	2.4709	n/a	-1.0877	0.5510
4	1.4725	1.7055	-0.4193	n/a	0.3421

is much greater than the other classes. This indicates that class 3 has a unique signature that is not well represented by a combination of the other three classes. Under these circumstances, the LS and TLS solutions estimate the target contribution (class 3) well, since the target class is clearly characterized by a very distinctive spectral density. However, while the estimate of the target contribution is very accurate for the LS and TLS solutions, the estimated relative MSEE is very poor. Thus, the LS and TLS solutions generate poor estimates of the remaining classes. Both the SPOCS (ls) and DPOCS (tls)\* estimates become much worse as the target contribution increases. However, estimated relative MSEE from the EVPOCS (tls) and GEVPOCS (tls) estimates remain relatively consistent as the target contribution increases.

The errors for class 4 are shown in Figures 5.17(a) and 5.17(b). The DPOCS (tls)\* and GEVPOCS (tls) schemes provide the best estimates of the class contributions



(a) Average Absolute Errors of the Target Class



(b) Average Relative Errors Overall Classes

Figure 5.16: Errors with respect to the Target Proportion Using Class 3 as Target Class

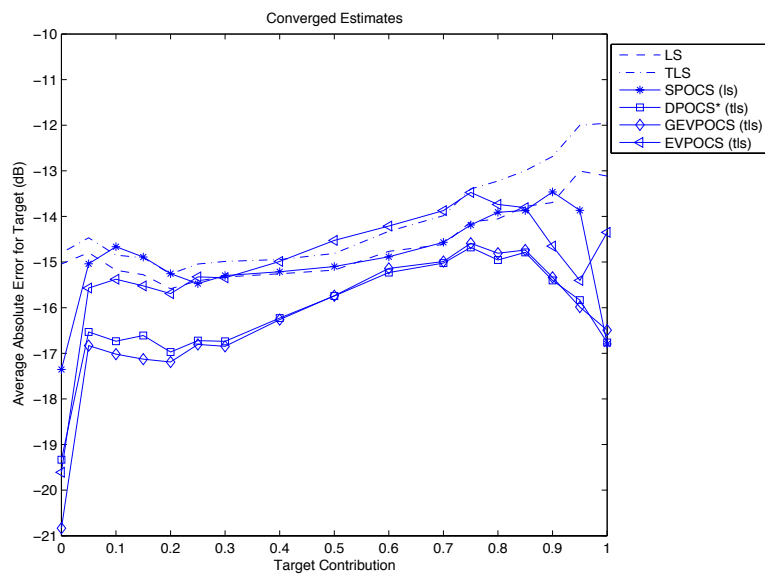
compared to the other estimates. The distinction of the spectral density for class 4 from wavelengths between 550 and 700 nm warrants this behavior. Furthermore, recall that the parameter  $\rho$  corresponds to the trace of the sample autocovariance matrix for class 4.

The number of iterations provides some indication of the speed of the set theoretic methods. The simulations for finite mixtures of probability densities revealed that the set theoretic methods often did not converge in less than 100 iterations. Table 5.9 lists the overall average number of iterations and the overall average percentage of converged estimates for the four set theoretic methods tested. These averages were computed from the four simulations, that is from each class tested as the target class. The SPOCS (ls) method converged for only 91.4618% of the simulations, which is less than the other methods. The DPOCS (tls)\* scheme converged more often, but more iterations were necessary to satisfy the convergence criteria. The EVPOCS (tls) method converged the most often with fewer iterations. This is a consequence of the magnitude of the parameter  $\nu$ , which controls the radius of the set  $\Gamma$ . The parameter approximates  $\tau E\|\Delta\mathbf{S}\|_2^2 + E\|\boldsymbol{\eta}\|_2^2$ , which is clearly larger than the set radius,  $E\|\boldsymbol{\eta}\|_2^2$ , defined for the GEVPOCS set. Finally, the GEVPOCS (tls) method converges almost as often as the EVPOCS (tls) method, but the number of iterations is the greatest. Recall that the DPOCS and GEVPOCS methods were shown to generate the most accurate estimates, and the GEVPOCS method performed best for small contributions of the target class. The accuracy of the estimates and frequency convergence due to the GEVPOCS (tls) method make it the most appealing among the set theoretic implementations.

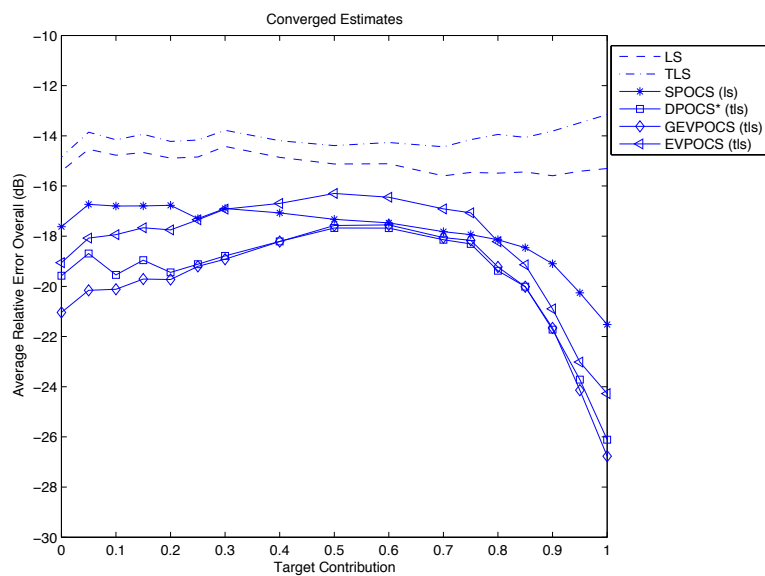
Table 5.9: Iteration Results for Set Theoretic Methods Estimates for Target Contribution Simulations

Method	Avg. Iterations	Converged Iterations
SPOCS (ls)	12.1265	91.4618
DPOCS (tls)*	14.3007	95.0647
EVPOCS (tls)	9.8591	99.1265
GEVPOCS (tls)	14.5946	98.1235

The simulations analyzing the effect of the target contribution verify that the set theoretic methods produce more accurate estimates of the target contributions. Though the set theoretic methods exhibit similar results, the DPOCS (tls)\* and GEVPOCS (tls) formulations lead to the most accurate estimates. Furthermore, the GEVPOCS (tls) method consistently provides accurate estimates when the target class contributes less to the ob-



(a) Average Absolute Errors of the Target Class



(b) Average Relative Errors Overall Classes

Figure 5.17: Errors with respect to the Target Proportion Using Class 4 as Target Class

served spectral density. If a threshold were established to decide if a target is present, then the GEVPOCS (tls) method would allow a lower threshold than the other methods. This will reduce the number of missed detections without a drastic increase in the false alarms, since the GEVPOCS (tls) method accurately estimates the target contribution. Target detection for such low contributions permits the observer to either maintain a safe distance from the target or use a lower spatial resolution camera to detect the target. The final simulation demonstrates the performs of the methods on a simulated hyperspectral image.

### 5.2.2 Target Detection with a Simulated Hyperspectral Image

The target contribution simulations established the behavior of the various estimation methods when estimating the class contributions for finite mixtures of spectral densities. In practice, an image containing mixtures of spectral classes captured by a hyperspectral camera may be analyzed for the presence of a chosen target class. This simulation models a simple hyperspectral camera that records an image containing pixels of mixed spectra.

A high spatial resolution image of pure spectra is generated first to simulate the capture of a hyperspectral image. Thus, each pixel of the generated image of pure spectra belongs to only one of the  $K = 4$  classes. This high spatial resolution image simulates the true spectra of a scene to be captured by a hyperspectral camera. The high resolution image is blurred with a point spread function (PSF) modeling a pinhole aperture. This mimics the perspective of a camera positioned at a great distance from the true scene. Simulated recording noise in a hyperspectral sensor is modeled by adding zero mean Gaussian noise to the blurred image to achieve a specified SNR. The original high resolution image imitates a real scene with infinite spatial resolution, so the image is sub-sampled to simulate an image recorded by a digital hyperspectral camera with  $M$  contiguous spectral bands.

A  $150 \times 150$  pixel hyperspectral image was generated following the hyperspectral image capture model described above. The second set of basis spectra pixels of pure spectra, associated with  $\tilde{\mathbf{S}}_a$ , were used to generate the sample pixel, which permits the use of the EVPOCS and GEVPOCS schemes. The pixels of the high spatial resolution images of pure spectra were produced according to Eq. (5.22), where  $\tilde{\mathbf{S}}_k$  represents the basis spectra of the  $k^{th}$  class. The PSF of the pinhole aperture has an  $11 \times 11$  region of support. Zero mean Gaussian noise is added to the blurred image to simulate an SNR of  $SNR_{dB} = 30$  dB. The image is sub-sampled by saving every  $10^{th}$  pixel, beginning with the top left corner of

the image. Figures 5.18(a) and 5.18(b) respectively show the image of pure spectra and the image of observed mixed spectra using the basis spectra associated with  $\bar{\mathbf{S}}_a$ . For figure 5.18(a) the colors have the following class associations: the brown pixels correspond to class 1, the green pixels denote class 2, the blue pixels represent class 3, and the yellowish-orange blocks come from class 4, the target class. The spectra were transformed according the procedure described in Section A to create these images.

The contributions were estimated pixel-by-pixel in the observed hyperspectral image. All of the estimation methods were employed to estimate the contributions of the spectral classes for the observed image shown in Figure 5.18(b), since the magnitude of the perturbations present did not violate any of the parameter constraints for the EVPOCS and GEVPOCS methods. For the set theoretic methods, the parameters  $\delta_\eta$ ,  $\rho$ , and  $\rho_a$  were computed as discussed for the target contribution simulation. The parameter  $\delta_a$ , introduced for the DPOCS scheme, was set to the value prescribed by Eq. (4.7). For EVPOCS, the effect of the estimate was evaluated for  $\tau$  given by the lower bound in Eq. (4.15).

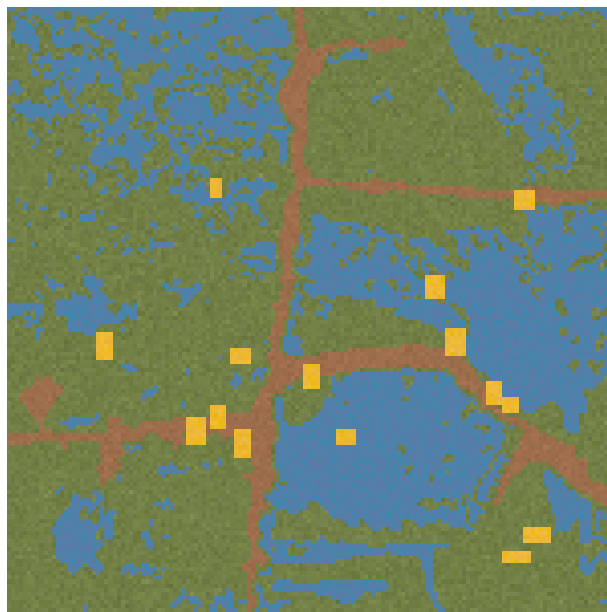
For the SPOCS scheme, the the LS solution was used as an initial estimate. The TLS solution served as the initial estimate for the DPOCS, EVPOCS, and GEVPOCS methods.

The actual contributions of the target class, class 4, are shown in Figure 5.19. The colorbar ranges from 0 to 0.4321 based on the estimates from all the different methods. The actual maximum contribution from the target class is 0.4321, which indicates that all the estimation methods underestimated the target class contribution. In this 225 pixel image, there are 28 pixels with a nonzero contribution from the target class.

The LS solutions of the target contributions are shown in Figure 5.20(a), and the distribution of the associated absolute errors are shown in Figure 5.20(b). Figures 5.21(a) and 5.21(b) show the TLS solution of the target contributions and the absolute errors, respectively. Neither the LS solution nor the TLS solution are guaranteed to satisfy both the sum-to-one and nonnegativity constraints. Thus, these solutions produce many negative values when the actual target contribution is close to or equal to zero. The scale used on the display of the estimates makes this phenomenon, but the presence of errors in the corresponding pixels verifies this observation.

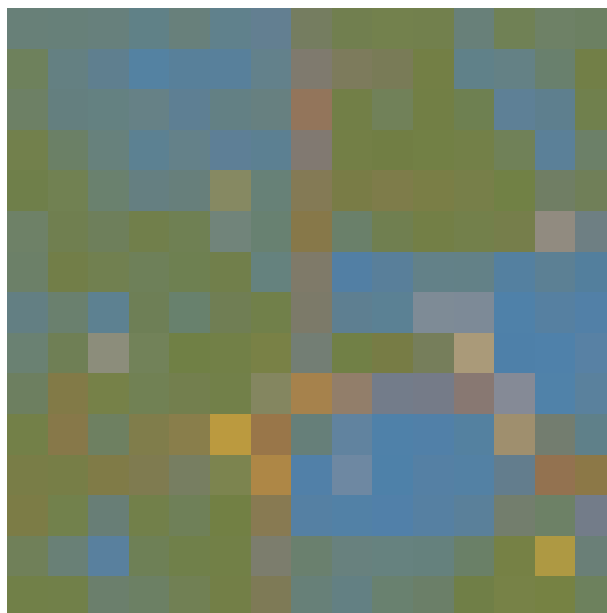
The set theoretic methods incorporate the sum-to-one and nonnegativity constraints. The SPOCS (ls) estimates of the target class contributions are shown in Figure 5.22(a), and the corresponding absolute errors are shown in Figure 5.22(b). Notice the

Original – Test Image



(a) Original Pure Spectra

Mixed Hyperspectral



(b) Observed Mixed Spectra

Figure 5.18: Comparison of the original image and the observed image. The observed image is obtained from the original image by blurring, subsampling, and adding noise to achieve an SNR of 30 dB. This image was constructed with the modified set of basis spectra.

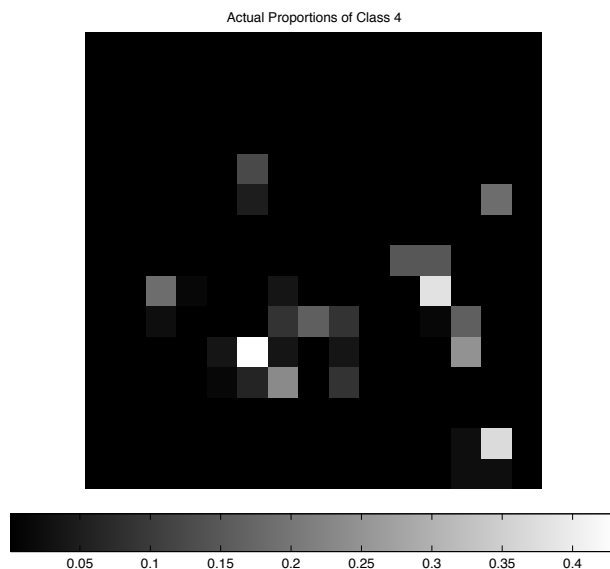


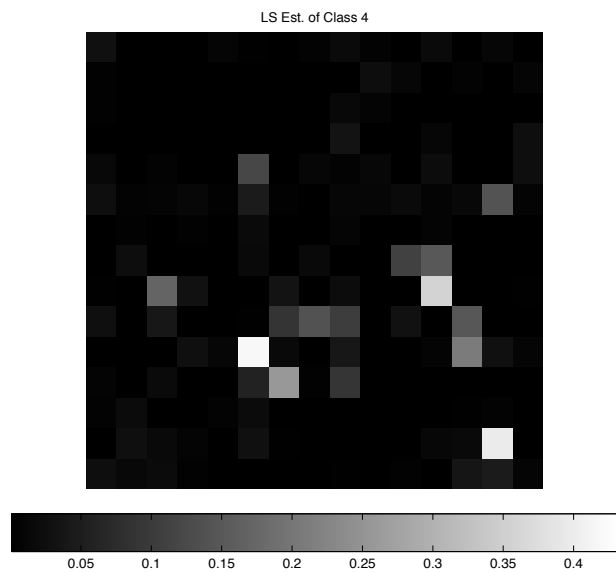
Figure 5.19: Actual contributions of class 4 (target)

significant reduction of the absolute errors. Much of this reduction reflects the utilization of both the sum-to-one and nonnegativity constraints on the LS solution.

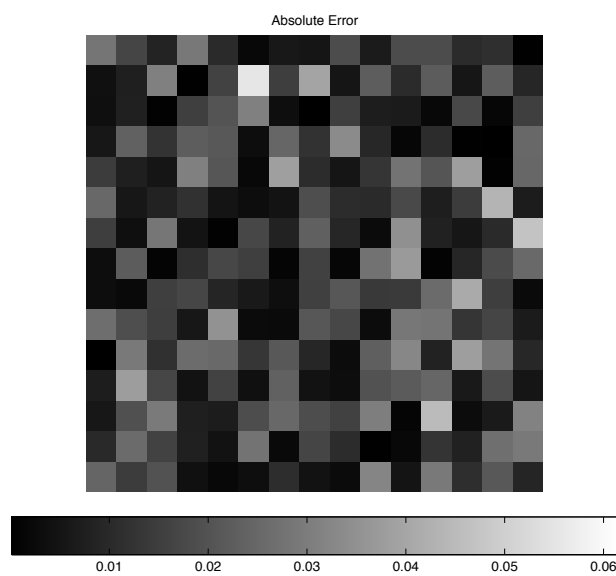
Figure 5.23(a) shows the DPOCS (tls)\* estimate, and Figure 5.23(b) shows the corresponding distribution of the absolute errors. The absolute errors in the lower region of the image are significantly reduced. In fact, the contributions throughout the image are more accurate than the other set theoretic estimates.

The EVPOCS (tls) and GEVPOCS (tls) estimates are shown in Figures 5.24(a) and 5.25(a), respectively. The absolute errors for the EVPOCS (tls) and GEVPOCS (tls) estimates, respectively shown in Figures 5.24(b) and 5.25(b), show smaller errors when the actual target contribution is approximately or equal to zero. The results for these two estimates are very similar, a pixel-by-pixel comparison is shown in Figure 5.26. In this figure, the columns of pixels from Figures 5.24(b) and 5.25(b) are interleaved, beginning with the absolute errors from the EVPOCS (tls) estimate. From this comparison, it is noted that the GEVPOCS (tls) errors are slightly smaller.

The errors for the various estimation methods of the simulated  $15 \times 15$  hyperspectral image are compiled in Table 5.10 and 5.11. Table 5.10 includes the average overall MSEE and, where applicable, the average number of iterations. Recall that “Uniform” refers to the uniform contribution estimate, i.e. guessing equal contribution from each class

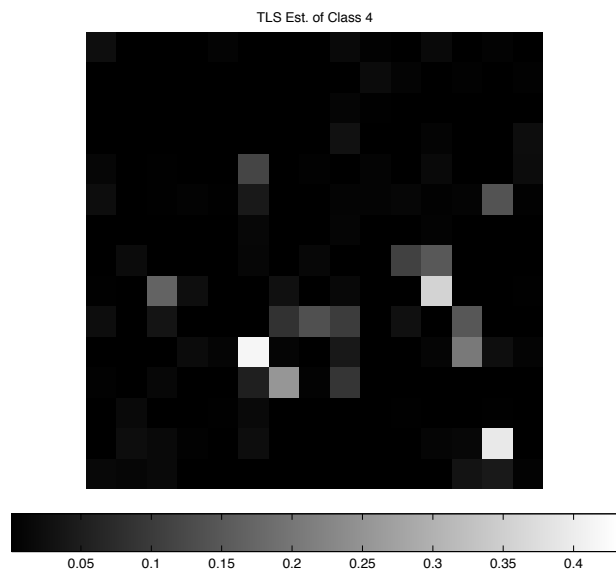


(a) estimate

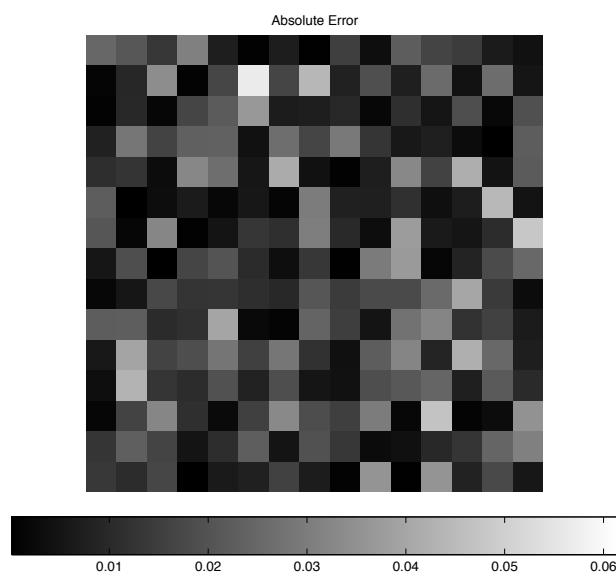


(b) absolute errors

Figure 5.20: LS solution of class 4 (target) contributions

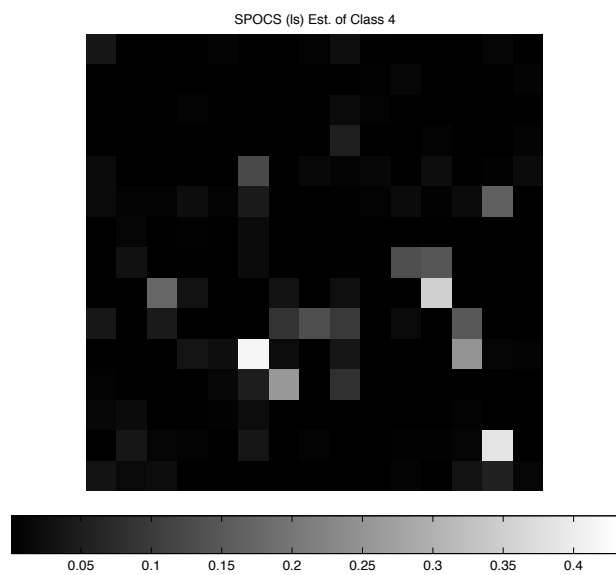


(a) estimate

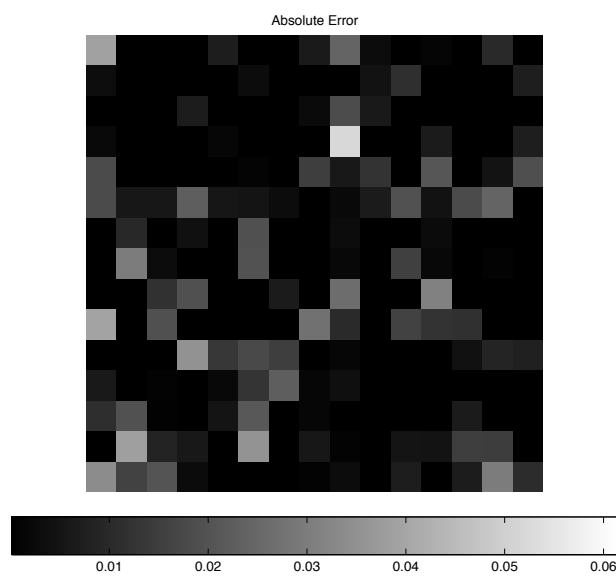


(b) absolute errors

Figure 5.21: TLS solution of class 4 (target) contributions

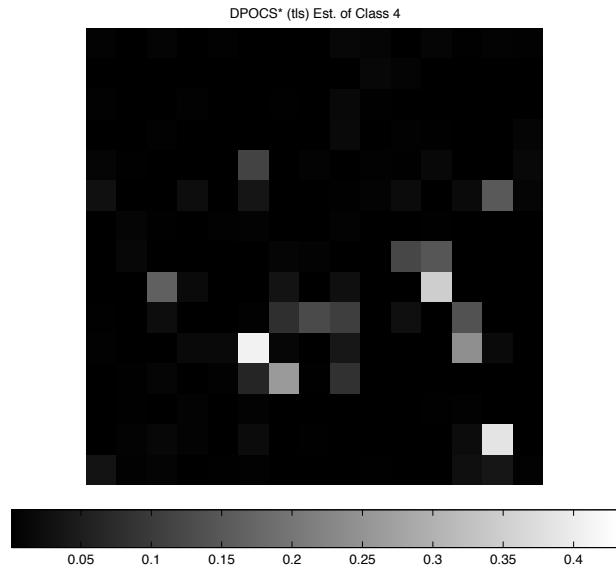


(a) estimate

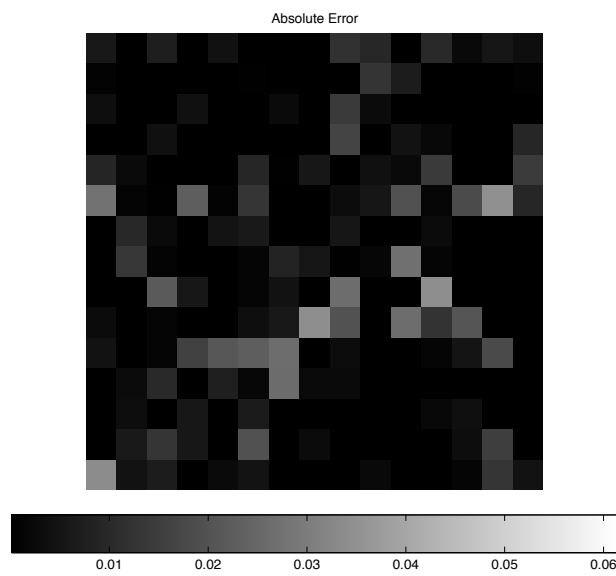


(b) absolute errors

Figure 5.22: SPOCS (ls) estimate of class 4 (target) contributions

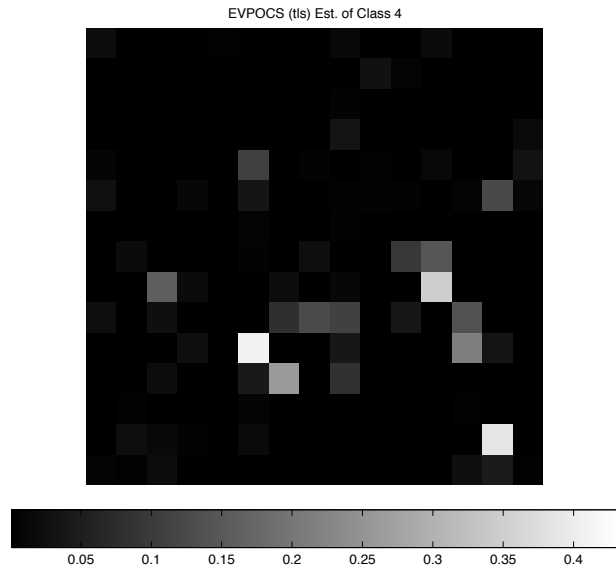


(a) estimate

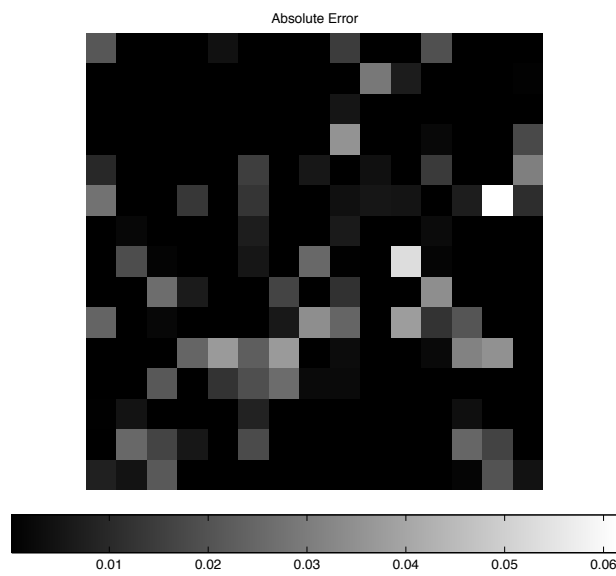


(b) absolute errors

Figure 5.23: DPOCS (tls)\* estimate of class 4 (target) contributions

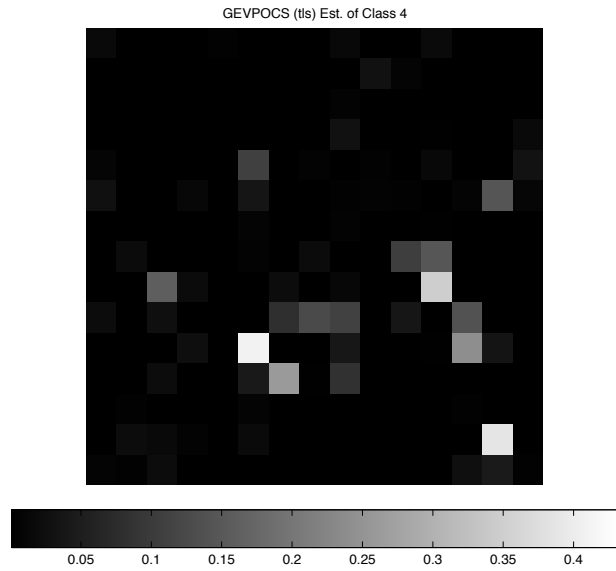


(a) estimate

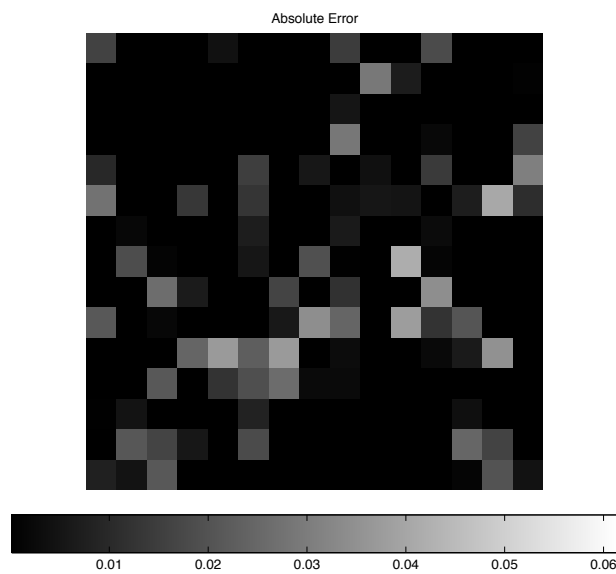


(b) absolute errors

Figure 5.24: EVPOCS (tls) estimate of class 4 (target) contributions



(a) estimate



(b) absolute errors

Figure 5.25: GEVPOCS (tls) estimate of class 4 (target) contributions

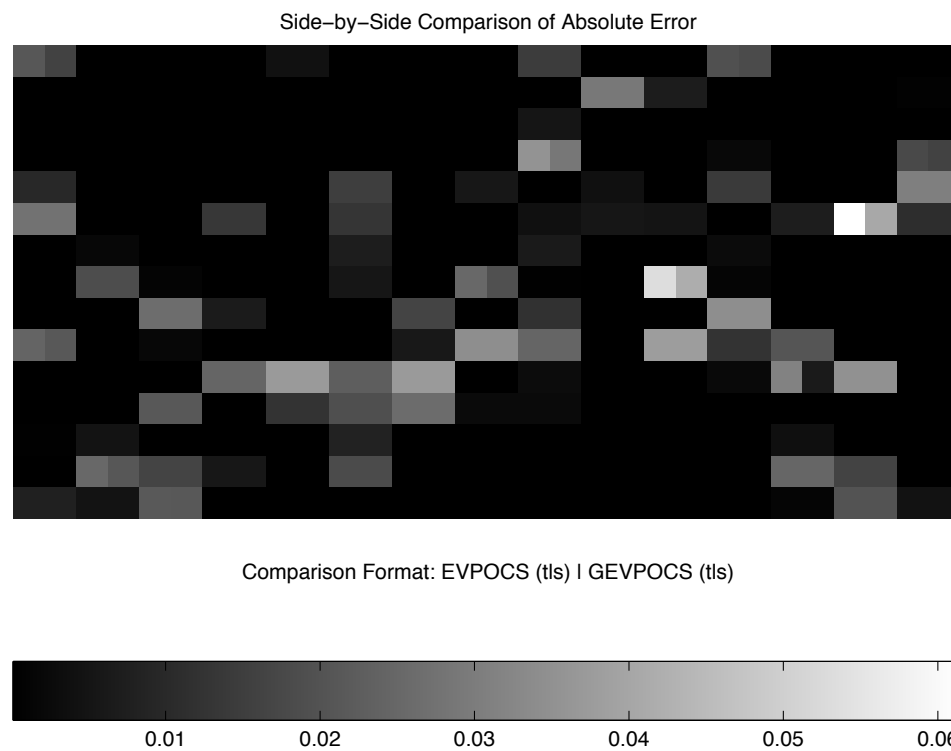


Figure 5.26: Side-by-side (pixel-by-pixel) comparison of EVPOCS (tls) and GEVPOCS (tls) Absolute Errors. The columns of Figures 5.24(b) and 5.25(b) are interleaved beginning with the pixels corresponding to the EVPOCS (tls) errors in the far left column.

Table 5.10: Overall Results for Hyperspectral Mixtures from 225 Pixels Using Modified Basis Set of Spectral Densities

<i>Method</i>	<i>Estimated Relative MSE (dB)</i>	<i>Avg. Number of Iterations</i>
LS	-23.16	n/a
TLS	-22.54	n/a
SPOCS (ls)	-27.95	17.32
DPOCS (tls)*	-28.51	16.36
EVPOCS (tls)	-25.75	7.26
GEVPOCS (tls)	-25.75	6.69
Uniform	-2.22	n/a

Table 5.11: Average Absolute MSE (decibels) of Individual Class Estimates for Hyperspectral Mixtures from 225 Pixels Using Modified Basis Set of Spectral Densities

<i>Method</i>	<i>Class 1</i>	<i>Class 2</i>	<i>Class 3</i>	<i>Class 4</i>
LS	-31.58	-27.59	-40.41	-34.73
TLS	-31.36	-26.85	-40.12	-34.34
SPOCS (ls)	-38.32	-33.01	-37.17	-39.14
DPOCS (tls)*	-39.43	-34.94	-35.75	-41.10
EVPOCS (tls)	-38.63	-32.12	-32.48	-38.50
GEVPOCS (tls)	-38.81	-32.43	-32.71	-39.32
Uniform	-12.60	-6.52	-9.93	-12.33

in the mixture. Table 5.11 shows the average absolute error, as defined in Eq. (5.3), for each class. From both tables it is clear that the DPOCS (tls)\* estimate produces the best feasible estimate of the class contributions. Unfortunately, the DPOCS (tls)\* scheme converges on average in 16.36 iterations, and the next best feasible estimates result from the SPOCS (ls) method, which has a slightly slower convergence rate on the average. The GEVPOCS (tls) estimate provides the next best feasible estimates with significantly fewer iterations on average. It is noted that the LS and TLS methods estimate class 3 exceptionally well. This result correlates to the discussion in Section 5.2.1 on page 86 explaining this behavior for the target proportion estimate simulation with target 3 as the target class.

Comparing the DPOCS (tls)\* and GEVPOCS (tls) results in Figures 5.12 and 5.12, it is noted that the GEVPOCS (tls) method generates fewer errors when the target is not actually present. This implies that the GEVPOCS (tls) estimate contains fewer false alarms. Table 5.12 lists the number of false alarms for each POCS method. The false alarm

Table 5.12: False Alarms Comparisons for POCS Estimates of Hyperspectral Mixtures

<i>Method</i>	<i>No Threshold</i>	<i>Thresholding at 0.01</i>	<i>Max. False Alarm Est.</i>
SPOCS (ls)	52	17	0.0521
DPOCS (tls)*	59	12	0.0337
EVPOCS (tls)	31	14	0.0378
GEVPOCS (tls)	31	14	0.0378

count in Table 5.12 was computed by counting the number of nonzero absolute errors for pixels when the target did not contribute. The false alarm counts were determined from the various POCS estimates before and after applying a threshold of 0.01 as indicated in Table 5.12. The actual target contribution is zero for 197 of the 225 pixels in the image. With no thresholding, the GEVPOCS (tls) and EVPOCS (tls) estimates produce far fewer false alarms than the DPOCS (tls)\* and SPOCS (ls) estimates. Using a threshold to eliminate estimates below 0.01, the SPOCS (ls) estimate has greatest number of false alarms. The other three POCS estimates have approximately the same number of false alarms. The last column of Table 5.12 shows the maximum estimate determined by the methods when the target is actually absent. This indicates that a threshold removing estimates of the target class less than 0.04 would eliminate all the false alarms for the DPOCS (tls)\*, EVPOCS (tls), and GEVPOCS (tls) estimates. However, the smallest actual target contribution is 0.0124, so a threshold at 0.04 would miss some targets. The threshold set at 0.01 is reasonable and drastically reduces the number of false alarms for all the POCS estimates of the target class in this simulated image.

### 5.2.3 Discussion

The simulations for finite mixtures of spectral densities reveal the robust behavior of the set theoretic methods. While all of the POCS methods generate much better results than the LS and TLS, the GEVPOCS (tls) avoids overestimating the contribution of the target when it is actually absent. Indeed, it was noted that a threshold of 0.01 eliminated many false alarms for the POCS methods. This produced nearly the same number of false alarms for the DPOCS (tls)\* and GEVPOCS (tls) estimates. On the other hand, the DPOCS (tls)\* is noted to require significantly more iterations on average to converge as compared to the GEVPOCS (tls) method. The added processing to eliminate the ex-

cessive number of false alarms for the DPOCS (tls)\* method may be undesirable, and the unprocessed GEVPOCS (tls) estimate may be preferred. Even if a threshold is applied to determine if a target is present, the GEVPOCS (tls) estimate required fewer iterations to converge, which is appealing. Nevertheless, an important feature of the GEVPOCS (tls) method noted from the target contribution simulations is the tendency to avoid estimating the presence of the target class when the target class does not actually contribute to the observed density.

## Chapter 6

# Conclusions

This work addressed the estimation of univariate finite mixture models using discrete approximations of the mixture components. The presence of applications encountering finite mixtures containing nonparametric components motivated the investigation of methods to investigate such mixture models. The mixture components were estimated from samples. When the components were given by probability densities, histograms of a collection of scalar samples served as approximations of the mixture components. In more general models, a vector or finite length discrete signal approximated the mixture components. The proposed methods incorporate the first and second order statistics obtained from the collection of vectors representing samples of the components.

The proposed methods, based on the set theoretic method of sequential projections onto convex sets (POCS), estimate the contributions of the components of finite mixture models. The proposed method found a feasible solution in the intersection of three sets: a finite mixture model set, the sum-to-one set, and the nonnegativity set. Four finite mixture model sets, distinguished by the characterization of the model errors, were studied. The DPOCS, EVPOCS, and GEVPOCS sets included the estimated second order statistics of the approximations of the mixture components.

The GEVPOCS set is introduced as a generalization of the EVPOCS set described in [12]. Recall that the goal of POCS was to find feasible solution in the intersection of sets describing the known properties of a desirable solution. The GEVPOCS set was shown to be smaller than the EVPOCS set, which eliminates many points in the set. The reduction

in the size of the set results in more iterations to converge when the initial estimate did not lie in the GEVPOCS set.

Comparisons of the proposed POCS methods with the EM algorithm for finite mixtures with parametric components indicated the viability of the new methods. The underlying assumptions of the EM algorithm differ from the proposed methods. The EM algorithm required a collection of scalar observations to determine the component contributions, while the proposed method assumed an aggregate representation of the collection of observed samples. In fact, the observations may be available only in an aggregate form, such as a hyperspectral pixel. Such representations prohibit estimation via the EM algorithm and increase the value of the POCS methods.

The EM algorithm, which estimated the parameters and the contributions of the components, performed poorly when the collection of scalar observations constituting a mixture is small. In addition, when a class contributed very few scalar samples to the observed collection, the EM algorithm produced unsatisfactory estimates, since it attempts to fit a finite mixture with certain number of components to the observed collection. The proposed method was unaffected if the expected number of classes contributing to a mixture is greater than the number actually present. However, underestimating the number of components could lead to erroneous proportion estimates. To address the poor performance with the EM algorithm, a modified form of the EM algorithm, which uses estimates of the parameter components from previously obtained labeled scalar samples, was implemented to provide a comparison with the new method for finite mixtures of parametric components.

The proposed methods performed similarly for simulations of finite mixtures of nonparametric components. The comparable performance stemmed from the use of histogram approximations of the component densities, which removed restrictions on the actual form of the component densities. Thus, the simulations for finite mixtures of nonparametric components were redundant and show that the proposed methods were unaffected by the nonparametric form of the component densities.

The proposed methods were applied to finite mixtures of spectral densities, specifically treating mixtures observed in hyperspectral pixels. Simulations showed slightly improved performance of about 0.5 dB with the POCS methods that incorporated the second order statistics of the perturbations  $\Delta\mathbf{S}$  and  $\boldsymbol{\eta}$  in comparison to those that did not incorporate the perturbations  $\Delta\mathbf{S}$ . This marginal improvement observed suggested that accounting for the perturbations  $\Delta\mathbf{S}$  was of little practical value. Nonetheless, the proposed method

addresses the uncertainty  $\Delta\mathcal{S}$ , which is disregarded in most studies of mixtures of spectral densities.

Of all the POCS methods tested for mixtures of spectral densities, the hyperspectral image simulation revealed that the new GEVPOCS method reduced the number of false alarms when estimating the target class. Furthermore, the target proportion simulations showed more accurate estimates by the GEVPOCS method for small target contributions in observed mixtures.

Accuracy at lower contributions of the target class is very important. This allows an observer to view the target at a much greater and safer distance. Alternatively, a lower spatial resolution camera may be utilized, since the geometry of the target need not be resolved in the captured image.

However, it may be beneficial to investigate methods that combine spectral density information with the observed geometric features.

The proposed methods assumed prior knowledge of the number of classes contributing to a mixture. In many realistic cases, knowledge regarding the number of classes may not be available, for other unknown classes may contribute to the mixture. Furthermore, inadequate knowledge about the form of the mixture may coincide with the presence of an unknown but important class. Several approaches mentioned in Section 1.1.2 address this problem with unsupervised methods when many mixtures are observed, such as in hyperspectral images. The methods assume a specific number of components and the presence of observations representing the contribution of only one class for characterization of the class. The latter assumption may not be appropriate for all applications, especially when the objective is to detect the target class when its spectral density is mixed with the spectral densities of the surrounding materials. These and many other aspects are critical to developing more general algorithms and remain to be addressed.

## Appendix A

# Generation of Color Images from Hyperspectral Data

This section explains the procedure to generate color images from hyperspectral data sensed in the human visible spectrum. The color images generated from hyperspectral data in this document follow underwent this transformation.

MATLAB displays color images represented by values corresponding to the sRGB color space. Color images captured by consumer digital cameras typically record the color information with respect to the sRGB color space. The sRGB values generated by such cameras vary according to the camera manufacturer and chosen color sensors. The implementations tend to be very similar.

A  $P \times Q$  digital image, where  $P$  and  $Q$  denote the spatial dimensions of the image, in the sRGB color space may be represented by 3,  $P \times Q$  matrices, where each matrix contains either red, green, or blue color information for the image. Each pixel in the image is defined by a three component vector containing the tristimulus values with respect to the sRGB color space. Hyperspectral data with  $M$  bands describes the image with respect to  $M$  wavelengths, and a  $P \times Q$  hyperspectral image is represented by  $M$ ,  $P \times Q$  matrices. The  $m^{th}$  matrix contains the spectral radiation sensed at the  $m^{th}$  wavelength. Manipulations must be performed to display hyperspectral data in the sRGB color space.

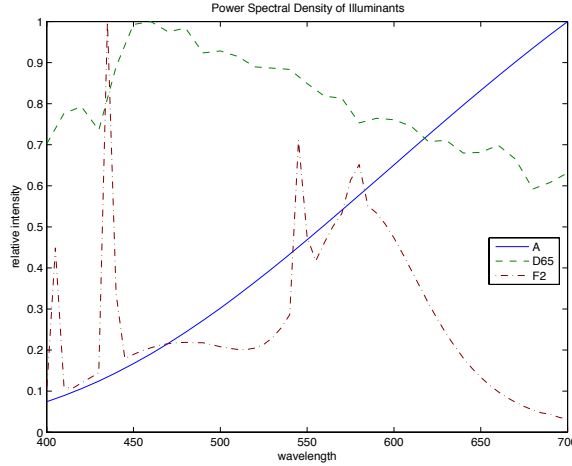


Figure A.1: Power Spectral Distribution of Illuminants A, D65, and F2 in the Visible Spectrum

To understand the manipulations, a short overview of hyperspectral data is necessary. Most objects reflect incident radiation. Accordingly, such objects may be described by a reflectance spectrum,  $\mathbf{r}$ . The incident radiation often is a light source, or illuminant. The relative spectral power distributions of several illuminant as a function of wavelength have been tabulated by Commission Internationale de l’Eclairage (CIE). Figure A.1 shows the relative power spectral distributions of three common CIE illuminants: A, D65, and F2. Note that the illuminants have been scaled such that the maximum spectral power is 1; hence, this figure compares only the shape of the three illuminants. Illuminant A corresponds to an incandescent lamp. The greater power at wavelengths longer than 600 nm justifies the orange hue frequently apparent in indoor photographs. Illuminant D65 represents the scaled relative spectral power of daylight. Illuminant F2 represents the scaled relative spectral power of a fluorescent illuminant. The F2 illuminant is shown at a 5 nm resolution, and the other two illuminants are shown with a 10 nm resolution. Illuminants A and D65 do not contain peaks as seen in the spectral of the F2 illuminant and are well sampled at 10 nm [45]. For the F2 illuminant, significant power is present in the wavelengths corresponding to blue and yellow color.

Figure A.2 illustrates the impact of the illuminant with the hyperspectral image used for simulations described in Section 5.2.2 under various illuminants. Notice that illuminant A results in a very orange hue across the image. The D65 illuminant mimics the presence of typical daylight and is used for the hyperspectral images shown in Figures

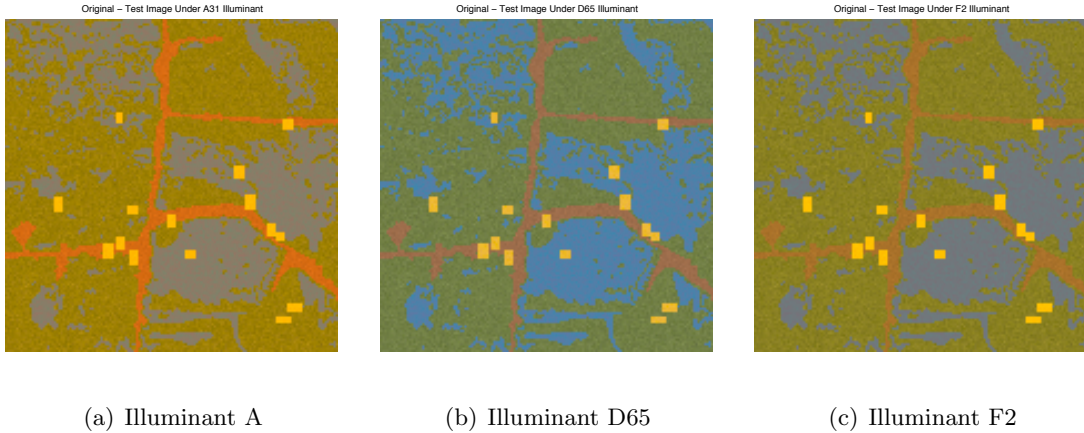


Figure A.2: Comparison of illuminants applied to hyperspectral image used for simulations

5.18(a) and 5.18(b). Note that the hyperspectral data is not scaled to reflect the presence of illuminant D65 when the analysis is performed. Thus, the D65 illuminant is applied only for display purposes. Illuminant F2 generates an image that has a yellow hue. The peak in the F2 illuminant at 440 nm corresponding to the color is suppressed due to necessary subsampling to apply the illuminant to the  $M = 31$  band hyperspectral data corresponding to the visible wavelengths (400 nm – 700 nm).

In matrix notation, the illuminant is represented by an  $M \times M$  diagonal matrix whose  $m^{\text{th}}$  element along the contains the spectral power at the  $m^{\text{th}}$  wavelength. The diagonal matrix notation simplifies the point-by-point multiplication with respect to wavelength of an illuminant with a reflectance spectrum denoted by the  $M \times 1$  vector  $\mathbf{r}$ . For a specified illuminant, defined by the diagonal matrix  $\mathbf{L}$ , the reflected radiation spectrum  $\mathbf{f}$  of a reflectance  $\mathbf{r}$  is given by

$$\mathbf{f} = \mathbf{L}\mathbf{r}. \quad (\text{A.1})$$

The spectrum  $\mathbf{f}$  must undergo two transformations to obtain the desired sRGB tristimulus vector. First,  $\mathbf{f}$  is transformed to the CIE XYZ color space via the  $M \times 3$  matrix  $\mathbf{A}$ , which defines the CIE standard observer with respect to the  $M$  wavelengths. The spectrum  $\mathbf{f}$  is mapped to the CIE XYZ color space as the  $3 \times 1$  vector  $\mathbf{t}_{XYZ}$  given by

$$\mathbf{t}_{XYZ} = \mathbf{A}^T \mathbf{f} = \mathbf{A}^T \mathbf{L}\mathbf{r}. \quad (\text{A.2})$$

Then, a transformation is used to map the CIE XYZ vector to sRGB color space. Before describing the transformation to the sRGB color space, a brief summary of the color response

of the human eye is provided as presented in [46]. This motivates the choice of the color matching functions used to define the standard observer.

The photoreceptors in the retina of the human eye belong to two main classes: rods and cones. The ratio of rods to cones in each human eye is approximately 50 to 3. The rods are very sensitive to radiant spectra, and thus rods are responsible for vision in low lighting environments. Vision in low lighting conditions is called scotopic. All of the rods in the retina have the same spectral sensitivity, and thus, the rods provide only monochromatic vision. Throughout the eye, the density of the rods in the eye far exceed the density of cones, but the density of the rods is greatest at a distance of 3 mm from the fovea. Vision is the sharpest at the fovea. This explains the need to view objects in low light “out of the corner of the eye.” Cones do not respond to the low light levels that stimulate rods, but cones have a density of roughly  $150000 \text{ mm}^{-2}$  in the central fovea. Vision in bright lighting conditions is termed photopic and is attributed to the cone receptors. The rod receptors saturate under bright lighting conditions. There are three different types of cones with peak sensitivities at wavelengths 560 nm (yellow green), 530 nm (green), and 420 nm (blue). Commonly, these are imprecisely called the respective red, green, and blue sensitivities of the eye.

Let  $\mathbf{S}$  be a  $M \times 3$  matrix whose  $i^{\text{th}}$  column defines the spectral response of the  $i^{\text{th}}$  cone of the human eye for  $M$  uniformly sampled wavelengths from approximately 400 nm to 700 nm. Similarly, define a set of primaries by the  $M \times 3$  matrix  $\mathbf{P}$  whose  $i^{\text{th}}$  column denotes the spectral content of the  $i^{\text{th}}$  lighting source. The three lighting sources are chosen such that the matrix  $\mathbf{P}$  has full column rank.

An experiment was conducted to define the color matching functions with regard to the primaries defined in  $\mathbf{P}$  [45]. The observer was presented with a monochromatic source on half of a visual field for each of the  $M$  wavelengths. The observer adjusted the relative amount of the three primaries combined in the other half of the visual field to match the monochromatic source. In some cases it was necessary to move one primary to the visual field with the monochromatic source to achieve a match. This corresponds to subtracting the primary. Let the three element vector  $\mathbf{a}_i$ , not to be confused with the contributions for finite mixture models, denote the relative intensity of the primaries. The match to a monochromatic spectra  $\mathbf{e}_i$  is mathematically such that

$$\mathbf{S}^T \mathbf{e}_i = \mathbf{S}^T \mathbf{P} \mathbf{a}_i,$$

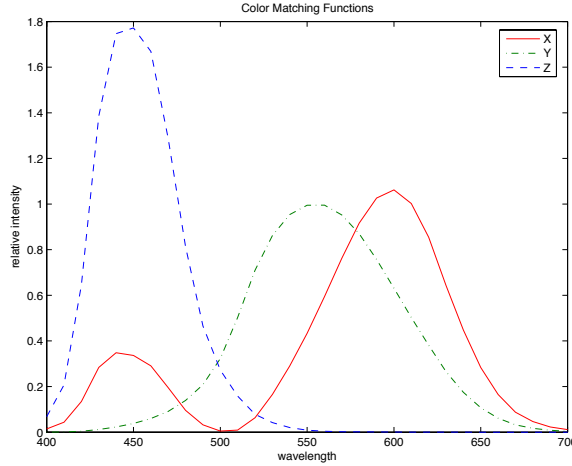


Figure A.3: Components of the CIE XYZ color matching functions

for  $i = 1, 2, \dots, M$ . Taken together, the  $M$  vectors  $\mathbf{e}_i$ , representing monochromatic sources, make up the identity matrix when ordered properly, and thus, the  $M \times 3$  matrix  $\mathbf{A}$ , formed by the  $M$  vectors  $\mathbf{a}_i$ , of color matching functions under the primaries in  $\mathbf{P}$  is defined as

$$\mathbf{A} = \mathbf{S} (\mathbf{P}^T \mathbf{S})^{-1}.$$

The primaries used to define the CIE red, green, and blue (RGB) color matching functions correspond to monochromatic spectra with wavelengths 435.8 nm, 546.1 nm, and 700 nm. The CIE XYZ color matching functions were defined in 1931 to allow for physically realizable (nonnegative) functions. The CIE XYZ set of color matching functions define the 1931 standard observer. Figure A.3 shows the X, Y, and Z components of the CIE XYZ set of color matching functions. The color matching function corresponding to the Y value is the luminous efficiency function which indicates the intensity perceived by the human eye of equally radiant monochrome sources. Thus, Eq. (A.2) defines the tristimulus values of the standard observer for the reflectance spectra  $\mathbf{r}$  viewed under the illuminant  $\mathbf{L}$ .

The transformation from the CIE XYZ color space to the sRGB color space involves two steps [47]. First, the tristimulus vector  $\mathbf{t}_{XYZ}$  is mapped to the sRGB space via the matrix  $\mathbf{T}$ , defined as

$$\mathbf{T} = \begin{bmatrix} 3.2410 & -1.5374 & -0.4986 \\ -0.9692 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0570 \end{bmatrix}. \quad (\text{A.3})$$

The vector  $\mathbf{t}_{sRGB}$  denotes the tristimulus vector for the sRGB color space and is given by

$$\mathbf{t}_{sRGB} = \mathbf{T}\mathbf{t}_{XYZ}. \quad (\text{A.4})$$

Additional gamma correction is applied to the vector  $\mathbf{t}_{sRGB}$  yielding the vector  $\mathbf{t}_{sRGB'}$ . The correction is defined as

$$[\mathbf{t}_{sRGB'}]_i = \begin{cases} 12.92 [\mathbf{t}_{sRGB}]_i & [\mathbf{t}_{sRGB}]_i \leq 0.00304 \\ 1.055 [\mathbf{t}_{sRGB}]_i^{\left(\frac{5}{12}\right)} - 0.055 & [\mathbf{t}_{sRGB}]_i > 0.00304 \end{cases}, \quad \text{for } i = 1, 2, 3. \quad (\text{A.5})$$

The vector  $\mathbf{t}_{sRGB'}$  represents the transformation of the reflectance spectrum  $\mathbf{r}$  for display with the MATLAB function `imshow`. The D65 illuminant is used to transform the reflectance spectra considered for finite mixtures of spectral densities.

## Appendix B

# Derivations for the Error-in-Variables POCS Method

This section contains two derivations for the error-in-variables POCS (EVPOCS) method described in section 4.1.1. The first derivation establishes the equivalence of the sets  $S_{TLS}$  and  $\Gamma$ . The second derivation verifies the definition of  $\tau$  in Eq. (4.15).

### B.1 Equivalence of $S_{TLS}$ and $\Gamma$ sets

The sets  $S_{TLS}$ , from Eq. (4.12), and  $\Gamma$ , from Eq. (4.13), were indicated to be equivalent according to [12]. For convenience, the set definitions have been reproduced below.

$$S_{TLS} = \left\{ \mathbf{a} \mid \exists \{\Delta\mathbf{S}, \boldsymbol{\eta}\} \ni (\bar{\mathbf{S}} + \Delta\mathbf{S}) \mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \tau \|\Delta\mathbf{S}\|_F^2 + \|\boldsymbol{\eta}\|_2^2 \leq \nu \right\} \quad (\text{B.1})$$

$$\Gamma = \left\{ \mathbf{a} \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \frac{\nu}{\tau} \|\mathbf{a}\|_2^2 - \nu \leq 0 \right\} \quad (\text{B.2})$$

In [12], a sketch of the proof that  $S_{TLS} = \Gamma$  is presented. This section describes the proof presented in [12] with greater detail.

First, suppose  $\mathbf{a} \in S_{TLS}$ . Then, there exists an  $\Delta\mathbf{S}$  such that  $\tau \|\Delta\mathbf{S}\|_F^2 + \|\bar{\mathbf{S}}\mathbf{a} -$

$\mathbf{r} + \Delta\mathbf{S}\mathbf{a}\|_2^2 - \nu \leq 0$ . This equation may be rewritten according the following steps.

$$\begin{aligned} \tau\|\Delta\mathbf{S}\|_F^2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r} + \Delta\mathbf{S}\mathbf{a}\|_2^2 - \nu &\leq 0 \\ \tau\|\Delta\mathbf{S}\|_F^2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 + 2(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T \Delta\mathbf{S}\mathbf{a} + \|\Delta\mathbf{S}\mathbf{a}\|_2^2 - \nu &\leq 0 \end{aligned} \quad (\text{B.3})$$

From the Cauchy-Bunyakovski-Schwartz (CBS) inequality where  $|\mathbf{x}^*\mathbf{y}| \leq \|\mathbf{x}\|\|\mathbf{y}\| \forall \mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ ,

$$2(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T \Delta\mathbf{S}\mathbf{a} \leq 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\Delta\mathbf{S}\mathbf{a}\|_2. \quad (\text{B.4})$$

Observed that the CBS inequality provides an upper bound. To incorporate this into Eq. (B.3) while satisfying the inequality, replace  $2(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T \Delta\mathbf{S}\mathbf{a}$  with the negative of  $2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\Delta\mathbf{S}\mathbf{a}\|_2$ . Substituting the negative of the upper bound obtained with the CBS inequality yields

$$\tau\|\Delta\mathbf{S}\|_F^2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\Delta\mathbf{S}\mathbf{a}\|_2 + \|\Delta\mathbf{S}\mathbf{a}\|_2^2 - \nu \leq 0. \quad (\text{B.5})$$

Rearrange the terms to match the equation presented in [12].

$$\tau\|\Delta\mathbf{S}\|_F^2 + \|\Delta\mathbf{S}\mathbf{a}\|_2^2 - 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\Delta\mathbf{S}\mathbf{a}\|_2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \nu \leq 0 \quad (\text{B.6})$$

The Frobenius matrix norm is compatible with the Euclidean norm, therefore

$$\|\Delta\mathbf{S}\|_F \geq \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2}{\|\mathbf{a}\|_2}, \quad (\text{B.7})$$

and in general

$$\|\Delta\mathbf{S}\|_2 = \max_{\mathbf{a} \neq 0} \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2}{\|\mathbf{a}\|_2} = \max_{\|\mathbf{a}\|_2=1} \|\Delta\mathbf{S}\mathbf{a}\|_2 = \sqrt{\lambda_{\max}} = \sigma_1, \quad (\text{B.8})$$

where  $\lambda_{\max}$  is the largest eigenvalue of  $\Delta\mathbf{S}^*\Delta\mathbf{S}$  and  $\sigma_1$  is the largest singular value of  $\Delta\mathbf{S}$ .

Define

$$\beta = \begin{cases} \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2}{\|\mathbf{a}\|_2} & \mathbf{a} \neq 0 \\ 0 & \mathbf{a} = 0 \end{cases}. \quad (\text{B.9})$$

Substitute  $\frac{\|\Delta\mathbf{S}\mathbf{a}\|_2}{\|\mathbf{a}\|_2}$  for  $\|\Delta\mathbf{S}\|_F$  in Eq. (B.6).

$$\tau \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2^2}{\|\mathbf{a}\|_2^2} + \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2^2}{\|\mathbf{a}\|_2^2} \|\mathbf{a}\|_2^2 - 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2 \frac{\|\Delta\mathbf{S}\mathbf{a}\|_2}{\|\mathbf{a}\|_2} \|\mathbf{a}\|_2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \nu \leq 0 \quad (\text{B.10})$$

Substituting  $\beta$  as defined by Eq. (B.9) yields the inequality

$$(\tau + \|\mathbf{a}\|_2^2) \beta^2 - 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2 \|\mathbf{a}\|_2 \beta + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \nu \leq 0. \quad (\text{B.11})$$

This inequality is a polynomial in  $\beta$ , which is real, and all the coefficients are real. This inequality has a nonnegative real solution satisfying Eq. (B.9) when the discriminant is nonnegative. The discriminant is given by

$$4\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|^2\|\mathbf{a}\|_2^2 - 4(\tau + \|\mathbf{a}\|_2^2)(\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \nu) \geq 0, \quad (\text{B.12})$$

and reduces to

$$\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \frac{\nu}{\tau}\|\mathbf{a}\|_2^2 - \nu \leq 0. \quad (\text{B.13})$$

Hence,  $\mathbf{a} \in \Gamma$ .

Conversely, suppose  $\mathbf{a} \in \Gamma$ . Then, there exists a  $\beta \geq 0$  satisfying Eq. (B.11). Let  $\beta_0$  be the minimum nonnegative solution of Eq. (B.11). Let

$$\Delta\mathbf{S} = \begin{cases} \frac{-\beta_0}{\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\mathbf{a}\|_2}(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})\mathbf{a}^T & \beta_0 \neq 0 \\ 0 & \beta_0 = 0 \end{cases} \quad (\text{B.14})$$

Replacing  $\beta$  with  $\beta_0$  in Eq. (B.11) and rearranging the equation leads to the following equation

$$\tau\beta_0^2 + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 + \|\mathbf{a}\|_2^2\beta_0^2 - 2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2\|\mathbf{a}\|_2\beta_0 + -\nu \leq 0. \quad (\text{B.15})$$

Observe that the fourth term may be written as

$$-2\beta_0\|\mathbf{a}\|_2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2 = -2\beta_0\frac{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2}{\|\mathbf{a}\|_2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2} = -2\beta_0\frac{(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})\mathbf{a}^T\mathbf{a}}{\|\mathbf{a}\|_2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2} \quad (\text{B.16})$$

$$= 2(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T\Delta\mathbf{S}\mathbf{a}. \quad (\text{B.17})$$

Next, observe that the third term is equivalently given by

$$\beta_0^2\|\mathbf{a}\|_2^2 = \beta_0^2\frac{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2}{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2}\|\mathbf{a}\|_2^2 = \beta_0^2\frac{\mathbf{a}^T\mathbf{a}(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})\mathbf{a}^T\mathbf{a}}{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2} \quad (\text{B.18})$$

$$= \mathbf{a}^T\Delta\mathbf{S}^T\Delta\mathbf{S}\mathbf{a} = \|\Delta\mathbf{S}\mathbf{a}\|_2^2. \quad (\text{B.19})$$

Finally, observe that the first term may be written as

$$\tau\beta_0^2 = \tau\frac{\beta_0^2}{\|\mathbf{a}\|_2^2}\frac{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2}{\|\mathbf{a}\|_2^2\|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2\|\mathbf{a}\|_2^2} = \tau\frac{\|\Delta\mathbf{S}\mathbf{a}\|_2^2}{\|\mathbf{a}\|_2^2}. \quad (\text{B.20})$$

Substituting Eqs. (B.17), (B.19), and (B.20) into Eq. (B.15) yields

$$\tau\frac{\|\Delta\mathbf{S}\mathbf{a}\|_2^2}{\|\mathbf{a}\|_2^2} + \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 + \|\Delta\mathbf{S}\mathbf{a}\|_2^2 + 2(\bar{\mathbf{S}}\mathbf{a} - \mathbf{r})^T\Delta\mathbf{S}\mathbf{a} + -\nu \leq 0. \quad (\text{B.21})$$

The term  $\tau\frac{\|\Delta\mathbf{S}\mathbf{a}\|_2^2}{\|\mathbf{a}\|_2^2}$  may be replaced by  $\tau\|\Delta\mathbf{S}\|_F^2$  if  $\nu = \tau E\|\Delta\mathbf{S}\|_F^2 + E\|\boldsymbol{\eta}\|_2^2$ . Making this substitution, Eq. (B.21) equals Eq. (B.3). Thus,  $\mathbf{a} \in S_{TLS}$ , and the theorem is proven. It is noted in [12] that the theorem holds if the Frobenius matrix norm is replaced by the spectral norm. This observation is straightforward from the relationship in Eq. (B.7).

## B.2 Derivation of $\tau$ for EVPOCS

This section presents the derivation of  $\tau$  for the noise variance set introduced by [12] and described in Section 4.1.1. Specifically, this section concludes with the definition of  $\tau$  given in Eq. (4.15). Let  $\|\cdot\|_2$  and  $\|\cdot\|_F$  denote the Euclidean and the Frobenius norms, respectively.

The TLS set is defined

$$S_{TLS} = \left\{ \mathbf{a} \mid \exists \{\Delta \mathbf{S}, \boldsymbol{\eta}\} \ni (\bar{\mathbf{S}} + \Delta \mathbf{S})\mathbf{a} = \mathbf{r} + \boldsymbol{\eta}, \quad \tau \|\Delta \mathbf{S}\|_F^2 + \|\boldsymbol{\eta}\|_2^2 \leq \nu \right\}, \quad (\text{B.22})$$

and [12] proves that the set is equally given by

$$\Gamma = \left\{ \mathbf{a} \in \mathbb{R}^N \mid \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \frac{\nu}{\tau} \|\mathbf{a}\|_2^2 - \nu \leq 0 \right\},$$

where  $\tau$  and  $\nu$  are chosen to satisfy the following

$$\nu = \tau E \|\Delta \mathbf{S}\|_F^2 + E \|\boldsymbol{\eta}\|_2^2 \quad \tau \geq \frac{E \|\boldsymbol{\eta}\|_2^2}{\sigma_K^2(\bar{\mathbf{S}}) - E \|\Delta \mathbf{S}\|_F^2},$$

where  $\sigma_K(\bar{\mathbf{S}})$  denotes the smallest singular value of  $\bar{\mathbf{S}}$ .

Beginning with Eq. (4.33) leads to

$$\begin{aligned} f(\mathbf{a}) &= \|\bar{\mathbf{S}}\mathbf{a} - \mathbf{r}\|_2^2 - \frac{\nu}{\tau} \|\mathbf{a}\|_2^2 - \nu \\ &= \mathbf{a}^T \left( \bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} \right) \mathbf{a} - 2\mathbf{r}^T \bar{\mathbf{S}}\mathbf{a} + \|\mathbf{r}\|_2^2 - \nu. \end{aligned}$$

Note that  $f(\mathbf{a})$  must be convex. Thus, the second derivative with respect to  $\mathbf{a}$  must be greater than or equal to zero. So, the first derivative is the vector

$$\frac{\partial f(\mathbf{a})}{\partial \mathbf{a}} = 2 \left( \bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} \right) \mathbf{a} - 2\bar{\mathbf{S}}^T \mathbf{r},$$

and the second derivative produces the matrix

$$\frac{\partial^2 f(\mathbf{a})}{\partial^2 \mathbf{a}} = 2 \left( \bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} \right).$$

For  $f(\mathbf{a})$  to be a convex function,  $\frac{\nu}{\tau}$  must satisfy

$$\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} \geq \mathbf{0},$$

for every element of the matrix [35]. Note that  $\bar{\mathbf{S}} \geq \mathbf{0}$  is necessary for this inequality to hold. The finite mixture model application satisfies this condition, since all elements of  $\mathbf{S}$ ,

and hence  $\bar{\mathbf{S}}$  are nonnegative. For the case of equality to zero with  $\frac{\nu}{\tau} > 0$ , then  $\frac{\nu}{\tau}$  is an eigenvalue of  $\bar{\mathbf{S}}^T \bar{\mathbf{S}}$ , which implies that  $\sqrt{\frac{\nu}{\tau}} > 0$  is a singular value of  $\bar{\mathbf{S}}$ . Assuming full rank on  $\bar{\mathbf{S}}$ , then the smallest singular value  $\sigma_K(\bar{\mathbf{S}})$  is greater than zero. Denote the singular value decomposition (SVD) of  $\bar{\mathbf{S}}$  as

$$\bar{\mathbf{S}} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T,$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are orthogonal and respectively contain left and right singular vectors. The matrix  $\mathbf{\Lambda}$  is given by

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}_K \\ \mathbf{0} \end{bmatrix}_{M \times K},$$

where  $\mathbf{\Lambda}_K$  is diagonal containing the singular values  $\sigma_k$  for  $k = 1, \dots, K$  ordered from largest to smallest. If  $\sqrt{\frac{\nu}{\tau}} > \sigma_K(\bar{\mathbf{S}})$ , then

$$\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} = \mathbf{V} \mathbf{\Lambda}^T \mathbf{U}^T \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T - \frac{\nu}{\tau} \mathbf{I} = \mathbf{V} \mathbf{\Lambda}_K^2 \mathbf{V}^T - \frac{\nu}{\tau} \mathbf{I},$$

and  $\bar{\mathbf{S}}^T \bar{\mathbf{S}} - \frac{\nu}{\tau} \mathbf{I} \geq 0$  is no longer satisfied. So, choose  $\sqrt{\frac{\nu}{\tau}} \leq \sigma_K(\bar{\mathbf{S}})$ .

Let  $\nu = \tau E \|\Delta \mathbf{S}\|_F^2 + E \|\eta\|_2^2$  and choose  $\tau > 0$  to ensure a convex function  $f(\mathbf{a})$ .

From the results above it is concluded that

$$\sqrt{\frac{\nu}{\tau}} \leq \sigma_K(\bar{\mathbf{S}}) \implies \frac{\nu}{\tau} \leq \sigma_K^2(\bar{\mathbf{S}}),$$

where  $\sigma_K(\bar{\mathbf{S}})$  is the smallest singular value of  $\bar{\mathbf{S}}$ . Substituting the definition of  $\nu$  and solving for  $\tau$  yields

$$\begin{aligned} \frac{\tau E \|\Delta \mathbf{S}\|_F^2 + E \|\eta\|_2^2}{\tau} &\leq \sigma_K^2(\bar{\mathbf{S}}) \\ \tau E \|\Delta \mathbf{S}\|_F^2 + E \|\eta\|_2^2 &\leq \tau \sigma_K^2(\bar{\mathbf{S}}) \\ \frac{E \|\eta\|_2^2}{\sigma_K^2(\bar{\mathbf{S}}) - E \|\Delta \mathbf{S}\|_F^2} &\leq \tau. \end{aligned} \tag{B.23}$$

# Bibliography

- [1] *Special Issue on Hyperspectral Imaging*, vol. 19, Jan. 2002.
- [2] C. Trivedi, H. J. Trussell, A. Nilsson, and M. Chow, “Implicit traffic classification for service differentiation,” NCSU - CACC, Tech. Rep., 2002.
- [3] B. S. Everitt and D. J. Hand, *Finite Mixture Distributions*. London: Chapman and Hall, 1981.
- [4] G. J. McLachlan and K. E. Basford, *Mixture Models: Inference and Applications to Clustering*. New York: M. Dekker, 1988.
- [5] G. J. McLachlan and D. Peel, *Finite Mixture Models*. New York: Wiley, 2000.
- [6] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of the Royal Statistical Society*, vol. 39B, no. 1, pp. 1–38, 1977.
- [7] M. A. T. Figueiredo and A. K. Jain, “Unsupervised learning of finite mixture models,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 3, pp. 381–396, March 2002.
- [8] M. Brand, “Structure learning in conditional probability models via an entropic prior and parameter extinction,” *Neural Computation*, vol. 11, pp. 1155–1182, 1999.
- [9] E. Parzen, “On estimation of a probability density function and mode,” *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.
- [10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley-Interscience, 2000.

- [11] P. L. Combettes, “The foundations of set theoretic estimation,” *Proc. IEEE*, vol. 81, pp. 182–208, 1993.
- [12] G. Sharma and H. J. Trussell, “Set theoretic signal restoration using an error in variables criterion,” *IEEE Trans. Image Processing*, vol. 6, pp. 1692–1697, 1997.
- [13] W. F. R. Weldon, “On certain correlated variations in *carcinus maenas*,” *Proceedings of the Royal Society of London*, vol. 54, pp. 318–329, 1893.
- [14] F. Galton and J. D. H. Dickson, “Family likeness in stature,” *Proceedings of the Royal Society of London*, vol. 40, pp. 42–73, 1886.
- [15] K. Pearson, “Contributions to the mathematical theory of evolution,” *Philosophical Transactions of the Royal Society of London A*, vol. 185, pp. 71–110, 1894.
- [16] G. Shaw and D. Manolakis, “Signal processing for hyperspectral image exploitation,” *IEEE Signal Processing Magazine*, no. 1, pp. 12–16, January 2002.
- [17] A. Kitamoto and M. Takagi, “Mixture density estimation under the existence of mix-els,” in *International Geoscience and Remote Sensing Symposium: quantitative remote sensing for science and applications*. IEEE, 1995.
- [18] C.-I. Chang and H. Ren, “An experiment-based quantitative and comparative analysis of target detection and image classification algorithms for hyperspectral imagery,” *IEEE Trans. Geosci. Remote Sensing*, vol. 38, pp. 1044–1063, 2000.
- [19] D. Gillis, J. Bowles, and M. E. Winter, “Using endmembers as a coordinate system in hyperspectral imagery,” presented at the SPIE’s 47th Annual Meeting, Seattle WA, 2002.
- [20] K. Hamada, M. Giffin, and C. L. Matson, “Improved spectral unmixing using a linear discriminant approach and the wavelet transform to optimize fractional abundance estimation accuracy,” in *2004 AMOS Technical Conference*, Maui, Hawaii, 13-17 September 2004.
- [21] J. C. Harsanyi and C.-I. Chang, “Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach,” *IEEE Trans. Geosci. Remote Sensing*, vol. 32, pp. 779–785, 1994.

- [22] Y. H. Hu, H. B. Lee, and F. L. Scarpace, "Optimal linear spectral unmixing," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 639–644, 1999.
- [23] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Processing Magazine*, pp. 44–57, January 2002.
- [24] J. M. P. Nascimento and J. M. B. Dias, "Does independent component analysis play a role in unmixing hyperspectral data?" *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 1, pp. 175–187, Jan. 2005.
- [25] ———, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898–910, April 2005.
- [26] P. Palmadesso, J. Antoniadis, M. Baumback, J. Bowles, and L. Rickard, "Use of filter vectors and fast convex set methods in hyperspectral analysis," in *Proceedings of the International Symposium on Spectral Sensing Research*, 1995.
- [27] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, pp. 411–430, 2000.
- [28] A. D. Stocker and A. P. Schaum, "Application of stochastic mixing models to hyperspectral detection problems," in *Algorithms for Multispectral and Hyperspectral Imagery III*, A. E. Iverson and S. S. Shen, Eds., vol. 3071. SPIE, 1997, pp. 47–60.
- [29] B. Sirkeci, D. Brady, and J. Burman, "Restricted total least squares solutions for hyperspectral imagery," in *Proc. of IEEE ICASSP 2000*, Istanbul, Turkey, 2000.
- [30] T. L. Harmon, J. Dabney, and N. Richert, *Advanced Engineering Mathematics with MATLAB*, 2nd ed. Brooks/Cole, 2000.
- [31] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.
- [32] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.
- [33] S. van Huffel and J. Vandewalle, *The Total Least Squares Problem*. Philadelphia, PA: SIAM, 1991.

- [34] G. H. Golub and C. F. V. Loan, "An analysis of the total least squares problem," *SIAM Journal on Numerical Analysis*, vol. 17, no. 6, pp. 883–893, 1980.
- [35] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Philadelphia: SIAM, 2000.
- [36] P. L. Combettes and H. J. Trussell, "Method of successive projections for finding a common point of sets in metric spaces," *J. of Opt. Theory and App.*, vol. 67, no. 3, pp. 487–507, 1990.
- [37] S. Kaczmarz, "Angenäherte auflösung von systemen linearer gleichungen," *Bulletin de l'Académie des Sciences de Pologne*, vol. A35, pp. 355–357, 1937.
- [38] G. Cimmino, "Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari," *La Ricerca Scientifica (Roma)*, vol. 1, pp. 326–333, 1938.
- [39] P. L. Combettes, "Convex set theoretic image recovery by extrapolated iterations of parallel subgradient projections," *IEEE Trans. Image Processing*, vol. 6, no. 1, pp. 493–506, 1997.
- [40] H. J. Trussell, "Applications of set theoretic methods to color systems," *Color Research and Applications*, vol. 16, no. 1, pp. 31–41, February 1991.
- [41] H. J. Trussell and M. R. Civanlar, "The feasible solution in signal restoration," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 201–212, 1984.
- [42] P. L. Combettes and H. J. Trussell, "Methods for digital restoration of signals degraded by a stochastic impulse response," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 393–401, 1989.
- [43] P. L. Combettes, "Models and algorithms for the digital restoration of stochastically degraded images," Master's thesis, North Carolina State University, 1987.
- [44] M. Vrhel, R. Gershon, and L. Iwan, "Measurement and analysis of object reflectance spectra," *Color Research and Applications*, vol. 19, no. 1, pp. 4–9, 1994.
- [45] H. J. Trussell, "Notes on fundamentals of digital image processing," 2004, unpublished.
- [46] H. B. Barlow, "Physiology of the retina," in *The Senses*, H. B. Barlow and J. D. Mollon, Eds. Cambridge, UK: Cambridge University Press, 1982, ch. 6.

- [47] M. Stokes, M. Anderson, S. Chandrasekar, and R. Motta. (1996, November) A standard default color space for the internet - srgb. [Online]. Available: [www.color.org/sRGB.html](http://www.color.org/sRGB.html)