Chapter 2
Linear Spectral Mixture Analysis

Abstract Linear Spectral Mixture Analysis (LSMA) is a widely used theory in hyperspectral data exploration. It first assumes that for a given finite set of basic material substances a data sample can be modeled as a linear admixture of these substances from which the data sample can be unmixed into their corresponding abundance fractions. In this case, analysis of the data sample can simply be performed on these abundance fractions rather than the sample itself. The commonly used Linear Spectral Unmixing (LSU) is such a technique that realizes LSMA. In other words, it assumes that a data sample is linearly mixed by a finite set of so-called image endmembers. It then unmixes this data sample via finding its abundance fractions of these endmembers for data analysis. So, technically speaking, LSU is a means of implementing LSMA as a data unmixing technique.

2.1 Introduction

Linear Mixture Analysis (LMA) has found numerous applications in data analysis. It describes a data sample in a linear model which can be formed and parameterized and determined by a finite set of basic constituents. By means of this linear model the data sample can be either restored or represented by finding the weighting coefficients on these constituents via solving a linear inverse problem. As a simple illustrative example, consider a real value a between 0 and 1. Suppose that we are only allowed to use 8 bits to restore this value. The first eight binary coefficients of a represented by 2^{-j} with $1 \le j \le 8$, $a_1, a_2, \ldots, a_8$, are the best way to restore the value of a. In other words, instead of representing the true value of a we make use of an eight-dimensional vector $\boldsymbol{\alpha} = (a_1, a_2, \ldots, a_8)$ to represent $a = \boldsymbol{\alpha}^{\mathrm{T}}\mathbf{m}_8$ where $\mathbf{m}_8 = (2^{-1}, 2^{-2}, \ldots, 2^{-8})^{\mathrm{T}}$ is an eight-dimensional vector and $\{2^{-j}\}_{j=1}^{8}$ in $\mathbf{m}_8$ are the eight basic constituents with $\{a_j\}_{j=1}^{8}$ being their corresponding weighting coefficients which can be considered as abundance fractions of $\{2^{-j}\}_{j=1}^{8}$ to represent

C.-I Chang, *Real-Time Progressive Hyperspectral Image Processing*,
DOI 10.1007/978-1-4419-6187-7_2

best the value a in the sense that the value of a is best linearly mixed by the eight basic constituents $\{2^{-j}\}_{j=1}^{8}$ with weighting coefficients $\{a_j\}_{j=1}^{8}$.

Depending upon what constituents are used in the linear model, various LMA can be developed. For example, when these constituents are training data the linear regression model can be viewed as LMA. On the other hand, if these constituents are feature vectors, LMA can be interpreted as a linear discriminant model such as Fisher's Linear Discriminant Analysis (FLDA). Furthermore, if these constituents are specified by eigenvectors and independent projection vectors, the resulting LMA is then called Principal Components Analysis (PCA) and Independent Component Analysis (ICA), respectively. In particular, when LMA is applied to spectral data such as remotely sensed data acquired by a range of wavelengths, these constituents are generally represented by so-called endmembers. As a result, LMA becomes the well-known LSMA which is the main focus in this chapter.

Suppose that L is the number of spectral bands and $\mathbf{r}$ is an L-dimensional image pixel vector. Assume that there are p material substances, $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_p$, present in an image scene. Let $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p$ denote their corresponding substance signatures, generally referred to as digital numbers (DN) which are digital representations of quantized radiance values measured by sensors. Such radiances generally correspond to mixed spectral signatures rather than to "pure material substances signatures." A linear mixture model of $\mathbf{r}$ models the spectral signature of $\mathbf{r}$ as a linear combination of $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p$ with appropriate abundance fractions specified by $\alpha_1, \alpha_2, \ldots, \alpha_p$. More precisely, $\mathbf{r}$ is an $L \times 1$ column vector and $\mathbf{M}$ an $L \times p$ substance spectral signature matrix, denoted by $\left[\mathbf{m}_1\ \mathbf{m}_2, \ \ldots\ , \mathbf{m}_p\right]$, where $\mathbf{m}_j$ is an $L \times 1$ column vector represented by the spectral signature of the jth substance $\mathbf{t}_j$ resident in the pixel vector $\mathbf{r}$. Let $\boldsymbol{\alpha} = \left(\alpha_1, \alpha_2, \ldots, \alpha_p\right)^{\mathrm{T}}$ be a $p \times 1$ abundance column vector associated with $\mathbf{r}$ where α_j denotes the abundance fraction of the jth substance signature $\mathbf{m}_j$ present in the pixel vector $\mathbf{r}$. To restore the pixel vector $\mathbf{r}$, we assume that the spectral signature of the pixel vector $\mathbf{r}$ is linearly mixed by $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p$, the spectral signatures of the p substances, $\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_p$ being as follows:

$$\mathbf{r} = \mathbf{M}\alpha + \mathbf{n} \tag{2.1}$$

where $\mathbf{n}$ is noise or can be interpreted as a measurement or model error.

By inspecting (2.1), the linear model used by LSMA is similar to a linear model used by Wiener filter except that (2.1) explores correlation among p substance signatures compared to the latter which uses a linear model to account for the p past observed data samples to perform prediction. Furthermore, by means of (2.1), a hyperspectral image viewed as an image cube can be restored by a set of p abundance fraction maps.

2.2 Solving LSMA Problems

In general there are two approaches to solving (2.1), least squares error (LSE)-based estimation and signal to noise ratio (SNR)-based detection, each of which is described in the following two subsections.

2.2.1 Least Squares Error (LSE)

The LSE problem resulting from (2.1) can be formed by

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.2}$$

A classical approach to solving (2.2) is given by

$$\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})=\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r} \tag{2.3}$$

where $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})=\left(\hat{\alpha}_1^{\mathrm{LS}}(\mathbf{r}),\hat{\alpha}_2^{\mathrm{LS}}(\mathbf{r}),\ldots,\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})\right)$ and $\hat{\alpha}_j^{\mathrm{LS}}(\mathbf{r})$ is the abundance fraction of the jth substance signature $\mathbf{m}_j$ estimated from the data sample vector $\mathbf{r}$. Here the data sample vector is included to emphasize that the abundance estimate is determined by $\mathbf{r}$.

2.2.2 Signal to Noise Ratio (SNR)

With a completely different approach, Harsanyi and Chang (1994) developed an SNR approach to solving (2.1). It represents (2.1) as a standard signal detection model where $\mathbf{M}\boldsymbol{\alpha}$ is a desired signal vector to be detected and $\mathbf{n}$ is a corrupted noise. Since we are interested in detecting one substance at a time, we can divide the set of the p substances, $\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_p$, into a desired substance, say $\mathbf{t}_p$, and a class of undesired substances, $\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_{p-1}$. In this case, a logical approach is to eliminate the effects caused by the undesired substances $\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_{p-1}$ which are considered as interferers to $\mathbf{t}_p$ before the detection of $\mathbf{t}_p$ takes place. With annihilation of the undesired substance signatures, the detectability of $\mathbf{t}_p$ can be enhanced. In doing so, we first separate $\mathbf{m}_p$ from $\mathbf{m}_1,\mathbf{m}_2,\ldots,\mathbf{m}_p$ in $\mathbf{M}$ and rewrite (2.1) as the so-called OSP model, $\mathbf{r}=\alpha_p\mathbf{d}+\gamma\mathbf{U}+\mathbf{n}$, where $\mathbf{d}=\mathbf{m}_p$ is the desired spectral signature of $\mathbf{t}_p$ and $\mathbf{U}=\left[\mathbf{m}_1\mathbf{m}_2,\ldots,\mathbf{m}_{p-1}\right]$ is the undesired substance spectral signature matrix made up of $\mathbf{m}_1,\mathbf{m}_2,\ldots,\mathbf{m}_{p-1}$ which are the spectral signatures of the remaining $p-1$ undesired substances, $\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_{p-1}$. Here, without loss of generality, we assume that the desired substance is a single substance $\mathbf{t}_p$. According to the

derivation in Harsanyi and Chang (1994) also provided in Chang (2003, 2013) the solution can be obtained from

$$\hat{\alpha}_p^{\mathrm{OSP}}(\mathbf{r}) = \mathbf{d}^{\mathrm{T}} P_{\mathbf{U}}^{\perp} \mathbf{r} \tag{2.4}$$

where

$$P_{\mathbf{U}}^{\perp} = \mathbf{I} - \mathbf{U}\mathbf{U}^{\#} = \mathbf{I} - \mathbf{U}\left(\mathbf{U}^{\mathrm{T}}\mathbf{U}\right)^{-1}\mathbf{U}^{\mathrm{T}} \tag{2.5}$$

and $\mathbf{U}^{\#}$ is the pseudo inverse of $\mathbf{U}$ given by $\left(\mathbf{U}^{\mathrm{T}}\mathbf{U}\right)^{-1}\mathbf{U}^{\mathrm{T}}$. It has been shown in Settle (1996), Tu et al. (1997), and Chang et al. (1998a, b) that

$$\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r}) = \hat{\alpha}_p^{\mathrm{LSOSP}}(\mathbf{r}) = \left(\mathbf{d}^{\mathrm{T}} P_{\mathbf{U}}^{\perp} \mathbf{d}\right)^{-1} \hat{\alpha}_p^{\mathrm{OSP}}(\mathbf{r}). \tag{2.6}$$

According to (2.4), $\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})$ and $\hat{\alpha}_p^{\mathrm{OSP}}(\mathbf{r})$ only differ by a constant $\left(\mathbf{d}^{\mathrm{T}} P_{\mathbf{U}}^{\perp} \mathbf{d}\right)^{-1}$ but the implication is quite different because $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ is an estimator compared to $\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})$ which is a detector. In addition, $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ estimates all abundance fractions of p substance signatures, $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p$ all together at the same time, while $\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})$ detects the presence of the substance signature $\mathbf{m}_p$ via its detected abundance fraction, in which case it can be performed one substance signature at a time.

To restore faithfully the spectral information contained in data sample vectors via LSMA, three issues need to be addressed. The first is determination of the value of p, i.e., number of substance signatures. The second is to find a desired set of p substance signatures, $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_p$, in an unsupervised manner. The third is to impose two physical constraints on model (2.1), which are abundance sum-to-one constraint (ASC), $\sum_{j=1}^{p} \alpha_j = 1$ and abundance non-negativity constraint (ANC), $\alpha_j \geq 0$ for all $1 \leq j \leq p$. In what follows we will address these three issues and provide solutions for each of them.

2.3 Abundance-Constrained LSMA

For mathematical tractability, the solutions given by (2.3) and (2.4) are abundance-unconstrained in the sense that no physical constraints are imposed on the model (2.1). However, to restore data integrity to satisfy reality, abundance constraints must be imposed on the substance signatures in (2.1) by either ASC or ANC or both on the abundance vector $\boldsymbol{\alpha}$ in (2.1). In this section, three abundance-constrained LSE problems can be derived from (2.1), referred to as abundance-constrained LSMA (AC-LSMA), and formulated as outlined below.

2.3.1 Abundance Sum-to-One Constrained LSMA

A simple way to impose a constraint on (2.1) is ASC on the solution $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$. In other words, a sum-to-one constrained least squares (SCLS) problem can be cast by

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \quad \text{subject to} \sum_{j=1}^{p}\alpha_j = 1. \tag{2.7}$$

To solve (2.7) we use the Lagrange multiplier λ_1 to constrain the ASC, $\mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha} = 1$ by

$$J(\boldsymbol{\alpha},\lambda_1) = \frac{1}{2}(\mathbf{M}\boldsymbol{\alpha}-\mathbf{r})^{\mathrm{T}}(\mathbf{M}\boldsymbol{\alpha}-\mathbf{r}) + \lambda_1(\mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha}-1) \tag{2.8}$$

where $\mathbf{1}$ is the unity vector given by $\mathbf{1} = \underbrace{(1,1,\ldots,1)}_{p}^{\mathrm{T}}$. Differentiating (2.8) with respect to $\boldsymbol{\alpha}$ and λ_1 yields

$$\begin{aligned}
\left.\frac{\partial J(\boldsymbol{\alpha},\lambda)}{\partial\boldsymbol{\alpha}}\right|_{\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r})} &= \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r}) - \mathbf{M}^{\mathrm{T}}\mathbf{r} + \lambda_1^*\mathbf{1} = 0 \\
\Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r}) &= \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r} - \lambda_1^*\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1} \\
\Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r}) &= \hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) - \lambda_1^*\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}
\end{aligned} \tag{2.9}$$

and

$$\left.\frac{\partial J(\boldsymbol{\alpha},\lambda_1)}{\partial\lambda_1}\right|_{\lambda_1^*} = \mathbf{1}^{\mathrm{T}}\hat{\alpha}^{\mathrm{LS}}(\mathbf{r}) - \mathbf{1} = 0 \tag{2.10}$$

It should be noted that (2.9) and (2.10) must be solved simultaneously in such a way that both the optimal solutions, $\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r})$ and λ_1^*, appear in (2.8). Using (2.10) further implies that

$$\begin{aligned}
&\mathbf{1}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) - \lambda_1^*\mathbf{1}^{\mathrm{T}}\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1} = 1 \\
&\Rightarrow \lambda_1^* = -\left[\mathbf{1}^{\mathrm{T}}\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\right]^{-1}\left(1-\mathbf{1}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})\right)
\end{aligned} \tag{2.11}$$

Substituting (2.11) into (2.10) yields

$$\begin{aligned}
\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r}) &= \hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) + \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\left[\mathbf{1}^{\mathrm{T}}(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{1}\right]^{-1}\left(1-\mathbf{1}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})\right) \\
&= P_{\mathbf{M},\mathbf{1}}^{\perp}\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) + \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\left[\mathbf{1}^{\mathrm{T}}(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{1}\right]^{-1}
\end{aligned} \tag{2.12}$$

where

$$P_{\mathbf{M},\mathbf{1}}^{\perp} = \mathbf{I} - \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\left[\mathbf{1}^{\mathrm{T}}(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{1}\right]^{-1}\mathbf{1}^{\mathrm{T}} \tag{2.13}$$

2.3.2 Abundance Non-negativity Constrained LSMA

Because there is no physical constraint imposed on the solution $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ their generated abundance fractions $\hat{\alpha}_1^{\mathrm{LS}}(\mathbf{r}), \hat{\alpha}_2^{\mathrm{LS}}(\mathbf{r}), \ldots, \hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})$ may be negative. To avoid this situation the ANC must be imposed on (2.1). A general approach to solving such an ANC-constrained problem is to introduce an LSE-based objective function imposed by the ANC for $j \in \{1, 2, \ldots, p\}$ defined as

$$J = \frac{1}{2}(\mathbf{M}\boldsymbol{\alpha} - \mathbf{r})^{\mathrm{T}}(\mathbf{M}\boldsymbol{\alpha} - \mathbf{r}) + \boldsymbol{\lambda}(\boldsymbol{\alpha} - \mathbf{c}) \tag{2.14}$$

where $\boldsymbol{\lambda} = \left(\lambda_1, \lambda_2, \ldots, \lambda_p\right)^{\mathrm{T}}$ is a Lagrange multiplier vector and a constraint vector $\mathbf{c} = \left(c_1, c_2, \ldots, c_p\right)^{\mathrm{T}}$ with $c_j > 0$ for $1 \le j \le p$. In analogy with (2.10) we obtain

$$\begin{aligned} \left.\frac{\partial J(\boldsymbol{\alpha}, \lambda)}{\partial \boldsymbol{\alpha}}\right|_{\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r})} &= \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) - \mathbf{M}^{\mathrm{T}}\mathbf{r} + \boldsymbol{\lambda} = 0 \\ \Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) &= \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r} - \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\boldsymbol{\lambda} \\ \Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) &= \hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) - \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\boldsymbol{\lambda} \end{aligned} \tag{2.15}$$

which implies

$$\begin{aligned} \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\boldsymbol{\lambda}^{*} &= \hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) - \widehat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) \\ \Rightarrow \boldsymbol{\lambda}^{*} &= (\mathbf{M}^{\mathrm{T}}\mathbf{M})\widehat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r}) - (\mathbf{M}^{\mathrm{T}}\mathbf{M})\widehat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) \\ \Rightarrow \boldsymbol{\lambda}^{*} &= \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)\left[\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r}\right] - (\mathbf{M}^{\mathrm{T}}\mathbf{M})\widehat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) \\ \Rightarrow \boldsymbol{\lambda}^{*} &= \mathbf{M}^{\mathrm{T}}\mathbf{r} - \mathbf{M}^{\mathrm{T}}\mathbf{M}\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r}) \end{aligned} \tag{2.16}$$

For $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ to satisfy ANC further, the following Kuhn–Tucker conditions must be implemented:

$$\begin{aligned} \lambda_i &= 0, \quad i \in P \\ \lambda_i &< 0, \quad i \in R \end{aligned} \tag{2.17}$$

where P and R represent passive and active sets that contain indices representing negative or positive abundances, respectively. By means of (2.15)–(2.17), a

numerical algorithm, referred to as NCLS, can be designed to start off with the initial estimate given by $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ in (2.3). If all abundance fractions are positive, the NCLS stops. Otherwise, all indices corresponding to negative and zero abundance fractions are moved to passive set P and all positive abundance indices are moved to R. According to Kuhn–Tucker conditions (2.17), any λ_i with index $i \in P$ is set to zero, and other indices are calculated based on (2.16). If all λ_is are negative, the NCLS stops. If not, the corresponding most negative index is moved from R to P. A new vector $\boldsymbol{\lambda}$ is then recalculated based on the modified index sets and a new Lagrange multiplier vector is further implemented to find a new set of abundance fractions. By comparing with index set S, any negative abundance indices are shuffled from P to R. By iteratively implementing (2.15) and (2.16) using $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ in (2.3) as an initial abundance vector, the NCLS can be derived to find an optimal solution, $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS}}(\mathbf{r})$ (Chang and Heinz 2000). A detailed step-by-step algorithmic implementation of NCLS is given in the following.

NCLS Algorithm

1. Initialization: Set the passive set $P^{(0)} = \{1, 2, \ldots, p\}$ and $R^{(0)} = \emptyset$ and $k = 0$.
2. Compute $\hat{\boldsymbol{\alpha}}^{LS}(\mathbf{r})$ by (2.3) and let $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k)}(\mathbf{r}) = \boldsymbol{\alpha}^{\mathrm{LS}}(\mathbf{r})$.
3. At the kth iteration, if all components in $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k)}(\mathbf{r})$ are non-negative, the algorithm is terminated. Otherwise, continue.
4. Let $k = k + 1$.
5. Move all indices in $P^{(k-1)}$ that correspond to negative components of $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k-1)}(\mathbf{r})$ to $R^{(k-1)}$, respectively. Create a new index $S^{(k)}$ and set it equal to $R^{(k)}$.
6. Let $\hat{\boldsymbol{\alpha}}_{R^{(k)}}$ denote the vector consisting of all components of $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k-1)}(\mathbf{r})$ in $R^{(k)}$.
7. Form a matrix $\boldsymbol{\Phi}_{\boldsymbol{\alpha}}^{(k)}$ by deleting all rows and columns in the matrix $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}$ that are specified by $P^{(k)}$.
8. Calculate $\boldsymbol{\lambda}^{(k)} = \left(\boldsymbol{\Phi}_{\boldsymbol{\alpha}}^{(k)}\right)^{-1}\hat{\boldsymbol{\alpha}}_{R^{(k)}}$. If all components in $\lambda^{(k)}$ are negative, go to step 13. Otherwise, continue.
9. Calculate $\lambda_{\max}^{(k)} = \arg\left\{\max_j \lambda_j^{(k)}\right\}$ and move the index in $R^{(k)}$ that corresponds to $\lambda_{\max}^{(k)}$ to $P^{(k)}$.
10. Form another matrix $\boldsymbol{\Psi}_{\lambda}^{(k)}$ by deleting every column of $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}$ specified by $P^{(k)}$.
11. Set $\hat{\boldsymbol{\alpha}}_{S^{(k)}} = \hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k)} - \boldsymbol{\Psi}_{\lambda}^{(k)}\boldsymbol{\lambda}^{(k)}$.
12. If any components of $\hat{\boldsymbol{\alpha}}_{S^{(k)}}$ in $S^{(k)}$ are negative, then move these components from $P^{(k)}$ to $R^{(k)}$. Go to step 6.
13. Form another matrix $\boldsymbol{\Psi}_{\lambda}^{(k)}$ by deleting every column of $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}$ specified by $P^{(k)}$.
14. Set $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},(k)} = \hat{\boldsymbol{\alpha}}^{\mathrm{LS}} - \boldsymbol{\Psi}_{\lambda}^{(k)}\boldsymbol{\lambda}^{(k)}$. Go to step 3.

2.3.3 Abundance Fully Constrained LSMA

Because NCLS does not impose the ASC, its generated abundance fractions do not necessarily add up to one. It must solve the following constrained optimization problem:

$$\min_{\boldsymbol{\alpha}\in\Delta}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \text{ subject to } \Delta=\left\{\alpha|\alpha_j\geq 0 \text{ for } \forall j, \sum_{j=1}^{p}\alpha_j=1\right\}. \quad (2.18)$$

The optimal solution to (2.18) first takes advantage of the LS solution, $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$, as an initial estimate to derive

$$\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS}}(\mathbf{r})=P^{\perp}_{\mathbf{M},\mathbf{1}}\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})+(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{1}[\mathbf{1}^{\mathrm{T}}(\mathbf{M}^{\mathrm{T}}\mathbf{M})\mathbf{1}]^{-1} \quad (2.19)$$

and

$$P^{\perp}_{\mathbf{M},\mathbf{1}}=\mathbf{I}_{L\times L}-(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{1}[\mathbf{1}^{\mathrm{T}}(\mathbf{M}^{\mathrm{T}}\mathbf{M})\mathbf{1}]^{-1}\mathbf{1}^{\mathrm{T}}. \quad (2.20)$$

It then uses SCLS together with ANC by introducing a new signature matrix $\mathbf{N}$ and an observation vector $\mathbf{s}$ into the NCLS specified by

$$\mathbf{N}=\begin{bmatrix}\eta\mathbf{M}\\ \mathbf{1}^{\mathrm{T}}\end{bmatrix} \text{ and } \mathbf{s}=\begin{bmatrix}\eta\mathbf{r}\\ 1\end{bmatrix} \quad (2.21)$$

where η is a parameter to control the effect of ASC on the NCLS and defined as the reciprocal of the maximal element in the matrix $\mathbf{M}=[m_{ij}]$, i.e., $\eta=1/\max_{ij}\{m_{ij}\}$. The utilization of η in (2.21) controls the impact of ASC. Using (2.21), an FCLS algorithm can be derived directly from the NCLS algorithm described in the previous section by replacing $\mathbf{M}$, $\mathbf{r}$, and $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$ used in the NCLS algorithm with $\mathbf{N}$, $\mathbf{s}$, and $\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}(\mathbf{r})$.

2.3.4 Modified FCLS

For mathematical tractability, the solutions given by (2.3) and (2.4) are abundance-unconstrained in the sense that no physical constraints are imposed on the model (2.1). Because there is no physical constraint imposed on the solution $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})$, their generated abundance fractions $\hat{\alpha}_1^{\mathrm{LS}}(\mathbf{r}),\hat{\alpha}_2^{\mathrm{LS}}(\mathbf{r}),\ldots,\hat{\alpha}_p^{\mathrm{LS}}(\mathbf{r})$ may be negative. The main difficulty with solving abundance non-negativity problems is that the Lagrange multiplier method cannot be used to find solutions analytically. In this section, we propose an alternative approach to modify this constraint. Instead of directly dealing with the inequalities, $\alpha_j\geq 0$ for each $1\leq j\leq p$ as abundance

non-negativity-constrained least squares (NCLS) does numerically in Sect. 2.3.2, here we replace ANC with an absolute abundance sum-to-one constraint (AASC), $\sum_{j=1}^{p} |\alpha_j| = 1$, an idea which was originally envisioned in Ren and Chang (1999) and has been overlooked in the past. The advantage of AASC is that the Lagrange multiplier method is now applicable and can be used to derive an analytical solution that leads to a desired optimal non-negativity constrained least squares (NCLS) solution. With this in mind, a modified LS linear mixing problem using AASC, $\sum_{j=1}^{p} |\alpha_j| = 1$ can be cast as follows:

$$\min_{\boldsymbol{\alpha}\in\Delta}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \tag{2.22}$$

subject to

$$\boldsymbol{\Delta} = \left\{\boldsymbol{\alpha} \middle| \sum_{j=1}^{p} |\alpha_j| = 1\right\} \tag{2.23}$$

As mentioned previously, the main difficulty with solving constrained linear mixing problems is the constraint of abundance non-negativity that prohibits us from using the Lagrange multiplier method to find solutions analytically. In this section, we propose an alternative approach to modify this constraint. Instead of directly dealing with ANC, which imposes the inequalities, $\alpha_j \geq 0$ for each $1 \leq j \leq p$, we replace ANC with an absolute AASC, $\sum_{j=1}^{p} |\alpha_j| = 1$. The advantage of AASC is that the Lagrange multiplier method is now applicable and can be used to derive an iterative algorithm that leads to a desired optimal constrained LS solution. Additionally, AASC also allows us to exclude negative abundances from solutions. In other words, the only possibility to satisfy both constraints, sum-to-one ($\sum_{j=1}^{p} \alpha_j = 1$) and AASC ($\sum_{j=1}^{p} |\alpha_j| = 1$) is that all the abundances $\{\alpha_j\}_{j=1}^{p}$ must satisfy both ASC and ANC. So, a modified LS linear mixing problem with constraints, ASC, $\sum_{j=1}^{p} \alpha_j = 1$ and AASC, $\sum_{j=1}^{p} |\alpha_j| = 1$ can be cast as follows:

$$\min_{\boldsymbol{\alpha}\in\Delta}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \tag{2.24}$$

subject to

$$\Delta = \left\{\boldsymbol{\alpha} \middle| \sum_{j+1}^{p} \alpha_j = 1 \text{ and } \sum_{j=1}^{p} |\alpha_j| = 1\right\} \tag{2.25}$$

Using a similar argument that derived the SCLS solution, we write

$$J(\boldsymbol{\alpha}) = (1/2)(\mathbf{r} - \mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\mathbf{r} - \mathbf{M}\boldsymbol{\alpha}) - \lambda_1\left(\sum_{j=1}^{p}\alpha_j - 1\right) - \lambda_2\left(\sum_{j=1}^{p}|\alpha_j| - 1\right) \tag{2.26}$$

Differentiating (2.26) with respect to $\boldsymbol{\alpha}$ and setting to zero yields

$$\begin{aligned} &\left.\frac{\partial J(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}}\right|_{\hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}}} = 0 \\ &\Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}} = \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\left[\mathbf{M}^{\mathrm{T}}\mathbf{r} - \lambda_1\mathbf{1} - \lambda_2\mathrm{sign}(\boldsymbol{\alpha})\right] \\ &\Rightarrow \hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}} = \boldsymbol{\alpha}^{\mathrm{LS}} - \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\left[\lambda_1\mathbf{1} + \lambda_2\mathrm{sign}(\boldsymbol{\alpha}^{\mathrm{LS}})\right] \end{aligned} \tag{2.27}$$

where $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}} = \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r}$ is the unconstrained LS estimate of $\boldsymbol{\alpha}$ given by (2.3). We then substitute $\hat{\boldsymbol{\alpha}}^{\mathrm{LS}}$ for $\boldsymbol{\alpha}$ in the following constraints:

$$\sum_{j=1}^{p}\alpha_j = \mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha} = 1 \tag{2.28}$$

$$\sum_{j=1}^{p}|\alpha_j| = \mathrm{sign}(\boldsymbol{\alpha})^{\mathrm{T}}\boldsymbol{\alpha} = 1 \tag{2.29}$$

to compute λ_1 and λ_2 where $\mathrm{sign}(\boldsymbol{\alpha})$ is a vector given by the sign function of $\boldsymbol{\alpha}$ with the jth component being the sign of α_j, namely, $\mathrm{sign}(\boldsymbol{\alpha}) = \left(\beta_1, \beta_2, \ldots, \beta_p\right)^{\mathrm{T}}$ where β_j is defined by

$$\beta_j = \frac{\alpha_j}{|\alpha_j|};\ \text{if}\ \alpha_j \neq 0\,\text{and}\,0;\ \ \text{if}\ \ \alpha_j = 0 \tag{2.30}$$

The solutions to (2.25) and (2.26) can then be obtained by iteratively computing λ_1, λ_2, and $\hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}}$ using (2.28)–(2.30). The details of implementing this algorithm are given below.

MFCLS Iterative Algorithm

1. Initialization: Set $\hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}} = \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}$ using (2.12)
2. Compute λ_1, λ_2 using (2.27)–(2.29)
3. Compute $\hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}} = \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}} - \left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}[\lambda_1\mathbf{1} + \lambda_2\mathrm{sign}(\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}})]$
4. If there exists one component in $\hat{\boldsymbol{\alpha}}^{\mathrm{MFCLS}}$, which is negative, go to step 2. Otherwise, stop.

It should be noted that in step 1 the algorithm is initialized by taking advantage of the SCLS solution $\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}$ obtained by (2.12). The stopping criterion described in step 4 provides a rule of when the algorithm will be terminated, in which case all components must be non-negative. In general, it requires a fair amount of computation time to reach this requirement. So, in real implementation, we suggest a simpler rule by checking $\sum_{j=1}^{p} |\alpha_j| - 1 < \varepsilon$ for a preselected threshold ε which will guarantee a quick termination.

The main difference between FCLS and MFCLS is their initial constraint condition. FCLS are constrained with ASC and non-negativity constraint by NCLS. Hence, FCLS can be constructed by modifying the desired signature and signature matrix by (2.21) and then by using NCLS algorithm to obtain FCLS results. On the other hand, MFCLS uses the absolute value for the constraint of non-negativity. As a result, the algorithm can be seen as follows:

MFCLS Algorithm

1. Initialization: Set $\mathbf{N} = \begin{bmatrix} \eta\mathbf{M} \\ \mathbf{1}^{\mathrm{T}} \end{bmatrix}$ and $\mathbf{s} = \begin{bmatrix} \eta\mathbf{r} \\ 1 \end{bmatrix}$, where $\delta \approx 1/\max(\max(\mathbf{M}))$.
2. Compute $\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}$ using (2.12) with $\mathbf{N}$ and $\mathbf{s}$. Set $\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS},(0)} = \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}$.
3. At the kth iteration. If all components in $\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS},(k)}$ are positive, the algorithm is terminated. Otherwise, continue.
4. Let $k = k + 1$.
5. Move all indices in $P^{(k-1)}$ that correspond to negative components of $\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS},(k-1)}$ to $R^{(k-1)}$ and the resulting index sets are denoted by $P^{(k)}$ and $R^{(k)}$, respectively.
6. Let $\hat{\boldsymbol{\alpha}}_{R^{(k)}}$ denote the vector consisting of all components $\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}}$ in $R^{(k)}$.
7. Form a steering matrix $\Phi_{\alpha}^{(k)}$ by deleting all rows and columns in the matrix $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}$ that are specified by $P^{(k)}$. Then append $\left(\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}_{R^{(k)}}\right)^{\mathrm{T}}\mathbf{1}$ after the last column, $\left(\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\right)_{R^{(k)}}$ after the last row, and $\mathbf{1}^{\mathrm{T}}\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}$ to the left-most corner to $\Phi_{\alpha}^{(k)}$.
8. Append a zero to the last row of $\hat{\boldsymbol{\alpha}}_{R(k)}$. Calculate $\boldsymbol{\lambda}^{(k)} = \left(\boldsymbol{\Phi}_{\alpha}^{(k)}\right)^{-1}\hat{\boldsymbol{\alpha}}_{R(k)}$.
9. If all components suggest the last row in $\lambda^{(k)}$ is negative, go to step 11. Otherwise, continue.
10. Find $\lambda_{\max}^{(k)} = \max_j \lambda_j^{(k)}$ and move the index in $R^{(k)}$ that corresponds to $\boldsymbol{\lambda}_{\max}^{(k)}$ to $P^{(k)}$. Go to step 7.
11. Form another matrix $\boldsymbol{\Psi}_{\lambda}^{(k)}$ by deleting every column of $\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}$ specified by $P^{(k)}$.
12. Let $\lambda_1^{(k)} = \boldsymbol{\lambda}^{(k)}(l)$ and $\boldsymbol{\lambda}_2^{(k)} = \boldsymbol{\lambda}^{(k)}[1 : l-1]$, where l equals the last row of $\lambda^{(k)}$.
13. Set $\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS},(k)} = \hat{\boldsymbol{\alpha}}^{\mathrm{SCLS}} - \left(\left(\mathbf{M}^{\mathrm{T}}\mathbf{M}\right)^{-1}\mathbf{1}\right)\lambda_1^{(k)}\boldsymbol{\Psi}_{\lambda}^{(k)}\boldsymbol{\lambda}_2^{(k)}$. Go to step 3.

2.4 Weighted LSMA

According to (2.2), the LSE is equally weighted for all bands which are assumed to have their uniform effects on LSE. In general, this may not be necessarily true. To generalize this concept, we consider a weighted LSE approach to (2.2) by introducing a weighting matrix $\mathbf{A}$ into (2.2) so that the LSE is weighted by $\mathbf{A}$ via

$$(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\mathbf{A}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha}). \tag{2.31}$$

which is reduced to (2.2) if $\mathbf{A}=\mathbf{I}$, identity matrix. The key to success in using (2.31) is to find an appropriate weighting matrix $\mathbf{A}$ that accounts for individual bands. As inspired by the three signal processing perspectives studied in Chang (2005) for OSP, this section investigates constrained LSMA with three different ways to select the weighting matrix $\mathbf{A}$ for (2.31), which are also derived from the same three perspectives. One is based on the parameter estimation perspective that is derived from the well-known Mahalanobis distance (MD) or Gaussian maximum likelihood estimator (GMLE). If the weighting matrix $\mathbf{A}$ in (2.31) is selected to be the inverse of the data sample covariance matrix $\mathbf{K}$, $\mathbf{K}^{-1}$, (2.31) becomes the MD or GMLE. The resulting constrained AC-LSMA is called MD-weighted AC-LSMA. As an alternative, if the weighting matrix $\mathbf{A}$ in (2.31) is replaced with $\mathbf{R}^{-1}$ (i.e., the inverse of data sample correlation matrix $\mathbf{R}$), (2.31) is then reduced to a form of the linearly constrained minimum variance (LCMV) classifier developed in Chang (2002a, b) which is referred to as LCMV-weighted AC-LSMA (Chang 2003). Another selection of the weighting matrix is based on pattern classification perspective which is derived from FLDA (Duda and Hart 1973). It has been shown in Chang and Ji (2006a, b) that with constraining Fisher's feature vectors to mutual orthogonal directions, maximizing Fisher's ratio is reduced to minimizing the within-class scatter matrix $\mathbf{S}_W$. As a result, selecting the $\mathbf{S}_W^{-1}$ for the weighting matrix $\mathbf{A}$ in (2.31) yields abundance-constrained Fisher's LSMA (AC-FLSMA) (Chang and Ji 2006a, b; Chang 2013), referred to as $\mathbf{S}_W^{-1}$-weighted AC-LSMA. A third way to select the weighting matrix is based on orthogonal subspace projection (OSP) (Harsanyi and Chang 1994) which is derived from signal detection perspectives. It is shown in Chang et al. (1998a, b) and Chang (2003) that the undesired signature rejection matrix, $P_{\mathbf{U}}^{\perp}$ used in the OSP can be approximated by $\mathbf{R}^{-1}$ if the prior knowledge of the undesired signatures in $\mathbf{U}$ is not available. Using this interpretation we can select $P_{\mathbf{U}}^{\perp}$ for the weighting matrix $\mathbf{A}$ in (2.31), which results in OSP-weighted AC-LSMA. An interesting finding in Chang and Ji (2006a, b) is that if the weighting matrix $\mathbf{A}$ is selected by the signature subspace projection (SSP) matrix that is formed by the signature matrix $\mathbf{M}$ in (2.1), the resulting SSP-weighted AC-LSMA can be shown in Chang and Ji (2006a, b) to be identical to the unweighted AC-LSMA in (2.2), in which case the $\mathbf{A}$ becomes the identity matrix. This is because both the SSP approach and LSMA are LS-based methods. As a consequence, the weighted matrix specified by the SSP does not provide any additional advantage. Nevertheless, as demonstrated by experiments, all these three types of weighted

AC-LSMA specified by appropriate selections for the weight matrix $\mathbf{A}$ in (2.31) generally perform better than unweighted AC-LSMA described by (2.2).

As noted, the LSE specified by (2.2) does not include any weighting matrix to account for significance of each band. In other words, this implies that the LSE caused by every band is equally significant. However, in reality this is not necessarily true. To address this issue, (2.31) includes a weighting matrix $\mathbf{A}$ to account for the LSEs resulting from different bands (i.e., replacing the $\mathbf{I}$ in (2.2) with $\mathbf{A}$). Consequently, we are required to solve the following $\mathbf{A}$-weighted LSE problem resulting from (2.31):

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\mathbf{A}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.32}$$

Suppose that $\mathbf{A}$ is a positive-definite and symmetric matrix. We can use $\mathbf{A}^{1/2}$, the square-root form of $\mathbf{A}$, to whiten the LSE in (2.32) as follows:

$$(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\left(\mathbf{A}^{1/2}\right)^{\mathrm{T}}\mathbf{A}^{1/2}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})=\left(\mathbf{A}^{1/2}\mathbf{r}-\mathbf{A}^{1/2}\mathbf{M}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(\mathbf{A}^{1/2}\mathbf{r}-\mathbf{A}^{1/2}\mathbf{M}\boldsymbol{\alpha}\right). \tag{2.33}$$

Using a linear transformation $\xi_{\mathbf{A}}$ defined by

$$\widehat{\mathbf{r}}=\xi_{\mathbf{A}}(\mathbf{r})=\mathbf{A}^{1/2}\mathbf{r}\ \text{and}\ \widehat{\mathbf{M}}=\xi_{\mathbf{A}}(\mathbf{M})=\mathbf{A}^{1/2}\mathbf{M}, \tag{2.34}$$

an $\mathbf{A}$-whitened LSE can be further simplified by $\xi_{\mathbf{A}}$ and given by

$$\min_{\boldsymbol{\alpha}}\left\{\left(\widehat{\mathbf{r}}-\widehat{\mathbf{M}}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(\widehat{\mathbf{r}}-\widehat{\mathbf{M}}\boldsymbol{\alpha}\right)\right\} \tag{2.35}$$

which is reduced to minimization of (2.2), except that both the image pixel vector $\mathbf{r}$ and the matrix $\mathbf{M}$ have been whitened by the weighting matrix $\mathbf{A}$ via the transformation $\xi_{\mathbf{A}}$. As a result, a new set of three types of $\mathbf{A}$-weighted AC-LSMA that are similar to (2.7), (2.14), and (2.18) can also be derived by replacing $\mathbf{M}$ in (2.7), (2.14), and (2.18) with $\widehat{\mathbf{M}}$ via $\xi_{\mathbf{A}}$ in (2.34) and they are referred to as MD-weighted SCLS, MD-weighted NCLS, and MD-weighted FCLS problems, respectively.

As shown in Chang (2005), LSMA can be interpreted from three different signal processing perspectives, signal detection which results in the OSP approach, parameter estimation which results in MD or GMLE, and pattern classification which results in FLDA. Following the same treatment, these three signal processing perspectives can also be used to develop in parallel various versions of $\mathbf{A}$-weighted AC LSMA by appropriately selecting a weighted matrix $\mathbf{A}$ in (2.31).

2.4.1 Weighting Matrix Derived from Parameter Estimation Perspective

Two ways to select the weighting matrix $\mathbf{A}$ in (2.31) which accounts for spectral correlation used in the parameter estimation are the use of the covariance spectral matrix $\mathbf{K}$ and the correlation spectral matrix $\mathbf{R}$.

2.4.1.1 MD-Weighted AC-LSMA

One well-known example of weight mean squared error is MD, also known as GMLE, which uses the data covariance matrix $\mathbf{K}^{-1}$ as a weighting matrix. Substituting the $\mathbf{K}^{-1}$ for the $\mathbf{A}$ in (2.31) yields

$$\min_{\alpha}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\mathbf{K}^{-1}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.36}$$

Replacing the $\mathbf{A}$ in (2.34) with $\mathbf{K}^{-1}$ yields a new linear transformation $\xi_{\mathbf{K}^{-1}}$ given by

$$\hat{\mathbf{r}}=\xi_{\mathbf{K}^{-1}}(\mathbf{r})=\mathbf{K}^{-1/2}\mathbf{r} \quad \text{and} \quad \hat{\mathbf{M}}=\xi_{\mathbf{K}^{-1}}(\mathbf{M})=\mathbf{K}^{-1/2}\mathbf{M}. \tag{2.37}$$

Then the resulting $\mathbf{K}^{-1}$-whitened LSE is found by,

$$\min_{\alpha}\left\{\left(\hat{\mathbf{r}}-\hat{\mathbf{M}}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(\hat{\mathbf{r}}-\hat{\mathbf{M}}\boldsymbol{\alpha}\right)\right\} \tag{2.38}$$

which is similar to (2.35). By means of (2.38), another new set of three types of $\mathbf{A}$-weighted AC-LSMA can be derived by replacing $\mathbf{M}$ in (2.7), (2.14) and (2.18) with $\hat{\mathbf{M}}$ via $\xi_{\mathbf{K}^{-1}}$ in (2.37) and they are referred to as MD-weighted SCLS, MD-weighted NCLS, and MD-weighted FCLS problems, respectively.

2.4.1.2 LCMV-Weighted AC-LSMA

The LSE in (2.36) was derived from the MD or Gaussian maximum likelihood estimation. If the data covariance matrix $\mathbf{K}$ in (2.36) is replaced with the data correlation matrix $\mathbf{R}$, a LCMV -based abundance-constrained LSE problems can be derived by

$$\min_{\alpha}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \tag{2.39}$$

which uses the data correlation matrix $\mathbf{R}$ as a weighting matrix. Using a linear transformation $\xi_{\mathbf{R}^{-1}}$ similar to $\xi_{\mathbf{K}^{-1}}$ defined in (2.37) by mapping $\mathbf{r}$ and $\mathbf{M}$ into

$$\bar{\mathbf{r}} = \xi_{\mathbf{R}^{-1}}(\mathbf{r}) = \mathbf{R}^{-1/2}\mathbf{r} \text{ and } \bar{\mathbf{M}} = \xi_{\mathbf{R}^{-1}}(\mathbf{M}) = \mathbf{R}^{-1/2}\mathbf{M}, \tag{2.40}$$

we can also obtain an $\mathbf{R}^{-1}$-whitened LSE problem given by

$$\min_{\boldsymbol{\alpha}} \left\{ (\bar{\mathbf{r}} - \bar{\mathbf{M}}\boldsymbol{\alpha})^{\mathrm{T}} (\bar{\mathbf{r}} - \bar{\mathbf{M}}\boldsymbol{\alpha}) \right\} \tag{2.41}$$

which is a correlation-based LSE problem. Three types of LCMV-weighted AC-LSMA can be derived by replacing $\mathbf{M}$ in (2.7), (2.14), and (2.18) with $\bar{\mathbf{M}}$ via $\xi_{\mathbf{R}^{-1}}$ in (2.40) and they are referred to as LCMV-weighted SCLS, LCMV-weighted NCLS, and LCMV-weighted FCLS problems, respectively.

2.4.2 *Weighting Matrix Derived from Fisher's Linear Discriminant Analysis Perspective*

FLDA is one of the most widely used pattern classification techniques in pattern recognition. An application of the FLDA to hyperspectral image classification was also explored in Chang and Ji (2006a, b). Its strength in pattern classification lies in the criterion used for optimality, which is called Fisher's ratio, defined as the ratio of between-class scatter matrix to within-class scatter matrix. More specifically, assume that there are n training sample vectors given by $\{\mathbf{r}_i\}_{i=1}^{n}$ for p-class classification, $C_1, C_2, \ldots, C_p$ with n_j being the number of training sample vectors in jth class C_j. Let $\boldsymbol{\mu}$ be the global mean of the entire training sample vectors, denoted by $\boldsymbol{\mu} = (1/n)\sum_{i=1}^{n} \mathbf{r}_i$ and $\boldsymbol{\mu}_j$ be the mean of the training sample vectors in the jth class C_j, denoted by $\boldsymbol{\mu}_j = \left(1/n_j\right)\sum_{\mathbf{r}_i \in C_j} \mathbf{r}_i$. Now we can define the within-class scatter matrix, $\mathbf{S}_W$ and between-class scatter matrix $\mathbf{S}_B$ as

$$\mathbf{S}_W = \sum_{j=1}^{p} \mathbf{S}_j,$$

where

$$\mathbf{S}_j = \sum_{\mathbf{r} \in C_j} \left(\mathbf{r} - \boldsymbol{\mu}_j\right)\left(\mathbf{r} - \boldsymbol{\mu}_j\right)^{\mathrm{T}} \tag{2.42}$$

$$\mathbf{S}_B = \sum\nolimits_{j=1}^{p} n_j \left(\boldsymbol{\mu}_j - \boldsymbol{\mu}\right)\left(\boldsymbol{\mu}_j - \boldsymbol{\mu}\right)^{\mathrm{T}} \tag{2.43}$$

Using (2.42) and (2.43), Fisher's ratio (also known as Raleigh's quotient) is then defined by

$$\frac{\mathbf{x}^{\mathrm{T}}\mathbf{S}_B\mathbf{x}}{\mathbf{x}^{\mathrm{T}}\mathbf{S}_W\mathbf{x}} \text{ for any vector } \mathbf{x}. \tag{2.44}$$

Fisher linear discriminant analysis (FLDA) can be used to find a set of feature vectors that maximize Fisher's ratio specified by (2.44). The number of feature vectors found by Fisher's ratio is determined by the number of classes, p, to be classified, which is $p - 1$.

It has been shown in Chang and Ji (2006a, b) that a Fisher's ratio-based LSE problem, referred to as Fisher linear spectral mixture analysis (FLSMA) could be formulated as

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\mathbf{S}_W^{-1}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\} \tag{2.45}$$

with $\mathbf{S}_W^{-1}$ being used as a weighting matrix to replace the weighting matrix $\mathbf{A}$ in (2.31). So, using a transformation $\xi_{\mathbf{S}_W^{-1}}$ defined by

$$\tilde{\mathbf{r}} = \xi_{\mathbf{S}_W^{-1}}(\mathbf{r}) = \mathbf{S}_W^{-1/2}\mathbf{r} \text{ and } \tilde{\mathbf{M}} = \xi_{\mathbf{S}_W^{-1}}(\mathbf{M}) = \mathbf{S}_W^{-1/2}\mathbf{M}, \tag{2.46}$$

(2.45) can be whitened by $\mathbf{S}_W^{-1}$ and becomes

$$\min_{\boldsymbol{\alpha}}\left\{\left(\tilde{\mathbf{r}}-\tilde{\mathbf{M}}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(\tilde{\mathbf{r}}-\tilde{\mathbf{M}}\boldsymbol{\alpha}\right)\right\}. \tag{2.47}$$

Therefore, three types of $\mathbf{S}_W^{-1}$-weighted AC-LSMA can be derived by replacing $\mathbf{M}$ in (2.7), (2.14), and (2.18) with $\tilde{\mathbf{M}}$ via $\xi_{\mathbf{S}_W^{-1}}$ in (2.46) and are referred to as $\mathbf{S}_W^{-1}$-weighted SCLS, $\mathbf{S}_W^{-1}$-weighted NCLS, and $\mathbf{S}_W^{-1}$-weighted FCLS problems, respectively.

2.4.3 *Weighting Matrix Derived from Orthogonal Subspace Projection Perspective*

As we have seen in Sects. 2.4.1 and 2.4.2, the weighting matrix $\mathbf{A}$ was selected by sample spectral correlation matrices and Fisher's ratio which resulted from the maximum likelihood estimator and FLDA. In this section, we investigate the selection of the weighting matrix $\mathbf{A}$ based on various OSP criteria.

2.4.3.1 OSP-Weighted AC-LSMA

According to the signal-decomposed interference-annihilated (SDIA) model in Du and Chang (2004a, b), the signal sources can be decomposed into signal sources which are assumed to be in the signature matrix $\mathbf{M}$ and unwanted signal sources which are assumed to be interferers to the signal sources in the $\mathbf{M}$. If we let $\mathbf{U}$ be the unwanted signature matrix made up of such interferers, we can project all image pixels onto the space $<\mathbf{U}>^{\perp}$ that is orthogonal to the space linearly spanned by the signal sources in $\mathbf{U}$ and then perform the LSE problem specified by (2.2) in $<\mathbf{U}>^{\perp}$. Inspired by this approach, the weighting matrix $\mathbf{A}$ in (2.31) can be selected by the unwanted signature rejector, $P_{\mathbf{U}}^{\perp}$, defined in Harsanyi and Chang (1994) by (2.5). The resulting LSE problem from replacing $\mathbf{A}$ in (2.31) with $P_{\mathbf{U}}^{\perp}$ in (2.5) is

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}P_{\mathbf{U}}^{\perp}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.48}$$

Since $P_{\mathbf{U}}^{\perp}$ is idempotent, $P_{\mathbf{U}}^{\perp}=\left(P_{\mathbf{U}}^{\perp}\right)^{2}$ and $\left(P_{\mathbf{U}}^{\perp}\right)^{\mathrm{T}}=P_{\mathbf{U}}^{\perp}$. This implies that

$$\begin{aligned}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}P_{\mathbf{U}}^{\perp}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha}) &= (\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}\left(P_{\mathbf{U}}^{\perp}\right)^{\mathrm{T}}P_{\mathbf{U}}^{\perp}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\\ &= \left(P_{\mathbf{U}}^{\perp}\mathbf{r}-P_{\mathbf{U}}^{\perp}\mathbf{M}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(P_{\mathbf{U}}^{\perp}\mathbf{r}-P_{\mathbf{U}}^{\perp}\mathbf{M}\boldsymbol{\alpha}\right).\end{aligned} \tag{2.49}$$

Using a linear transformation $\xi_{P_{\mathbf{U}}^{\perp}}$ defined by mapping $\mathbf{r}$ and $\mathbf{M}$ into

$$\vec{\mathbf{r}}=\xi_{P_{\mathbf{U}}^{\perp}}(\mathbf{r})=P_{\mathbf{U}}^{\perp}\mathbf{r}\ \text{and}\ \vec{\mathbf{M}}=\xi_{P_{\mathbf{U}}^{\perp}}(\mathbf{M})=P_{\mathbf{U}}^{\perp}\mathbf{M}, \tag{2.50}$$

we can also obtain a form similar to (2.35) given by

$$\min_{\boldsymbol{\alpha}}\left\{\left(\vec{\mathbf{r}}-\vec{\mathbf{M}}\boldsymbol{\alpha}\right)^{\mathrm{T}}\left(\vec{\mathbf{r}}-\vec{\mathbf{M}}\boldsymbol{\alpha}\right)\right\} \tag{2.51}$$

which is referred to as the OSP-weighted abundance-constrained LSE problem. Consequently, three types of OSP-weighted AC LSMA can be derived by replacing $\mathbf{M}$ in (2.7), (2.14), and (2.18) with $\vec{\mathbf{M}}$ via $\xi_{P_{\mathbf{U}}^{\perp}}$ in (2.50) and they are referred to as OSP-weighted SCLS, OSP-weighted NCLS, and OSP-weighted FCLS problems, respectively. A key to success in the OSP-weighted ACLSMA is to find the unknown signal sources used in the matrix $\mathbf{U}$ in an unsupervised manner. The one of particular interest is called the Automatic Target Detection and Classification Algorithm (ATDCA) that was developed in Ren and Chang (2003) and can be used for this purpose.

2.4.3.2 SSP-Weighted AC-LSMA

As an alternative to (2.48), we can also formulate an LSE problem based on performing abundance estimation in the space that is linearly spanned by the signal sources in the signature matrix $\mathbf{M}$ exclusively. Such an LSE problem resulting from replacing $P_{\mathbf{U}}^{\perp}$ in (2.5) with a signature subspace projector, $P_{\mathbf{M}}$, defined in Chang et al. (1998a, b) and Chang (2003) by

$$P_{\mathbf{M}} = \mathbf{M}(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}} \tag{2.52}$$

is referred to as SSP-weighted AC-LSMA which can be used to find the solution to the following optimization problem:

$$\min_{\boldsymbol{\alpha}}\left\{(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}P_{\mathbf{M}}(\mathbf{r}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.53}$$

Once again, $P_{\mathbf{M}}$ is idempotent and $(P_{\mathbf{M}})^2 = P_{\mathbf{M}}$ and $(P_{\mathbf{M}})^{\mathrm{T}} = P_{\mathbf{M}}$. Using a linear transformation $\xi_{P_{\mathbf{M}}}$ defined by mapping $\mathbf{r}$ and $\mathbf{M}$ into

$$\vec{\mathbf{r}} = \xi_{P_{\mathbf{M}}}(\mathbf{r}) = P_{\mathbf{M}}\mathbf{r} \text{ and } \vec{\mathbf{M}} = \xi_{P_{\mathbf{M}}}(\mathbf{M}) = P_{\mathbf{M}}\mathbf{M} = \mathrm{M}, \tag{2.54}$$

(2.54) becomes

$$\min_{\boldsymbol{\alpha}}\left\{(\vec{\mathbf{r}}-\mathbf{M}\boldsymbol{\alpha})^{\mathrm{T}}(\vec{\mathbf{r}}-\mathbf{M}\boldsymbol{\alpha})\right\}. \tag{2.55}$$

Interestingly, the solution to (2.55) is

$$\begin{aligned}\hat{\boldsymbol{\alpha}}(\vec{\mathbf{r}}) &= (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}\vec{\mathbf{r}} = (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}P_{\mathbf{M}}\mathbf{r} \\ &= (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{M}(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r} = (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{r} = \hat{\boldsymbol{\alpha}}^{\mathrm{LS}}(\mathbf{r})\end{aligned} \tag{2.56}$$

which is identical to the unconstrained LS LSMA solution given by (2.3). As a result, the three types of SSP-weighted AC-LSMA obtained by the linear transformation $\xi_{P_{\mathbf{M}}}$ in (2.54) turn out to be the same unweighted ASC-LSMA, ANC-LSMA, and AFC-LSMA described by (2.7), (2.14), and (2.18), respectively. This is because the weighted matrix specified by SSP does not provide any additional advantage as shown by (2.56) because $P_{\mathbf{M}}\mathbf{M} = \mathbf{M}$.

2.5 Kernel-Based WAC-LSMA

Kernel-based approaches have found potentials in various applications in remote sensing target detection and image classification, specifically LSMA. Recently, kernel-based LSMA (KLSMA) has also been developed to improve LSMA

unmixing performance (Liu et al. 2011). This section extends the LSMA techniques OSP, LSOSP, NCLS, and FCLS to their kernel counterparts. Despite the fact that a kernel-based OSP and a kernel-based FCLS have been derived in Kwon and Nasrabadi (2005b) and Broadwater et al. (2007), respectively, this section presents a unified theory of KLSMA by first developing the kernel-based OSP (KOSP) and LSOSP (KLSOSP) and then using the KLSOSP to derive the kernel-based NCLS (KNCLS) which further leads to the derivation of the KFCLS.

In a high dimensional feature space, all the data sample vectors are mapped into feature space F with a nonlinear function $\phi : \Re^N \to F$ by $\mathbf{x} \mapsto \phi(\mathbf{x})$. The goal of kernel WAC-LSMA is to find a solution to solve

$$\min_{\boldsymbol{\alpha}} \Big\{ (\phi(\mathbf{r}) - \phi(\mathbf{M})\boldsymbol{\alpha})^{\mathrm{T}} \mathbf{A}_{\phi} (\phi(\mathbf{r}) - \phi(\mathbf{M})\boldsymbol{\alpha}) \Big\}. \tag{2.57}$$

Because $\mathbf{A}_{\phi}$ is a non-negative definite and symmetric matrix in the feature space, we can de-weight the LSE in (2.31) via $\mathbf{A}_{\phi}^{1/2}$ as follows:

$$\begin{aligned} &(\phi(\mathbf{r}) - \phi(\mathbf{M})\boldsymbol{\alpha})^{\mathrm{T}} (\mathbf{A}_{\phi}^{1/2})^{T} \mathbf{A}_{\phi}^{1/2} (\phi(\mathbf{r}) - \phi(\mathbf{M})\boldsymbol{\alpha}) \\ &= \Big(\mathbf{A}_{\phi}^{1/2} \phi(\mathbf{r}) - \mathbf{A}_{\phi}^{1/2} \phi(\mathbf{M})\boldsymbol{\alpha} \Big)^{\mathrm{T}} \Big(\mathbf{A}_{\phi}^{1/2} \phi(\mathbf{r}) - \mathbf{A}_{\phi}^{1/2} \phi(\mathbf{M})\boldsymbol{\alpha} \Big). \end{aligned} \tag{2.58}$$

Using the following transformation $\xi_{\mathbf{A}_{\phi}}$ defined by

$$\hat{\phi}(\mathbf{r}) = \xi_{\mathbf{A}_{\phi}}(\phi(\mathbf{r})) = \mathbf{A}_{\phi}^{1/2} \phi(\mathbf{r}) \quad \hat{\phi}(\mathbf{M}) = \xi_{\mathbf{A}_{\phi}}(\phi(\mathbf{M})) = \mathbf{A}_{\phi}^{1/2} \phi(\mathbf{M}), \tag{2.59}$$

$\mathbf{A}_{\phi}$-de-weighted LSE can be further simplified as

$$\min_{\boldsymbol{\alpha}} \Big\{ \Big(\hat{\phi}(\mathbf{r}) - \hat{\phi}(\mathbf{M})\boldsymbol{\alpha} \Big)^{\mathrm{T}} \Big(\hat{\phi}(\mathbf{r}) - \hat{\phi}(\mathbf{M})\boldsymbol{\alpha} \Big) \Big\}. \tag{2.60}$$

Three types of kernel-based WAC-LSMA problems, referred to as kernel-based MD-WACSCLS, kernel-based MD-WACNCLS, and kernel-based MD-WACFCLS, are described as follows.

The SCLS solution to (2.60) in the feature space using $\phi(\mathbf{M})$ and $\phi(\mathbf{r})$ is given by

$$\hat{\boldsymbol{\alpha}}^{\mathrm{SCLS},\phi} = P_{\mathrm{M},1_{\phi}}^{\perp} \hat{\boldsymbol{\alpha}}^{\mathrm{LS},\phi}(\mathbf{r}) + (\hat{\phi}(\mathbf{M})^{\mathrm{T}} \hat{\phi}(\mathbf{M}))^{-1} \hat{\phi}(1) [\hat{\phi}(1)^{\mathrm{T}} \hat{\phi}(\mathbf{M})^{\mathrm{T}} \hat{\phi}(\mathbf{M}))^{-1} \hat{\phi}(1)]^{-1} \tag{2.61}$$

where

$$\hat{\boldsymbol{\alpha}}^{\mathrm{LS},\phi}(\mathbf{r}) = (\hat{\phi}(\mathbf{M})^{\mathrm{T}} \hat{\phi}(\mathbf{M}))^{-1} \hat{\phi}(\mathbf{M})^{\mathrm{T}} \hat{\phi}(\mathbf{r}) \tag{2.62}$$

$$P^{\perp}_{\mathrm{M},1_\phi} = \hat{\phi}(\mathbf{I}) - (\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M}))^{-1}\hat{\phi}(1)[\hat{\phi}(1)^{\mathrm{T}}(\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M}))^{-1}\hat{\phi}(1)]^{-1}\hat{\phi}(1)^{\mathrm{T}}. \tag{2.63}$$

As for KNCLS and KFCLS, the following two equations will be used to iterate the process:

$$\hat{\boldsymbol{\alpha}}^{\hat{\phi}} = (\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M}))^{-1}\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{r}) - (\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M}))^{-1}\boldsymbol{\lambda} \tag{2.64}$$

$$\boldsymbol{\lambda} = \hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{r}) - \hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M})\hat{\boldsymbol{\alpha}}^{\phi} \tag{2.65}$$

where $\hat{\boldsymbol{\alpha}}^{\hat{\phi}}$ is either $\hat{\boldsymbol{\alpha}}^{\mathrm{NCLS},\hat{\phi}}$ or $\hat{\boldsymbol{\alpha}}^{\mathrm{FCLS},\hat{\phi}}$. It is easy to see that only two terms are of interest in the kernelization process, $\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M})$ and $\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{r})$, which can be obtained from

$$\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{M}) = [\mathbf{A}_\phi^{1/2}\phi(\mathbf{M})]^{\mathrm{T}}[\mathbf{A}_\phi^{1/2}\phi(\mathbf{M})] = \phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{M}) \tag{2.66}$$

$$\hat{\phi}(\mathbf{M})^{\mathrm{T}}\hat{\phi}(\mathbf{r}) = [\mathbf{A}_\phi^{1/2}\phi(\mathbf{M})]^{\mathrm{T}}[\mathbf{A}_\phi^{1/2}\phi(\mathbf{r})] = \phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r}). \tag{2.67}$$

In order to kernelize $\phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{M})$ and $\phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r})$, we first expand both equations by the following matrix forms:

$$\begin{aligned}\phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_l\phi(\mathbf{M}) &= \begin{bmatrix}\phi(\mathbf{m}_1)^{\mathrm{T}}\\ \phi(\mathbf{m}_2)^{\mathrm{T}}\\ \vdots\\ \phi(\mathbf{m}_p)^{T}\end{bmatrix}\mathbf{A}_\phi[\,\phi(\mathbf{m}_1)\quad \phi(\mathbf{m}_2)\quad \cdots\quad \phi(\mathbf{m}_p)\,]\\ &= \begin{bmatrix}\phi(\mathbf{m}_1)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_1) & \phi(\mathbf{m}_1)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_2) & \cdots & \phi(\mathbf{m}_1)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_p)\\ \phi(\mathbf{m}_2)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_1) & \ddots & & \vdots\\ \vdots & & \ddots & \vdots\\ \phi(\mathbf{m}_p)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_1) & \cdots & \cdots & \phi(\mathbf{m}_p)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{m}_p)\end{bmatrix}\end{aligned} \tag{2.68}$$

and

$$\phi(\mathbf{M})^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r}) = \begin{bmatrix}\phi(\mathbf{m}_1)^{\mathrm{T}}\\ \phi(\mathbf{m}_2)^{\mathrm{T}}\\ \vdots\\ \phi(\mathbf{m}_p)^{\mathrm{T}}\end{bmatrix}\mathbf{A}_\phi\phi(\mathbf{r}) = \begin{bmatrix}\phi(\mathbf{m}_1)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r})\\ \phi(\mathbf{m}_2)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r})\\ \vdots\\ \phi(\mathbf{m}_p)^{\mathrm{T}}\mathbf{A}_\phi\phi(\mathbf{r})\end{bmatrix}. \tag{2.69}$$

As a matter of fact, each entry in the matrix (2.68) as well as the vector (2.70) can be calculated individually by the following simple kernelization problem:

$$\phi(\mathbf{a})^{\mathrm{T}}\mathbf{A}_{\phi}\phi(\mathbf{b}) \text{ where } \mathbf{a},\mathbf{b}\in\{(\mathbf{m}_1,\mathbf{m}_2,\ldots,\mathbf{m}_p)\cup\mathbf{r}\} \tag{2.70}$$

where four different approaches to selection of weighting matrix $\mathbf{A}$ to kernelize $\mathbf{A}_{\phi}$: (1) covariance matrix $\mathbf{K}$ used by MD, (2) correlation matrix $\mathbf{R}$ used by CEM or LCMV (Frost 1972), (3) within-class covariance matrix $\mathbf{S}_{\mathrm{W}}$ used by FLDA, and (4) unwanted and desired signature subspaces resulting from OSP.

2.5.1 A = Inverse of Covariance Matrix, K^{-1}

In this case, set $\mathbf{A}_{\phi}=\mathbf{K}_{\phi}^{-1}$ where K_{ϕ} is a kernelized covariance matrix K in a high dimensional feature space. The K_{ϕ} can be rewritten as $\mathbf{K}_{\phi}=\mathbf{V}\Lambda\mathbf{V}^{\mathrm{T}}$ where Λ and V are formed by eigenvalues, along with their corresponding eigenvectors of K_{ϕ}, respectively. As a result, (2.70) can be re-expressed as

$$\begin{aligned}\phi(\mathbf{a})^{\mathrm{T}}\mathbf{A}_{\phi}\phi(\mathbf{b}) &= \phi(\mathbf{a})^{\mathrm{T}}\mathbf{K}_{\phi}^{-1}\phi(\mathbf{b})\\ &= \phi(\mathbf{a})^{\mathrm{T}}(\mathbf{V}\Lambda\mathbf{V}^{\mathrm{T}})^{-1}\phi(\mathbf{b}) = (\phi(\mathbf{a})^{\mathrm{T}}\mathbf{V})\Lambda^{-1}(\mathbf{V}^{\mathrm{T}}\phi(\mathbf{b})).\end{aligned} \tag{2.71}$$

According to the derivation of KPCA (Scholkopf et al. 1999a, b), a kernel gram matrix $\mathbf{K}=(\phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_j))$, for $i, j = 1, \ldots, N$ can be derived similarly, as follows.

Let $\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_N$ denote the normalized eigenvalues of K, with $\mathbf{v}_1,\mathbf{v}_2,\ldots,\mathbf{v}_N$ as their corresponding eigenvectors. Let $\mathbf{v}_m$ be the eigenvector of the first non-positive eigenvalue λ_m. Then for $k = 1, 2, \ldots, m$ we can further expand v_k by a set of coefficients $\{\beta_1,\beta_2,\ldots,\beta_N\}$ via the kernelized training sample vectors, $\{\phi(\mathbf{x}_1),\phi(\mathbf{x}_2),\ldots,\phi(\mathbf{x}_N)\}$ as follows:

$$\phi(\mathbf{a})^{\mathrm{T}}\mathbf{v}_k=\phi(\mathbf{a})^{\mathrm{T}}\sum_{i=1}^{N}\beta_{k_i}\phi(\mathbf{x}_i)=\sum_{i=1}^{N}\beta_{k_i}(\phi(\mathbf{a})\cdot\phi(\mathbf{x}_i))=\sum_{i=1}^{N}\beta_{k_i}K(\mathbf{a}\cdot\mathbf{x}_i) \tag{2.72}$$

$$\mathbf{v}_k^{\mathrm{T}}\phi(\mathbf{b})=\sum_{i=1}^{N}\beta_{k_i}\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{b})=\sum_{i=1}^{N}\beta_{k_i}(\phi(\mathbf{x}_i)\cdot\phi(\mathbf{b}))=(\sum_{i=1}^{N}\beta_{k_i}K(\mathbf{x}_i,\mathbf{b})) \tag{2.73}$$

and

$$\mathbf{\Lambda}^{-1} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_m \end{bmatrix}^{-1}. \tag{2.74}$$

Consequently, solutions of KWAV-SCLS in (2.61) and of KWAC-NCLS and KWAC-FCLS in (2.64) and (2.65) can be found by calculating (2.66) and (2.67) via (2.70) through (2.72)–(2.74).

2.5.2 A = Inverse of Covariance Matrix, R^{-1}

In this case, $\mathbf{A}_\phi = \mathbf{R}_\phi^{-1}$. Following the same treatment carried out in Sect. 2.5.1, $\phi(\mathbf{a})^{\mathrm{T}}\mathbf{R}_\phi^{-1}\phi(\mathbf{b})$ can be calculated using (2.72)–(2.74). The only difference is that the kernel gram matrix $\mathbf{R}$ needs no normalization before eigen-decomposition as it is applied to $\mathbf{K}$.

2.5.3 A = Inverse of Within-Class Matrix, $\mathbf{S}_{\mathrm{W}}^{-1}$

In this case, $\mathbf{A}_\phi = \left(\mathbf{S}_{\mathrm{W}}^{\phi}\right)^{-1}$. Based on the work in Chang and Ji (2006a, b) we assume that there are n training sample vectors, $\{\mathbf{r}_i\}_{i=1}^n$, for p-class classification, $c_1, c_2, \ldots, c_p$, with n_j being the number of training sample vectors in the jth class c_j. Let $\boldsymbol{\mu}$ be the global mean of the entire training sample vectors, denoted by $\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^n \mathbf{r}_i$ and $\boldsymbol{\mu}_j$ be the mean of the training sample vectors in the jth class c_j, denoted by $\boldsymbol{\mu}_j = \frac{1}{n_j}\sum_{\mathbf{r}_i \in c_j} \mathbf{r}_i$. Then the within-class scatter matrix, $\mathbf{S}_W$, between-class scatter matrix, $\mathbf{S}_B$, and total scatter matrix can be defined as $\mathbf{S}_{\mathrm{W}} = \sum_{j=1}^p \mathbf{S}_j$ with $\mathbf{S}_j = \sum_{\mathbf{r}\in C_j} (\mathbf{r} - \boldsymbol{\mu}_j)(\mathbf{r} - \boldsymbol{\mu}_j)^{\mathrm{T}}$ and $\mathbf{S}_{\mathrm{B}} = \sum_{j=1}^p n_j(\boldsymbol{\mu}_j - \boldsymbol{\mu})(\boldsymbol{\mu}_j - \boldsymbol{\mu})^{\mathrm{T}}$. In addition, let $\boldsymbol{\mu}_c$ denote the sample mean of a particular class c. In a high dimensional feature space, the input data are mapped into a high dimensional feature space F with a nonlinear function $\Phi : \Re^N \to F$ by $\mathbf{x} \mapsto \Phi(\mathbf{x})$. The kernel-based feature vector constrained-FLSMA (KFVC-FLSMA) can be derived from the FCV-FLSMA in Chang and Ji (2006b) and presented by

$$\alpha(\mathbf{r}) = \mathbf{W}^{\mathrm{KFVC-FLSMA}}\mathbf{r} = (\Phi(\mathbf{M})^{\mathrm{T}}\Phi(\mathbf{S}_{\mathrm{w}})^{-1}\Phi(\mathbf{M}))^{-1}\Phi(\mathbf{M})^{\mathrm{T}}\Phi(\mathbf{S}_{\mathrm{w}})^{-1}\Phi(\mathbf{r}). \tag{2.75}$$

To simplify the derivation, we only focus on one signature vector $\mathbf{d}$ at a time instead of signature matrix $\mathbf{M}$. So the equation can be simplified to

$$\alpha_{\mathbf{d}}(\mathbf{r}) = \frac{\Phi(\mathbf{d})^{\mathrm{T}}\Phi(\mathbf{S}_{\mathrm{w}})^{-1}\Phi(\mathbf{r})}{\Phi(\mathbf{d})^{\mathrm{T}}\Phi(\mathbf{S}_{\mathrm{w}})^{-1}\Phi(\mathbf{d})} \tag{2.76}$$

where

$$\Phi(\mathbf{S}_{\mathrm{w}}) = \sum_{c=1}^{p}\sum_{j=1}^{N_c}(\Phi(\mathbf{x}_{c_j}) - \Phi(\boldsymbol{\mu}_c))(\Phi(\mathbf{x}_{c_j}) - \Phi(\boldsymbol{\mu}_c))^{\mathrm{T}}. \tag{2.77}$$

We rewrite $\Phi(\mathbf{S}_{\mathrm{w}}) = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathrm{T}}$ with $\mathbf{V}$ formed by the eigenvectors of $\Phi(\mathbf{S}_{\mathrm{w}})$ and $\boldsymbol{\Lambda}$ is an eigenvector matrix found by their associated eigenvalues:

$$\Phi(\mathbf{S}_{\mathrm{w}})^{-1} = \mathbf{V}\boldsymbol{\Lambda}^{-1}\mathbf{V}^{\mathrm{T}}. \tag{2.78}$$

Substituting (2.78) into (2.76), we have

$$\alpha_{\mathbf{d}}(\mathbf{r}) = \frac{\Phi(\mathbf{d})^{\mathrm{T}}(\mathbf{V}\boldsymbol{\Lambda}^{-1}\mathbf{V}^{\mathrm{T}})\Phi(\mathbf{r})}{\Phi(\mathbf{d})^{\mathrm{T}}(\mathbf{V}\boldsymbol{\Lambda}^{-1}\mathbf{V}^{\mathrm{T}})\Phi(\mathbf{d})} = \frac{(\Phi(\mathbf{d})^{\mathrm{T}}\mathbf{V})\boldsymbol{\Lambda}^{-1}(\mathbf{V}^{\mathrm{T}}\Phi(\mathbf{r}))}{(\Phi(\mathbf{d})^{\mathrm{T}}\mathbf{V})\boldsymbol{\Lambda}^{-1}(\mathbf{V}^{\mathrm{T}}\Phi(\mathbf{d}))}. \tag{2.79}$$

Now we have to find eigenvalues $\lambda \geq 0$ and their corresponding eigenvectors $\mathbf{V}$ satisfying

$$\lambda\mathbf{V} = \Phi(\mathbf{S}_{\mathrm{w}})\mathbf{V}. \tag{2.80}$$

Consider the set of equations

$$\lambda(\Phi(\mathbf{x}_k)\cdot\mathbf{V}) = (\Phi(\mathbf{x}_k)\cdot\Phi(\mathbf{S}_{\mathrm{w}})\mathbf{V}) \quad \text{for all } k = 1,\ldots,N. \tag{2.81}$$

According to the theory of reproducing kernel, and following the same treatment deriving (2.72)–(2.74), the $\mathbf{V}$ must lie in the space spanned by $\Phi(\mathbf{x}_1),\ldots,\Phi(\mathbf{x}_N)$:

$$\mathbf{V} = \sum_{i=1}^{N}\beta_i\Phi(\mathbf{x}_i). \tag{2.82}$$

Substituting (2.77) and (2.82) into (2.81) yields

$$\lambda\left(\Phi(\mathbf{x}_k)\cdot\sum_{i=1}^{N}\beta_i\Phi(\mathbf{x}_i)\right) = (\Phi(\mathbf{x}_k)\cdot\Phi(\mathbf{S}_W))\left(\sum_{i=1}^{N}\beta_i\Phi(\mathbf{x}_i)\right). \tag{2.83}$$

The left and right sides of (2.83) can be further presented by

$$\lambda \sum_{i=1}^{N} \beta_i (\Phi(\mathbf{x}_k) \cdot \Phi(\mathbf{x}_i)) \tag{2.84}$$

and

$$\begin{aligned} \Phi(\mathbf{x}_k) &\left(\sum_{c=1}^{p} \sum_{j=1}^{N_c} \left(\Phi(\mathbf{x}_{c_j}) - \Phi(\boldsymbol{\mu}_c) \right) \cdot \left(\Phi(\mathbf{x}_{c_j}) - \Phi(\boldsymbol{\mu}_c) \right) \right) \left(\sum_{i=1}^{N} \beta_i \Phi(\mathbf{x}_i) \right) \\ &= \sum_{c=1}^{p} \beta_i \sum_{j=1}^{N_c} \Phi(\mathbf{x}_k) \left(\left(\sum_{j=1}^{N} \Phi(\mathbf{x}_{c_j}) \Phi(\mathbf{x}_{c_j})^{\mathrm{T}} - \Phi(\mathbf{x}_{c_j}) \Phi(\boldsymbol{\mu}_c)^{\mathrm{T}} - \Phi(\boldsymbol{\mu}_c) \Phi(\mathbf{x}_{c_j})^{\mathrm{T}} \right. \right. \\ &\qquad \left. \left. + \Phi(\boldsymbol{\mu}_c) \Phi(\boldsymbol{\mu}_c)^{\mathrm{T}} \right) \Phi(\mathbf{x}_i) \right) \end{aligned} \tag{2.85}$$

respectively, where (2.85) can be expressed as a special matrix $\mathbf{K}_{\mathrm{w}}$ as follows:

$$\mathbf{K}_{\mathrm{w}} = \mathbf{K}^{-1} \sum_{c=1}^{p} \left(\mathbf{K}_c \mathbf{K}_c^{\mathrm{T}} - \mathbf{K}_c \mathbf{K}_{\boldsymbol{\mu}_c}^{\mathrm{T}} - \mathbf{K}_{\boldsymbol{\mu}_c} \mathbf{K}_c^{\mathrm{T}} + \mathbf{K}_{\boldsymbol{\mu}_c} \mathbf{K}_{\boldsymbol{\mu}_c}^{\mathrm{T}} \right) \tag{2.86}$$

where

$$\mathbf{K} = (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)) \quad \text{for } i, j = 1, \ldots, N \tag{2.87}$$

$$\mathbf{K}_{\mu_c} = (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_{\mu_c})) \quad \text{for} i = 1, \ldots, N \quad \text{for } j = 1, \ldots, N_{\mathrm{c}} \tag{2.88}$$

$$\mathbf{K}_c = (\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_{c_j})) \quad \text{for } i = 1, \ldots, N \quad \text{for } j = 1, \ldots, N_{\mathrm{c}} \tag{2.89}$$

Equations (2.88) and (2.89) are then used to compute the sample spectral information between pixel $\mathbf{x}_i$ and training samples in each class c. Once $\mathbf{K}_{\mathrm{w}}$ is obtained, the eigenvalues and eigenvectors obtained from eigen-decomposition of $\mathbf{K}_{\mathrm{w}}$ are further used to compute (2.70) by (2.83). More details of the above derivations can be found in Liu (2011).

2.5.4 *A = Unwanted Signature Projection Matrix,* $P_{\mathbf{U}_\Phi}^{\perp}$

In this case, $\mathbf{U}_\phi = \left[\phi(\mathbf{m}_1) \phi(\mathbf{m}_2) \ldots \phi(\mathbf{m}_{p-1}) \right]$ and we set $\mathbf{A}_\phi = P_{\mathbf{U}_\phi}^{\perp}$. Since $P_{\mathbf{U}_\Phi}^{\perp} = \mathbf{I}_\Phi - \mathbf{U}_\Phi (\mathbf{U}_\Phi^{\mathrm{T}} \mathbf{U}_\Phi)^{-1} \mathbf{U}_\Phi^{\mathrm{T}}$, we can express (2.70) as

$$\phi(\mathbf{a})^{\mathrm{T}} P_{\mathbf{U}_\phi}^{\perp} \phi(\mathbf{b}) = \phi(\mathbf{a})^{\mathrm{T}} \phi(\mathbf{b}) - \phi(\mathbf{a})^{\mathrm{T}} \mathbf{U}_\phi (\mathbf{U}_\phi^{\mathrm{T}} \mathbf{U}_\phi)^{-1} \mathbf{U}_\phi^{\mathrm{T}} \phi(\mathbf{b}). \tag{2.90}$$

Then the kernel trick is applied to (2.90) to obtain

$$\phi(\mathbf{a})^{\mathrm{T}} P_{\mathbf{U}_\phi}^{\perp} \phi(\mathbf{b}) = K(\mathbf{a}, \mathbf{b}) - K(\mathbf{a}, \mathbf{U}) K(\mathbf{U}, \mathbf{U})^{-1} K(\mathbf{U}, \mathbf{b}). \tag{2.91}$$

which is the same as that used to derive KLSMA (Liu et al. 2012).

2.5.5 *A = Signature Projection Matrix, $P_{\mathbf{M}_\Phi}$*

Analogous to Sect. 3.4, the same derivation used there can also be applied to the case of $\mathbf{A}_\phi = P_{\mathbf{M}_\phi}$ where $\mathbf{M}_\phi = \left[\phi(\mathbf{m}_1)\phi(\mathbf{m}_2)\ldots\phi(\mathbf{m}_p)\right]$. Since $P_{\mathbf{M}_\phi} = \mathbf{M}_\phi (\mathbf{M}_\phi^{\mathrm{T}} \mathbf{M}_\phi)^{-1} \mathbf{M}_\phi$, (2.70) can be re-expressed by

$$\phi(\mathbf{a})^{\mathrm{T}} P_{\mathbf{M}_\phi} \phi(\mathbf{b}) = \Phi(\mathbf{a})^{\mathrm{T}} \mathbf{M}_\phi (\mathbf{M}_\phi^{\mathrm{T}} \mathbf{M}_\phi)^{-1} \mathbf{M}_\phi \phi(\mathbf{b}) \tag{2.92}$$

which can be further simplified by the kernel trick to obtain

$$\phi(\mathbf{a})^{\mathrm{T}} P_{\mathbf{M}_\phi} \phi(\mathbf{b}) = K(\mathbf{a}, \mathbf{M}) K(\mathbf{M}, \mathbf{M})^{-1} K(\mathbf{M}, \mathbf{b}) \tag{2.93}$$

However, it was shown in Chang and Ji (2006a, b) that signature subspace-weighted LSMA was essentially identical to unweighted LSMA. Its results are actually the same as WAC-LSMA with $\mathbf{A} = \mathbf{I}$.

2.5.6 *Note on Kernelization*

The key idea of kernelization is the use of the kernel trick which has not been really clarified in many reported efforts when it is used. As a consequence, confusion always occurs. As an example, Support Vector Machine (SVM) was originally developed as a binary classifier to find a hyperplane maximizing the distance between support vectors in two separate classes and has nothing to do with kernelization. The introduction of kernelization into SVM allows SVM to make nonlinear boundary decisions in a feature space to resolve linear non-separability problems. Unfortunately, SVM has been somewhat over-used in the literature where the used SVM is indeed kernel-based SVM (KSVM), not SVM without using kernel. In general, there are three ways to use the kernel trick to perform kernelization. One is to kernelize a transformation to map all data samples in a high dimensional feature space. Examples of this type include kernel-based PCA (Scholkopf et al. 1999a, b) and kernel-based RX detection (Kwon and Nasrabadi 2005b), both of which require

all data samples for kernelization of the sample covariance matrix. The advantage of this approach is that no prior knowledge is needed because all data sample vectors are involved for kernelization. However, its major disadvantage is computational complexity which is exceedingly expensive. Unfortunately, most reported research efforts along this line have avoided addressing this issue. To mitigate this dilemma, an alternative approach is to use specifically selected data sample vectors, referred to as training sample vectors, to perform kernelization such as kernel-based FLDA and KSVM. Consequently, the computation complexity is reduced to that required to perform kernelization only on these selected sample vectors. However, this also comes at the cost of judiciously selecting data sample vectors for kernelization. This is a very challenging issue. A third approach is that proposed in this chapter which kernelizes a classifier/detector instead of training samples such as kernel OSP (Kwon and Nasrabadi 2005b). More specifically, the KLSMA only kernelizes the signatures used by OSP, LSOSP, NCLS, and FCLS as a mixed pixel detector/classifier to perform spectral unmixing where these signatures are provided a priori, not necessarily training sample vectors. This is quite different from existing kernel-based techniques as described above, which operate data sample vectors or training sample vectors to be mapped into a high dimensional feature space. More importantly, the computational saving is tremendous because the number of signatures kernelized by KLSMA is generally very small compared to other kernel-based methods, which require a large number of training sample vectors such as SVM or FLDA or the entire data samples such as PCA.

2.6 Unsupervised LSMA

So far, the techniques developed for LSMA are managed in the sense that the knowledge about the signature matrix $\mathbf{M}$ in (2.1) is assumed to be known and provided a priori. In practical applications, obtaining such prior knowledge may be either very costly or impossible because many unknown material substances can be uncovered by very high spectral resolution via hundreds of contiguous spectral bands but cannot be identified by ground truth or visual inspection. Under such circumstance, one best way to perform LSMA is to obtain such knowledge directly from the data to be processed. However, this is easier said than done because two challenging issues need to be addressed. One is determining the number of signatures used in the signature matrix $\mathbf{M}$, p. Once the value of p is determined, a follow-up task is how to find a set of desired p signatures to perform LSMA. Regarding the first issue, an approach recently proposed is the use of Virtual Dimensionality (VD) to estimate the value of p. As for the second issue, two approaches, LS-based (Chang 2003) and component analysis-based (Chang et al. 2011a) were developed, each of which is described below.

2.6.1 Least Squares-Based Approaches for Finding Signatures

In this section, we present three LS-based algorithms for finding spectral targets of interest using a posteriori knowledge which can be obtained directly from the data.

As noted above, two main issues are investigated. One is what are material substances of interest? Once targets of interest are defined, the next issue is how to find these targets directly from the data in an unsupervised manner without prior knowledge. To address the first issue, we first explore a new concept of so-called "*spectral*" targets which is developed to differentiate targets commonly addressed in traditional image processing. With no spectral bands used in traditional image processing, the targets of interest are generally identified by their spatial properties such as size, shape, and texture. In this case, targets to be recognized based on their spatial properties can be considered as "*spatial*" targets and the techniques developed to recognize such spatial targets are referred to as spatial domain-based image processing techniques. On the other hand, because of the use of spectral bands specified by a range of wavelengths, a multispectral or hyperspectral image pixel is actually a column vector, of which a pixel of one spectral band is produced by a particular wavelength. As a consequence, a single image pixel vector of a hyperspectral image already contains abundant spectral information provided by hundreds of contiguous spectral bands which can be used for data exploitation. Such spectral information within a single image pixel vector is referred to as intra-pixel spectral information. A target analyzed based on its spectral properties characterized by the intra-pixel spectral information on a single image pixel vector basis is called a "*spectral target*" as opposed to a "*spatial target*" analyzed by inter-pixel spatial information provided by spatial correlation among sample pixels. More specifically, three major types of spectral targets are of particular interest here. One type is endmembers, whose spectral signatures are idealistically pure (Schowengerdt 1997). Endmembers do not usually appear in multispectral images because of low spatial and spectral resolution, but have become increasingly important in hyperspectral imaging because an endmember can be used to identify a spectral class. Another type is subpixel targets which do not fully occupy a pixel but rather are completely embedded in a single image pixel vector. This type of target cannot be visualized spatially and can only be recognized by their spectral properties. Subpixel targets occur when their spatial extent is smaller than pixel resolution. A third type of targets is mixed targets whose spectral signatures are linearly or nonlinearly mixed by a number of target spectral signatures with appropriate portions of fractions present in a single image pixel vector. The occurrence of a mixed target is a result of low spatial and spectral resolution and it may partially occupy more than one pixel vector. Apparently, none of these three types of spectral targets can be effectively analyzed by spatial domain-based techniques.

With a spectral target defined as above, what we are particularly interested in here from an aspect of statistical signal processing are two types of spectral targets, one characterized by second-order sample intra-pixel Spectral Information Statistics (SIS) and the other by sample intra-pixel SIS of order higher than 2, referred to as high-order SIS. It should be noted that the term of sample intra-pixel SIS is defined as correlation of intra-pixel SIS among sample vectors. In the context of sample SIS, we assume that background (BKG) pixels are those spectral targets characterized by second sample intra-pixel SIS while the target pixels of interest are those high-order sample intra-pixel SIS. In hyperspectral image analysis this seems a reasonable assumption since the spectral targets of interest in hyperspectral data exploitation are those which either (1) occur with low probability or (2) have small populations when they are present. In other words, these types of spectral target are usually relatively small, appear in small population, and also occur with low probabilities, e.g., special spices in agriculture and ecology, toxic wastes in environmental monitoring, rare minerals in geology, drug/smuggler trafficking in law enforcement, combat vehicles in the battlefield, landmines in war zones, chemical/biological agents in bioterrorism, weapon concealment, and mass graves. These spectral targets are generally considered as insignificant objects because of their very limited spatial information, but they are actually critical and crucial for defense and intelligence analysis because they are insignificant compared to targets with large sample pools and generally hard to identify by visual inspection. From a statistical point of view, the SIS of such special targets cannot be captured by second-order sample intra-pixel SIS but rather by high-order sample intra-pixel SIS.

Once image pixel vectors are categorized into BKG and target classes according to sample intra-pixel SIS, a follow-up task is how to find them, in which case two issues need to be addressed. One is how many of them. The other is how to find them. The first issue can be resolved by a new concept, VD which is recently developed (Chang 2003; Chang and Du 2004). The idea of VD is based on the assumption that if a signal source is presenting in the data, it will contribute energy to first-order statistics. In doing so, both the eigenvalues of sample correlation matrix, $\left\{\hat{\lambda}_l\right\}$, and the eigenvalues of covariance matrix, $\{\lambda_l\}$, are calculated. If $\hat{\lambda}_l - \lambda_l$ is greater than zero resulting from the lth component sample mean, it implies that there is a signal source; otherwise, no signal is present. To materialize this idea, a binary composite hypothesis testing problem is formulated in such a way that the null and alternative hypotheses, H_0 and H_1, represent two scenarios, $H_0 : \hat{\lambda}_l - \lambda_l = 0$ and $H_1 : \hat{\lambda}_l - \lambda_l > 0$, respectively. The Neyman–Pearson detection theory (Poor 1994) is then applied to find how many times the test fails run over all spectral bands for a given false alarm probability, P_{F}. It is the number of test failures which indicates the number of signal sources assumed to be in the data. The beauty of VD lies in the fact that its value is completely determined by the P_{F}. By varying the value of P_{F}, the number of spectrally distinct signatures estimated by the VD varies. For example, if

P_F is set low, fewer tests will fail and thus fewer targets are assumed to be in the data, and vice versa. To address the second issue, an unsupervised spectral target finding algorithm (USTFA) is developed which is based on three LS-based algorithms, Automatic Target Generation Process (ATGP) (Ren and Chang 2003), Unsupervised Non-negativity Constrained Least Squares (UNCLS) method (Chang and Heinz 2000), and Unsupervised Fully Constrained Least Squares method (UFCLS) (Heinz and Chang 2001). For these unsupervised methods to extract and distinguish spectral targets of second-order sample intra-pixel SIS from high-order sample intra-pixel SIS, it assumes that the BKG in a hyperspectral image is most likely characterized by second-order sample intra-pixel SIS while hyperspectral targets will be more likely to be captured by high-order sample intra-pixel SIS as outliners because of their small spatial presence. In this case, high-order spectral targets are referred to as desired targets to be used for image analysis, while second-order spectral targets are considered as undesired targets for which we would like to annihilate or suppress prior to data processing to improve image analysis.

To accomplish this task, a designed LS-based algorithm is first applied to the original data to extract data sample vectors characterized by second-order SIS. Then, the same algorithm is further applied to the data which is sphered by removing the data sample mean and co-variances while making data variances ones so that all second-order SIS-characterized data samples will be on the sphere and all other data sample vectors that are characterized by high-order SIS are either inside (sub-Gaussian samples) or outside the sphere (super-Gaussian samples). As a consequence of such a sphering process, the resulting data has the first- and second-order SIS removed from the original data because of zero mean and constant variance one so that samples characterized by SIS of orders higher than second can be extracted from inside or outside the sphere. Interestingly, despite the fact that the idea of using the same algorithm in two passes—one pass for the original data and another pass for the sphered data to extract two types of targets of interest, second-order targets and high-order targets for data analysis—seems simple, it is by no means a trivial matter because its novelty has never been explored in the open literature. In what follows, we design and develop three LS-based algorithms for this purpose.

2.6.1.1 Automatic Target Generation Process (ATGP)

The first LS-based algorithm is ATGP which can be considered as an unsupervised and unconstrained OSP technique. It performs a succession of OSPs specified by (2.5) to find a set of sequential data sample vectors that represents targets of interest as follows.

OSP Target Generation Process

1. Initial condition:
   Let ε be a prescribed error threshold and $\mathbf{t}_0$ be a data sample vector with maximal vector length, i.e., $\mathbf{t}_0 = \arg\{\max_{\mathbf{r}} ||\mathbf{r}||\}$ where $\mathbf{r}$ is a data sample vector and $||\mathbf{r}||^2 = \mathbf{r}^{\mathrm{T}}\mathbf{r}$. Set $k = 0$ and use (2.5) to define $P^{\perp}_{\mathbf{t}_0} = \mathbf{I} - \mathbf{t}_0\left(\mathbf{t}_0^{\mathrm{T}}\mathbf{t}_0\right)^{-1}\mathbf{t}_0^{\mathrm{T}}$ with $\mathbf{U}_0 = [\mathbf{t}_0]$.
2. Let $k \leftarrow k+1$ and apply $P^{\perp}_{\mathbf{U}_{k-1}} = \mathbf{I} - \mathbf{U}_{k-1}\left(\mathbf{U}_{k-1}^{\mathrm{T}}\mathbf{U}_{k-1}\right)^{-1}\mathbf{U}_{k-1}^{\mathrm{T}}$ via (2.5) with $\mathbf{U}_{k-1} = [\mathbf{t}_0\mathbf{t}_1 \ldots \mathbf{t}_{k-1}]$ to all image pixels $\mathbf{r}$ in the image and find the kth target $\mathbf{t}_k$ generated at the kth stage which has the maximum orthogonal projection as follows:

$$\mathbf{t}_k = \arg\left\{\max_{\mathbf{r}}\left[\left(P^{\perp}_{[\mathbf{U}_{k-1}\mathbf{t}_k]}\mathbf{r}\right)^{\mathrm{T}}\left(P^{\perp}_{[\mathbf{U}_{k-1}\mathbf{t}_k]}\mathbf{r}\right)\right]\right\}. \tag{2.94}$$

3. If $m_{\mathrm{OSP}}(\mathbf{t}_k) > \varepsilon$ where $m_{\mathrm{OSP}}(\cdot)$ is a well-defined OSP measure, then go to step 2. Otherwise, the algorithm is terminated. At this point, all the generated target pixels $\mathbf{t}_0, \mathbf{t}_1, \ldots, \mathbf{t}_{k-1}$ are considered as the desired targets.

Although different criteria can be used to define $m_{\mathrm{OSP}}(\cdot)$ in step 3, there are at least two ways to do so. One is defined by Ren and Chang (2003) as Orthogonal Projection Correlation Index (OPCI) given by

$$m_{\mathrm{OSP}}(\mathbf{t}_k) \equiv \mathrm{OPCI}(\mathbf{t}_k) = \mathbf{t}_0^{\mathrm{T}} P^{\perp}_{\mathbf{U}_k}\mathbf{t}_0 > \varepsilon. \tag{2.95}$$

The above OSP-based Target Generation Process is called ATDCA developed by Ren and Chang (2003). Another is defined by the popular ATGP in Chang (2003) as

$$m_{\mathrm{OSP}}(\mathbf{t}_k) \equiv m_{\mathrm{ATGP}}(\mathbf{t}_k) \equiv \mathbf{t}_k^{\mathrm{T}} P^{\perp}_{\mathbf{U}_{k-1}}\mathbf{t}_k > \varepsilon \tag{2.96}$$

In this case, the resulting OSP-based Target Generation Process is called ATGP.

2.6.1.2 Unsupervised Non-negativity Least Squares (UNCLS) Method

A second LS-based algorithm is UNCLS which is an unsupervised version of the abundance NCLS where the NCLS is a partially abundance-constrained least squares-based technique that imposes the ANC with $\boldsymbol{\alpha} \geq \mathbf{0}$, i.e., $\alpha_j \geq 0$ for all on the linear mixing model specified by (2.1). It can be implemented in the following.

UNCLS

1. Initial condition:
   Select ε to be a prescribed error threshold and let $\mathbf{t}_0 = \arg\{\max_{\mathbf{r}}[\mathbf{r}^{\mathrm{T}}\mathbf{r}]\}$ where $\mathbf{r}$ is run over all image pixel vectors, and set $k = 0$.

2. Let $\mathrm{LSE}^{(0)}(\mathbf{r}) = \left(\mathbf{r} - \hat{\alpha}_0^{(1)}(\mathbf{r})\mathbf{t}_0\right)^{\mathrm{T}}\left(\mathbf{r} - \hat{\alpha}_0^{(1)}(\mathbf{r})\mathbf{t}_0\right)$ and check if $\max_{\mathbf{r}} \mathrm{LSE}^{(0)}(\mathbf{r}) < \varepsilon$. If yes, the algorithm is terminated; otherwise continue.
3. Let $k \leftarrow k+1$ and find $\mathbf{t}_k = \arg\left\{\max_{\mathbf{r}}\left[\mathrm{LSE}^{(k-1)}(\mathbf{r})\right]\right\}$.
4. Apply the NCLS method with the signature matrix $\mathbf{M}^{(k)} = [\mathbf{t}_0\mathbf{t}_1 \ldots \mathbf{t}_{k-1}]$ to estimate the abundance fraction of $\mathbf{t}_0, \mathbf{t}_1, \ldots, \mathbf{t}_{k-1}$, $\hat{\alpha}_1^{(k)}(\mathbf{r}), \hat{\alpha}_2^{(k)}(\mathbf{r}), \ldots, \hat{\alpha}_{k-1}^{(k)}(\mathbf{r})$.
5. Find the kth maximum LSE defined by

$$\max_{\mathbf{r}}\left\{\mathrm{LSE}^{(k)}(\mathbf{r})\right\} = \max_{\mathbf{r}}\left\{\left(\mathbf{r} - \sum_{j=1}^{k-1}\hat{\alpha}_j^{(k)}\mathbf{t}_j\right)^{\mathrm{T}}\left(\mathbf{r} - \sum_{j=1}^{k-1}\hat{\alpha}_j^{(k)}\mathbf{t}_j\right)\right\} \tag{2.97}$$

6. If $\max_{\mathbf{r}}\mathrm{LSE}^{(k-1)}(\mathbf{r}) < \varepsilon$, the algorithm is terminated; otherwise go to step 3.

2.6.1.3 Unsupervised Fully Least Squares (UFCLS) Method

A third LS-based algorithm is UFCLS which is an unsupervised abundance Fully Constrained Least Squares (FCLS) where the FCLS is a fully abundance-constrained least squares-based technique that imposes both ASC, i.e., $\sum_{j=1}^{p}\alpha_j = 1$ and Abundance Non-negativity Constraint (ANC), $\boldsymbol{\alpha} \geq \mathbf{0}$, i.e., $\alpha_j \geq 0$ on the linear mixing model (2.1). Its implementation is provided below.

UFCLS

1. Initial condition:
   Select ε to be a prescribed error threshold and let $\mathbf{t}_0 = \arg\{\max_{\mathbf{r}}[\mathbf{r}^{\mathrm{T}}\mathbf{r}]\}$ where $\mathbf{r}$ is run over all image pixel vectors, and let $k = 0$.
2. Let $\mathrm{LSE}^{(0)}(\mathbf{r}) = \left(\mathbf{r} - \hat{\alpha}_0^{(1)}(\mathbf{r})\mathbf{t}_0\right)^{\mathrm{T}}\left(\mathbf{r} - \hat{\alpha}_0^{(1)}(\mathbf{r})\mathbf{t}_0\right)$ and check if $\max_{\mathbf{r}} \mathrm{LSE}^{(0)}(\mathbf{r}) < \varepsilon$. If yes, the algorithm is terminated; otherwise continue.
3. Let $k \leftarrow k+1$ and find $\mathbf{t}_k = \arg\left\{\max_{\mathbf{r}}\left[\mathrm{LSE}^{(k-1)}(\mathbf{r})\right]\right\}$.
4. Apply the FCLS method with the signature matrix $\mathbf{M}^{(k)} = [\mathbf{t}_0\mathbf{t}_1 \ldots \mathbf{t}_{k-1}]$ to estimate the abundance fraction of $\mathbf{t}_0, \mathbf{t}_1, \ldots, \mathbf{t}_{k-1}$, $\hat{\alpha}_1^{(k)}(\mathbf{r}), \hat{\alpha}_2^{(k)}(\mathbf{r}), \ldots, \hat{\alpha}_{k-1}^{(k)}(\mathbf{r})$.
5. Find the kth maximum LSE defined by

$$\max_{\mathbf{r}}\left\{\mathrm{LSE}^{(k)}(\mathbf{r})\right\} = \max_{\mathbf{r}}\left\{\left(\mathbf{r} - \sum_{j=1}^{k-1}\hat{\alpha}_j^{(k)}\mathbf{t}_j\right)^{\mathrm{T}}\left(\mathbf{r} - \sum_{j=1}^{k-1}\hat{\alpha}_j^{(k)}\mathbf{t}_j\right)\right\} \tag{2.98}$$

If $\max_{\mathbf{r}} \mathrm{LSE}^{(k-1)}(\mathbf{r}) < \varepsilon$, the algorithm is terminated; otherwise go to step 3.

When the above-mentioned three unsupervised LS-based algorithms are implemented, a prescribed error ε which is determined by various applications is required to terminate the algorithms. In general, it is done by visual inspection on a trial-and-error basis and is not practical for our purpose. Therefore, instead of using ε as a stopping rule, we use the VD as an alternative rule to determine how many targets are required for our designed LS algorithms to generate.

For the proposed LS-based algorithms to be successful, we assume that the most image BKG is characterized by a very large number of uninteresting data sample vectors which can be characterized by second-order statistics as opposed to target pixels which can be captured by high-order statistics because of a small number of target pixels. By means of this assumption we can consider two sets of data for processing. One is the original data and the other is the sphered data which has the mean and covariance removed from the original data for consideration. We then apply the three unsupervised LS-based algorithms to these two data sets to extract second-order BKG pixels as well as high-order target pixels. However, if a sample pixel shows strong signal statistics in both original and sphered data sets, it is considered as a target pixel and can be removed from the BKG category.

A detailed implementation of an LS-based Unsupervised Spectral Target Finding Algorithm (USTFA) can be briefly described as follows where the LS-based unsupervised algorithm used in the USTFA can be one of the three LS unsupervised algorithms described in Sect. 2.3.

LS-Based Unsupervised Spectral Target Finding Algorithm

1. Apply VD on the image data to determine the number of targets required to be generated, n_{VD}.
2. Apply an LS-based algorithm to the original image data and find n_{VD} BKG pixels $S^{\mathrm{BKG}} = \left\{\mathbf{b}_j^{\mathrm{LS}}\right\}_{j=1}^{n_{\mathrm{VD}}}$.
3. Apply the LS-based algorithm to the sphered data and find n_{VD} target pixels, $S^{\mathrm{target}} = \left\{\mathbf{t}_j^{\mathrm{LS}}\right\}_{j=1}^{n_{\mathrm{VD}}}$.
4. Since there may be some pixels in S^{BKG} whose spectra are very close to those also showing up in S^{target}, a spectral measure such as Spectral Angle Mapper (Chang 2003) is applied to extract these pixels which will be removed from S^{BKG}. Let the resulting BKG sample set be denoted by $\tilde{S}^{\mathrm{BKG}} = \left\{\tilde{\mathbf{b}}_i^{\mathrm{LS}}\right\}_{i=1}^{n_{\mathrm{BKG}}}$ where n_{BKG} is the total number of remaining BKG pixels in S^{BKG} after the pixels in $S^{\mathrm{target}} \cap S^{\mathrm{BKG}}$ are removed.
5. Form a signature matrix $\mathbf{M}$ by merging $\tilde{S}^{\mathrm{BKG}}$ and S^{target}, i.e., finding pixels in $\left\{\tilde{\mathbf{b}}_i^{\mathrm{LS}}\right\}_{i=1}^{n_{\mathrm{BKG}}} \cup \left\{\mathbf{t}_j^{\mathrm{LS}}\right\}_{j=1}^{n_{\mathrm{VD}}}$. It should be noted that the number of pixels in $\mathbf{M}$ is between n_{VD} and $2n_{\mathrm{VD}}$, i.e., $n_{\mathrm{VD}} \le n_{\mathrm{VD}} + n_{\mathrm{BKG}} \le 2n_{\mathrm{VD}}$.

Apply an LS method such as abundance-unconstrained classifier LSOSP, abundance non-negativity constrained classifier NCLS and abundance fully constrained classifier FCLS to perform mixed pixel classification where only the target pixels in S^{target} are classified by their corresponding abundance fractions while the target pixels in $\tilde{S}^{\text{BKG}} = \left\{\tilde{\mathbf{b}}_i^{\text{LS}}\right\}_{i=1}^{n_{\text{BKG}}}$ is used for BKG suppression. It should be noted that for LSMA to perform pure-pixel classification we need a value to threshold the LSMA-estimated abundance fractions of each of targets for making hard decisions. In this case, finding an appropriate threshold value is generally very challenging. In our experiments only LSMA is performed to produce abundance fraction estimates for target pixels. So, when a specific LS algorithm is used in USTFA, the superscript "LS" in the above algorithm will be replaced with this particular algorithm. For example, if ATGP is used for the USTFA, it is then called ATGP-USTFA.

2.6.2 *Component Analysis-Based Approaches for Finding Signatures*

Since all the three algorithms described in Sect. 2.6.1 are second order spectral statistics-based techniques, they may not be effective in extracting signatures characterized by the high order of statistics-IBSI(S^{target}). To solve this problem, the data was sphered so that the signatures characterized by second-order-IBSI(S^{BKG}) can be removed prior to application of these algorithms. Interestingly, such an idea can be materialized and implemented by component-based transformation without appealing for any algorithm.

In Sect. 2.6.1, hyperspectral signatures are assumed to be categorized into BKG signatures characterized by second-order-IBSI(S^{BKG}) and target signatures characterized by the high order of statistics-IBSI(S^{target}). To realize this idea, we use PCA to produce a set of PCs to represent the data where eigenvectors are projection vectors to specify PCs with eigenvalues specified by data variances. In this case, PCA can be used to extract BKG signatures characterized by second-order-IBSI (S^{BKG}) in the first few PCs. On the other hand, we also use ICA to find desired target signatures characterized by the high order of statistics-IBSI(S^{target}) in the first few ICs. In both cases, the VD is used to estimate how many PCs and ICs are required to used for signature extraction. Because PCs and ICs are obtained by mapping all data samples onto the projection vectors, the projection values of data samples are real values. So, an issue arises—"how many data samples should be selected from each PC and each IC?" Two sample values in each IC are of major interest, one with maximal projection value and the other with minimal projection value. These two samples represent maximal projections in two opposite directions of a projection vector that specifies an IC. Both indicate their importance in data analysis. This idea was previously explored in development of pixel purity index in Boardman (1994) where the most likely endmembers are those sample vectors with either maximal or minimal projections on each randomly generated vector referred

to as skewers. All such data sample vectors extracted from PCs and ICs are called Virtual Signatures (VSs). However, it is worth noting that there is no similar selection of sample vectors with minimal projections in PCs because PCA is a transformation of second order statistics with variance representing signal energy no less than zero. This is because the samples with minimal projections, i.e., variances, are supposed to be noisy sample vectors. Accordingly, selecting data sample vectors with minimal projections in PCs is not appropriate. However, with sphering performed by ICA, the noisy sample vectors with small variances have been removed. As a result, it makes sense to select data samples with minimal projections in ICs as desired samples.

Using VD to determine the numbers of PCs and ICs and selecting data sample vectors with maximal and minimal projections of ICs, a CA-based Unsupervised Virtual Signature Finding Algorithm (CA-UVSFA) can be described as follows.

CA-UVSFA

1. Find the VD, which determines the number of components required to generate, p.
2. Apply the PCA to the original image data to find p PCs and extract the brightest pixels (i.e., data samples with maximal projections in PCs) as a VS from each of p PCs to form $S^{\mathrm{BKG}} = \left\{\mathbf{b}_j^{\mathrm{PCA}}\right\}_{j=1}^{p}$.
3. Apply the ICA to the sphered data to find p ICs and extract points with the maximal and minimal projections from each of p ICs as VSs to form $S^{\mathrm{target}} = \left\{\mathbf{t}_j^{\mathrm{ICA}}\right\}_{j=1}^{2p}$. Because some pixels with minimal projections may close to certain pixels with maximal projections in other ICs, a spectral measure such as SAM (Chang 2003) is applied to identify these pixels and eliminate them from S^{target} to form $\tilde{S}^{\mathrm{target}} = \left\{\tilde{\mathbf{t}}_j^{\mathrm{ICA}}\right\}$. It should be noted that the ICA used here is the one developed in Hyvarinen and Oja (2000), called FastICA, which is actually based on a criterion derived from a combination of the third- and fourth-order statistics (Hyvarinen et al. 2001, (5.35), p. 115).
4. Furthermore, there may be some target signatures extracted in $\tilde{S}^{\mathrm{target}} = \left\{\tilde{\mathbf{t}}_j^{\mathrm{ICA}}\right\}$, which also exhibit strong energies in the PCs where pixels corresponding to these signatures may also be extracted in S^{BKG}. In this case, the SAM is used to extract these pixels and remove them from S^{BKG}. Let the remaining BKG sample set be denoted by $\tilde{S}^{\mathrm{BKG}} = \left\{\tilde{\mathbf{b}}_j^{\mathrm{PCA}}\right\}$.
5. Construct a set of VSs, $S^{\mathrm{VS}} = \left\{\tilde{\mathbf{b}}_j^{\mathrm{PCA}}\right\} \cup \left\{\tilde{\mathbf{t}}_j^{\mathrm{ICA}}\right\}$, by merging $\tilde{S}^{\mathrm{BKG}}$ and $\tilde{S}^{\mathrm{target}}$ for spectral unmixing. It should be noted that the number of VSs in S^{VS} is between p and $3p$.

A comment on why each PCA and ICA is required to generate p components is noteworthy. It is often the case that target signatures may show up in either PCs or

ICs, but not both. To ensure that it doesn't matter which scenario will occur, using p PCs and p ICs should be able to capture all these target signatures.

Determining the number of components and finding VSs are two key steps in implementing the CA-UVSFA. With help of CA-UVSFA we can develop an algorithm, referred to as CA-Unsupervised LSMA (CA-ULSMA), as follows.

CA-ULSMA

1. Use the HFC/NWHFC methods to determine the VD for the number of components to be generated, p.
2. Apply the CA-UVSFA to produce a set of VSs, S^{VS}.
3. Apply an SLSMA technique such as least squares OSP (unconstrained classifier), non-negativity constrained least squares (NCLS) (partially abundance constrained classifier), and FCLS (fully abundance-constrained classifier) to perform LSU where only the target pixels in $\tilde{S}^{\mathrm{target}}$ will be unmixed, while the target pixels in $\tilde{S}^{\mathrm{BKG}}$ will be used for BKG suppression.

In step 1 of CA-ULSMA, for the HFC/NWHFC method to work effectively, the spectrally distinct signatures defined by the VD are those which do not have significant contributions to data variances. Such signatures are generally characterized by three unique features. First, the probabilities of their occurrence are usually low. Second, when such signatures are present, there are not too many sample vectors. Third, as a result, the variances of such signatures are generally very small and can be considered to be negligible. So, when the HFC/NWHFC methods are used to estimate the VD, which determines the number of VSs used by the LSMA, two assumptions are made on the VSs. The first and foremost assumption is that all the VSs in S^{VE} are assumed to have the least inter-sample spectral correlation, more specifically, least IBSI. This is a reasonable assumption since different VSs should have the least spectral correlation among all the data sample vectors to allow VSs as distinct as possible in terms of spectral characteristics via IBSI. Another is that the number of data sample vectors specified by a VS must be relatively small because they represent most spectrally distinctive signatures. With these two assumptions in mind, a VS is only contributed to the sample mean but not the variance of each spectral band. This is also the key idea used to develop the HFC/NWHFC methods. It also explains why the HFC/NWHFC-estimated VD works very effectively for HYperspectral Digital Imagery Collection Experiment (HYDICE) data in Fig. 1.9, where the number of target panel pixels specified by each of five panel signatures is very small. When these two assumptions are violated, the HFC/NWHFC-estimated VD may not be accurate. So, when the HFC/NWHFC methods are used to estimate the VD, users should be aware of these assumptions.

2.7 Conclusions

LSMA has great potential in hyperspectral data exploitation. Its major application in the literature has been LSU, to be discussed in Chap. 4. However, other applications have recently received considerable interest. For example, LSMA can be used to find endmembers in Chaps. 9 and 13. It can also be used for data compression (Du and Chang 2004a, b). In addition, it can be shown that LSMA is very closely related to OSP and Simplex Volume Analysis (SGA) in Chap. 4.

References

Boardman, J.W. 1994. Geometric mixture analysis of imaging spectrometry data. *International geoscience remote sensing symposium*, vol. 4, 2369–2371.

Broadwater, J., J.R. Chellappa, A. Banerjee, and P. Burlina. 2007. Kernel fully constrained least squares abundance estimates. *2007 IEEE international geoscience and remote sensing symposium* (IGARSS), 4041–4044, 23–28 July, 2007.

Chang, C.-I 2002a. Relationship among orthogonal subspace projection, constrained energy minimization and RX-algorithm. In *SPIE conference on algorithms and technologies for multispectral, hyperspectral and ultraspectral imagery VIII*, SPIE 4725, Orlando, Florida, April 1–5, 2002.

Chang, C.-I 2002b. Target signature-constrained mixed pixel classification for hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* 40(2): 1065–1081.

Chang, C.-I 2003. *Hyperspectral imaging: techniques for spectral detection and classification*. Dordrecht, the Netherlands: Kluwer Academic/Plenum Publishers.

Chang, C.-I 2005. Orthogonal subspace projection revisited: a comprehensive study and analysis. *IEEE Transactions on Geoscience and Remote Sensing* 43(3): 502–518.

Chang, C.-I 2013. *Hyperspectral data processing: algorithm design and analysis*. New Jersey: Wiley. 2013.

Chang, C.-I, and D. Heinz. 2000. Constrained subpixel detection for remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing* 38(3): 1144–1159.

Chang, C.-I, and Q. Du. 2004. Estimation of number of spectrally distinct signal sources in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* 42(3): 608–619.

Chang, C.-I, and B. Ji. 2006a. Weighted least squares error approaches to abundance-constrained linear spectral mixture analysis. *IEEE Transactions on Geoscience and Remote Sensing* 44(2): 378–388.

Chang, C.-I, and B. Ji. 2006b. Fisher's linear spectral mixture analysis. *IEEE Transactions on Geoscience and Remote Sensing* 44(8): 2292–2304

Chang, C.-I, T.-L.E. Sun, and M.L.G. Althouse. 1998a. An unsupervised interference rejection approach to target detection and classification for hyperspectral imagery. *Optical Engineering* 37(3): 735–743.

Chang, C.-I, X. Zhao, M.L.G. Althouse, and J.-J. Pan. 1998b. Least squares subspace projection approach to mixed pixel classification in hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing* 36(3): 898–912.

Chang, C.-I, X. Jiao, Y. Du, and M.-L. Chang. 2010a. A review of unsupervised hyperspectral target analysis. *EURASIP Journal on Advanced in Signal Processing* 2010, Article ID 503752, 26 pages. doi:10.1155/2010/503752.

Chang, C.-I, S. Chakravarty, and C.-S. Lo. 2010b. Spectral feature probabilistic coding for hyperspectral signatures. *IEEE Sensors Journal* 10(3): 395–409.

Chang, C.-I, C.-C. Wu, and C.-T. Tsai. 2011a. Random N-finder algorithm. *IEEE Transactions on Image Processing* 20(3): 641–656.

Chang, C.-I, and H. Safavi. 2011b. Progressive dimensionality reduction by transform for hyperspectral image analysis. *Pattern Recognition*. April 30, 2011, online access: doi:10.1016/j.patcog.2011.03.030.

Du, Q., and C.-I Chang. 2004a. Linear mixture analysis-based compression for hyperspectral image analysis. *IEEE Transactions on Geoscience and Remote Sensing* 42(4): 875–891.

Du, Q., and C.-I Chang. 2004b. A signal-decomposed and interference-annihilated approach to hyperspectral target detection. *IEEE Transactions on Geoscience and Remote Sensing* 42(4): 892–906.

Duda, R.O., and P.E. Hart. 1973. *Pattern classification and scene analysis*. New York, NY: Wiley.

Frost III, O.L. 1972. An algorithm for linearly constrained adaptive array processing. *Proceedings of the IEEE* 60: 926–935.

Harsanyi, J.C., and C.-I Chang. 1994. Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *IEEE Transactions on Geoscience and Remote Sensing* 32(4): 779–785.

Heinz, D., and C.-I Chang. 2001. Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* 39(3): 529–545.

Hyvarinen, A., and E. Oja. 2000. Independent component analysis: algorithms and applications. *Neural Networks* 13: 411–430.

Hyvarinen, A., J. Karhunen, and E. Oja. 2001. *Independent component analysis*. New York: Wiley.

Kwon, H., and N.M. Nasrabadi. 2005b. Kernel orthogonal subspace projection for hyperspectral signal classification. *IEEE Transactions on Geoscience and Remote Sensing* 43(12): 2952–2962.

Liu, K. 2011. *Progressive band prioritization and selection for linear spectral mixture analysis*, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, Baltimore, MD.

Liu, K., and C.-I Chang. 2011. Dynamic band selection for hyperspectral imagery. *International geoscience and remote sensing symposium*, 24–29 July, Vancouver, Canada.

Liu, K., E. Wong, Y. Du, C.C.C. Chen, and C.-I Chang. 2012. Kernel-based linear spectral mixture analysis. *IEEE Geoscinece and Remote Sensinbg Letters* 9(1): 129–133.

Poor, H.V. 1994. *An introduction to detection and estimation theory*, 2nd ed. New York: Springer.

Ren, H., and C.-I Chang. 1999. A constrained least squares approach to hyperspectral image classification. In *1999 conference on information science and systems*, 551–556, Johns Hopkins University, Baltimore, MD, March 17–19, 1999.

Ren, H., and C.-I Chang. 2003. Automatic spectral target recognition in hyperspectral imagery. *IEEE Transactions on Aerospace and Electronic Systems* 39(4): 1232–1249.

Scholkopf, N., A.J. Smola, and K.-R. Miller. 1999a. Kernel principal component analysis. In *Advances in Kernel Methods* by N. Scholkopf, C.J.C. Burges, and A.J. Smola. *Learning with Kernels*. Cambridge MA: MIT Press.

Scholkopf, N., A.J. Smola, and K.-R. Miller. 1999b. Kernel principal component analysis. *Neural Computation* 10: 1299–1319

Schowengerdt, R.A. 1997. *Remote sensing: models and methods for image processing*, 2nd ed. New York: Academic Press.

Settle, J.J. 1996. On the relationship between spectral unmixing and subspace projection. *IEEE Transactions on Geoscience and Remote Sensing* 34(4): 1045–1046.

Tu, T.M., C.-H. Chen, and C.-I Chang. 1997. A posteriori least squares orthogonal subspace projection approach to weak signature extraction and detection. *IEEE Transactions on Geoscience and Remote Sensing* 35(1): 127–139.

http://www.springer.com/978-1-4419-6186-0

Real-Time Progressive Hyperspectral Image Processing
Endmember Finding and Anomaly Detection
Chang, C.-I.
2016, XXIII, 623 p. 331 illus., 256 illus. in color.,
Hardcover
ISBN: 978-1-4419-6186-0