Mini-projet: séparation de sources hyperspectrale

Table of content

Hyperspectral imaging

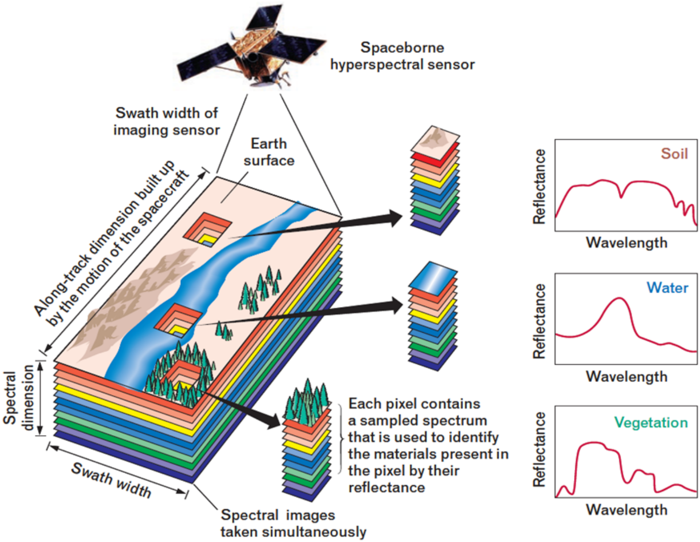

Figure: Schematic view of hyperspectral imaging for remote sensing applications

Hyperspectral unmixing

Due to spatial resolution, the measured spectra is usually a combination of multiple objects of interest present in the scene.

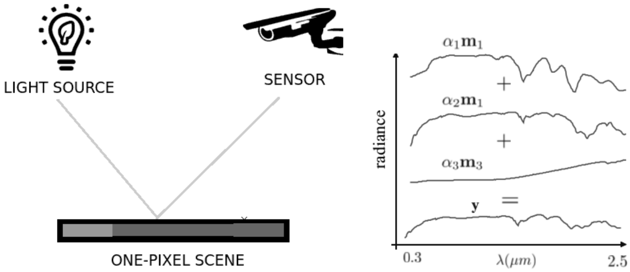

Figure: Schematic view of the mixing model

Hyperspectral unmixing algorithms try to separate the pixel spectra of an hyperspectral image into a collection of constituent spectra or **endmembers**.

Table of content

Linear mixing model

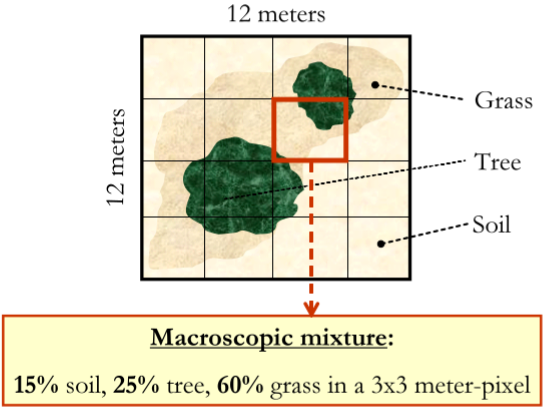

Figure: Schematic view of the linear mixing model

Linear mixing model (ii)

The spectrum y_i at pixel i is considered to be a **linear combination** of the spectra of the "pure" materials present in the scene, weighted by their respective **abundances**:

$$\mathbf{y_i} = \sum_{r=1}^{R} \alpha_{i,r} \mathbf{m_r} + \mathbf{w_i}.$$

- $\mathbf{y_i}$: vector in $\mathbb{R}^L$ containing L spectral bands.
- R: number of sources or endmembers (i.e pure materials). Typically, $R \ll L$.
- $\mathbf{m_r} \in \mathbb{R}^L$: spectrum of the r-th endmember.
- $\alpha_{i,r}$: abundance of the r-th endmember at pixel i.
- $\mathbf{w_i} \in \mathbb{R}^L$: noise vector.

Linear mixing model (iii)

Additional constraints:

- **Non-negativity constraint**

$$\forall i, \forall r, 1 \leq r \leq R, \alpha_{i,r} \geq 0.$$

The abundance of a material is non-negative.

- **Sum-to-one constraint**

$$\forall i, \sum_{r=1}^{R} \alpha_{i,r} = 1.$$

The abundances always sum to 1.

Linear mixing model (iv)

The linear mixing model can be reformulated in matrix form to yield

$$\mathbf{y} = \mathbf{M}\mathbf{A} + \mathbf{W},$$

where:

1. $\mathbf{y}$ is the L by N matrix $[y_1, .., y_N]$.
2. $\mathbf{M}$ is the L by R matrix containg the endmembers $[m_1, .., m_R]$.
3. $\mathbf{A}$ is the R by N abundance matrix containing the abundances at each location.
4. $\mathbf{W}$ is a L by N noise matrix.

Endmembers simplex

- If we assume the family of vectors $\{\mathbf{m_2} - \mathbf{m_1}, .., \mathbf{m_R} - \mathbf{m_1}\}$ to be linearly independent, then the subset

$$C := \{\mathbf{y} = \mathbf{MA} | \forall i, \forall r, 1 \leq r \leq R, \alpha_{i,r} \geq 0 \text{ and } \forall i, \sum_{r=1}^{R} \alpha_{i,r} = 1\}.$$

  is a $(R-1)$-simplex in $\mathbb{R}^L$.

- The data points lie in a $(R-1)$-simplex embedded in an affine subspace of dimension $R \ll L$ of $\mathbb{R}^L$.

Example: SAMSON dataset

The SAMSON dataset is a publicly available hyperspectral image (952 by 952 pixels image with 156 spectral bands). Each spectrum combine 3 endmembers: soil, tree and water.

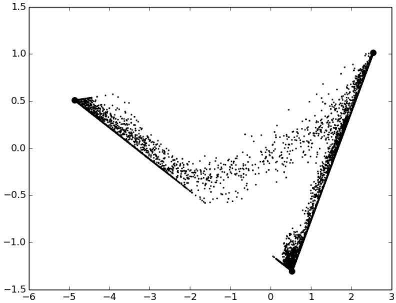

Figure: 2-Simplex identified on the SAMSON dataset by principal component analysis.

Table of content

Principal component analysis

- **Principal component analysis (PCA)** aims at determining the direction $\hat{\mathbf{u}}$ in $\mathbb{R}^L$ for which the projection of the dataset $\{y_n \in \mathbb{R}^L, 1 \leq n \leq N\}$ has the highest variance.
- Let $\mathbf{u}$ be some unitary vector in $\mathbb{R}^L$. The mean of the projected data is:

$$\mathbf{u}^T\bar{\mathbf{y}} = \sum_{n=1}^{N} \mathbf{u}^T\mathbf{y_n}.$$

- Variance of the projected data is:

$$\frac{1}{N}\sum_{n=1}^{N}(\mathbf{u}^T\mathbf{y_n} - \mathbf{u}^T\bar{\mathbf{y}})^2 = \mathbf{u}^T C\mathbf{u}.$$

where C is the covariance matrix of the original dataset.

Principal component analysis (ii)

- The variance is maximal along the unit vector $\hat{\mathbf{u}}$ solution of the optimization problem

$$\hat{\mathbf{u}} = \arg\max_{\mathbf{u}\in\mathbb{R}^L} \mathbf{u}^T C \mathbf{u}$$

subject to $\mathbf{u}^\mathbf{T}\mathbf{u} = 1$.

- The corresponding **Laplacian** is

$$L(\mathbf{u}, \lambda) = \mathbf{u}^T C \mathbf{u} + \lambda(1 - \mathbf{u}^\mathbf{T}\mathbf{u}).$$

- The **optimality conditions** are given by

$$C\mathbf{u} = \lambda \mathbf{u}, \quad \mathbf{u}^\mathbf{T}\mathbf{u} = 1.$$

- The optimal vector $\hat{\mathbf{u}}$ is therefore the eigenvector of C with the **largest eigenvalue**.

Dimensionality reduction

- Let $\mathbf{U}$ denote the projection matrix on the subspace generated by the $R-1$ largest eigenvectors of the covariance matrix C. The projected data points are

$$\mathbf{z_n} = \mathbf{U}(\mathbf{y_n} - \mathbf{m})$$

where $m = \frac{1}{N}\sum_{n=1}^{N}\mathbf{y_n}$.

- If $\mathbf{U^T}$ denotes the projection matrix on the subspace generated by the $R-1$ largest eigenvectors of the covariance matrix C, then a point $\mathbf{y_n}$ in $\mathbb{R}^L$ can be **approximately recovered** from its projection $\mathbf{t_n}$ in $\mathbb{R}^{R-1}$ through relation:

$$\mathbf{y_n} = \mathbf{m} + (\mathbf{U^T}\mathbf{U})^{-1}\mathbf{U^T}\mathbf{z_n}.$$

Table of content

Geometrical approaches

Geometrical methods estimate endmembers by identifying a simplex incorporating the spectral dataset $\{\mathbf{y_n}, 1 \leq n \leq N\}$.

1. **Pure pixel based methods** assume that at least one pure pixel per endmember is available within the dataset.

2. **Minimum volume based algorithms** seek to determine a simplex with minimal volume enclosing the dataset.

Table of content

N-FINDR algorithm

The N-FINDR algorithm[1] seeks to determine the largest volume simplex built with R endmembers selected from the data.

1. *(Initialization)* Select R random points $\{\mathbf{e_1^{(1)}}, .., \mathbf{e_R^{(1)}}\}$ from the dataset.
2. At step $k \geq 1$, compute the volume of the simplex $\mathcal{S}(\mathbf{e_1^{(k)}}, .., \mathbf{e_R^{(k)}})$:

$$V(\mathcal{S}^{(k)}) = \frac{1}{R!} |\det(\mathbf{e_2^{(k)}} - \mathbf{e_1^{(k)}}, .., \mathbf{e_R^{(k)}} - \mathbf{e_1^{(k)}})|$$

[1] Winter (1999). N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data

N-FINDR algorithm (ii)

3. For all $\mathbf{r}$ in the dataset, for all p between 1 and R, calculate the volume of the simplex $\mathcal{S}_p^{(k)}(\mathbf{e_1^{(k)}}, .., \mathbf{e_{p-1}^{(k)}}, \mathbf{r}, \mathbf{e_{p+1}^{(k)}}, .., \mathbf{e_R^{(k)}})$
4. *(Replacement)* if $V(\mathcal{S}^{(k)}) < \mathcal{S}_p^{(k)}$, update the set of endmembers by replacing $\mathbf{e_p}^{(k)}$ by $\mathbf{r}$.
5. *(Stopping condition)* The algorithm terminates if for all $\mathbf{r}$ in the dataset, for all p between 1 and R, $V(\mathcal{S}^{(k)}) \geq \mathcal{S}_p^{(k)}$.

N-FINDR algorithm (ii)

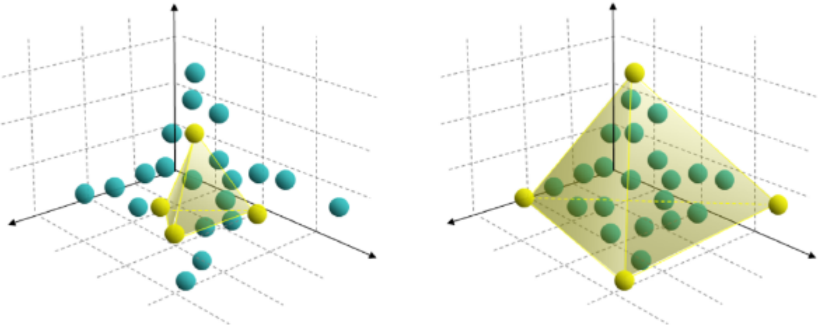

Figure: Schematic view of NFINDR algorithm.

Table of content

Abundances estimation

- When the set of endmembers is known, **abundances** can be estimated by least square.

- A first approach is to minimize for each $n = 1, .., N$ the **least square reconstruction error**

$$E = ||\mathbf{M}\mathbf{a_n} - \mathbf{y_n}||^2$$

- An **analytical solution** is available,

$$\hat{\mathbf{a}}_\mathbf{n} = (\mathbf{M^T M})^{-1}\mathbf{M}^T\mathbf{y_n},$$

where $(\mathbf{M^T M})^{-1}\mathbf{M}^T$ is the **pseudo-inverse** of the matrix M.

- However, this approach does not account for the **non-negativity and sum-to-one constraints**.

Abundances estimation (ii)

- To account for both **non-negativity and sum-to-one constraints**, one has to solve the optimization problem

$$\min_{\mathbf{a_n}} ||\mathbf{M}\mathbf{a_n} - \mathbf{y_n}||^2,$$

subject to $\sum_i \mathbf{a}_{\mathbf{n},i} = 1$, $0 \leq \mathbf{a}_{\mathbf{n},i} \leq 1$ for $i = 1, ..., R$.

- A useful trick to remove the sum-to-one constraint is to consider the equivalent problem

$$\min_{\mathbf{a_n}} ||\tilde{\mathbf{M}}\mathbf{a_n} - \tilde{\mathbf{y}}_\mathbf{n}||^2,$$

subject to $0 \leq \mathbf{a}_{\mathbf{n},i} \leq 1$ for $i = 1, ..., R$, where:

$$\tilde{\mathbf{M}} = \begin{bmatrix} \mathbf{M} \\ \mathbf{1}_\mathbf{D}^\mathbf{T} \end{bmatrix} \qquad \tilde{\mathbf{y}}_\mathbf{n} = \begin{bmatrix} \mathbf{y_n} \\ 1 \end{bmatrix}$$

Gradient descent

- Let f be a **convex**, **differentiable** function defined on $\mathbb{R}^D$ A simple way to solve the unconstrained minimization problem

$$\min_x f(x)$$

  is to use a **gradient descent** algorithm.

- The idea behind the gradient descent algorithm is to produce a **minimizing sequence** $x^k, k = 1, ...,$ in $\mathbb{R}^D$ where

$$x^{k+1} = x^k + t^k \Delta x^k.$$

  $\Delta x \in \mathbb{R}^D$ is the **search direction** and t the **step length**. The sequence is said to be minimizing if for all k, $f(x^{k+1}) < f(x^k)$.

Gradient descent (ii)

The gradient $\nabla f(x^k)$ constitutes a natural choice for the search direction. The resulting method is referred to as **gradient descent algorithm.**

Gradient descent method
Given a starting point $x \in \mathbb{R}^D$
Repeat

1. $\Delta x := -\nabla f(x)$
2. Choose a step size t
3. Update: $x := x + t\Delta x$.

until some stopping criterion is satisfied.

Backtracking line search

How to select the step size?

Backtracking line search

Given a descent direction $\Delta x = -\nabla f(x)$ for f at x and $\alpha \in [0, 0.5]$, $\beta \in [0, 1]$, do:

$t \leftarrow 1$

while $f(x - t\nabla f(x)) > f(x) - \alpha t \|\nabla f(x)\|^2$:

    $t \leftarrow \beta t$.

Projected gradient descent

Can the gradient descent algorithm be adapted to account for inequality constraints?

Gradient descent method
Given a starting point $x \in \mathbb{R}^D$
Repeat

1. $\Delta x := -\nabla f(x)$
2. Choose a step size t through a Backtracking line search
3. Update and project on the admissible set: $x := \Pi_C(x + t\Delta x)$

until some stopping criterion is satisfied.

Abundances estimation

- Problem to solve:

$$\text{minimize}_{\mathbf{a}\in\mathbb{R}^R} J(\mathbf{a}) := ||\tilde{\mathbf{M}}\mathbf{a} - \tilde{\mathbf{y}}||^2,$$

subject to $0 \leq \mathbf{a}_i \leq 1$ for $i = 1, ..., R$, where:

$$\tilde{\mathbf{M}} = \begin{bmatrix} \mathbf{M} \\ \mathbf{1}_{\mathbf{D}}^{\mathbf{T}} \end{bmatrix} \qquad \tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix}$$

- Gradient expression:

$$\nabla_{\mathbf{a}}(J) = 2\tilde{M}^T(\tilde{\mathbf{M}}\mathbf{a} - \tilde{\mathbf{y}})$$

- Projection on the admissible set

$$\pi(a_i) = \begin{cases} 0 & \text{if } a_i < 0 \\ 1 & \text{if } a_i > 1 \\ a_i & \text{otherwise} \end{cases}$$