Implementation of Hyperspectral Image Unmixing

Brandon Richardson
EE364b: Convex Optimization II Class Project

Background

- An imaging spectrometer is an instrument that captures a continuous spectrum of light (often ranging from 460nm to 2.5microns) for every pixel in a photograph.
- Image unmixing is a key aspect of hyperspectral imaging, allowing a complex spectrum produced by a single pixel to be decomposed into a set of basis spectrums. The basis spectrum when linearly combined make up the spectrum observed in that pixel.
- This set of basis spectrums can then be easily compared to a known spectrum library to determine the composition of that pixel. This particular technique is extremely useful when the image is of unknown composition.

Problem

- In a standard hyperspectral image unmixing problem we are given a hyperspectral image cube \( Y \in \mathbb{R}^{n \times m \times k} \) with \( n \) pixels and \( m \) frequency bands, we are then asked to find two matrices \( W \) and \( H \) such that \( Y \approx WH \).
- The \( W \) matrix is called the end member matrix. It contains the set of basis spectrums that make up the image \( Y \). The matrix \( H \) is called the abundance matrix, it chooses the mixing coefficients.
- Our hyperspectral image unmixing problem becomes a biconvex optimization problem where we want to find the \( W \) and \( H \) matrices.
- The \( Y \) matrix is the set of all pixels in close proximity to pixel \( i \).

Algorithm

- The algorithm I used for this project was originally presented by Zymnis, et al in his paper titled Hyperspectral Image Unmixing via Alternating Projected Subgradients. A condensed version of the algorithm is show below:

\[
\begin{align*}
    & T, \lambda, \alpha \\
    & H^{(0)} = W^{(0)} \\
    & \text{for} \quad t = 1 \text{ to } T \\
    & \quad \text{for} \quad p = 1 \text{ to } P \\
    & \quad \quad H = (H - (1/\sqrt{P})(2W^T WH - 2W^T Y + \lambda S))p \\
    & \quad W = (W + (1/||H^T||)(WH - Y)H^T)_+ \\
    & \text{end} \\
    & \text{end} \\
    & \text{end}
\end{align*}
\]

Results

- To test the algorithm I applied the subgradient version of this method to two sets of Hyper spectral images.
- The first data set was taken of a rock sample which was chosen for its unique minerals properties. This set has large uniform areas of minerals, which should stress the algorithms ability to pull out the proper end member spectra. As you can see in Figures 1 and 2 the algorithm properly pulled out two known end members, one of which was easy to find in the USGS library.
- The second data set was taken from the images produced by the Moon Mineralogy Mapper (M3). The M3 image I picked was of the Mare Orientale, a region know to contain hydroxyl signatures. This set is spectrally very uniform; it will stress the algorithms ability to pull out the proper end member spectra.

Implementation of ADMM

- In an attempt to improve the algorithm I implemented alternating direction method of multiplier (ADMM) for the abundance matrix update shown below.

\[
\begin{align*}
    H^{(t+1)} &= (W^T W + \rho I)^{-1}(W^T Y + \rho Z^{(t)} - U^{(t)}) \\
    Z^{(t+1)} &= (H^{(t+1)} + \rho U^{(t)})/\rho \\
    U^{(t+1)} &= U^{(t)} + \rho (H^{(t+1)} - Z^{(t+1)})
\end{align*}
\]

- This method can be easily substituted in directly for the sub-gradient update that Zymnis, et al used. The method shown above is actually a condensed version of the update which is being preformed on each \( h_i \) separately.
- This particular implementation is a standard lasso problem. This update excludes the penalty function which addresses pixel dissimilarity, and adds a regularization constraint on \( H \).

Results

- To test the algorithm I applied the subgradient version of this method to two sets of Hyper spectral images.
- The first data set was taken of a rock sample which was chosen for its unique minerals properties. This set has large uniform areas of minerals, which should stress the algorithms ability to reduce pixel dissimilarity. As you can see in Figures 1 and 2 the algorithm properly pulled out two known end members, one of which was easy to find in the USGS library.
- The second data set was taken from the images produced by the Moon Mineralogy Mapper (M3). The M3 image I picked was of the Mare Orientale, a region know to contain hydroxyl signatures. This set is spectrally very uniform; it will stress the algorithms ability to pull out the proper end member spectra. As you can see in Figure 3 the algorithm only found two end members. Unfortunately, these two end members are spectrally identical with the only difference being total illumination.
- There simply isn't enough spectral variation in the M3 data; this causes the algorithm to favor an end member matrix based only on total luminance with little to no spectral difference.

Results

- To test the algorithm I applied the subgradient version of this method to two sets of Hyper spectral images.
- The first data set was taken of a rock sample which was chosen for its unique minerals properties. This set has large uniform areas of minerals, which should stress the algorithms ability to reduce pixel dissimilarity. As you can see in Figures 1 and 2 the algorithm properly pulled out two known end members, one of which was easy to find in the USGS library.
- The second data set was taken from the images produced by the Moon Mineralogy Mapper (M3). The M3 image I picked was of the Mare Orientale, a region know to contain hydroxyl signatures. This set is spectrally very uniform; it will stress the algorithms ability to pull out the proper end member spectra. As you can see in Figure 3 the algorithm only found two end members. Unfortunately, these two end members are spectrally identical with the only difference being total illumination.
- There simply isn't enough spectral variation in the M3 data; this causes the algorithm to favor an end member matrix based only on total luminance with little to no spectral difference.

Results Rock Images

- In an attempt to improve the algorithm I implemented alternating direction method of multiplier (ADMM) for the abundance matrix update shown below.

\[
\begin{align*}
    H^{(t+1)} &= (W^T W + \rho I)^{-1}(W^T Y + \rho Z^{(t)} - U^{(t)}) \\
    Z^{(t+1)} &= (H^{(t+1)} + \rho U^{(t)})/\rho \\
    U^{(t+1)} &= U^{(t)} + \rho (H^{(t+1)} - Z^{(t+1)})
\end{align*}
\]

- This method can be easily substituted in directly for the sub-gradient update that Zymnis, et al used. The method shown above is actually a condensed version of the update which is being preformed on each \( h_i \) separately.
- This particular implementation is a standard lasso problem. This update excludes the penalty function which addresses pixel dissimilarity, and adds a regularization constraint on \( H \).

Results

- To test the algorithm I applied the subgradient version of this method to two sets of Hyper spectral images.
- The first data set was taken of a rock sample which was chosen for its unique minerals properties. This set has large uniform areas of minerals, which should stress the algorithms ability to reduce pixel dissimilarity. As you can see in Figures 1 and 2 the algorithm properly pulled out two known end members, one of which was easy to find in the USGS library.
- The second data set was taken from the images produced by the Moon Mineralogy Mapper (M3). The M3 image I picked was of the Mare Orientale, a region know to contain hydroxyl signatures. This set is spectrally very uniform; it will stress the algorithms ability to pull out the proper end member spectra. As you can see in Figure 3 the algorithm only found two end members. Unfortunately, these two end members are spectrally identical with the only difference being total illumination.
- There simply isn't enough spectral variation in the M3 data; this causes the algorithm to favor an end member matrix based only on total luminance with little to no spectral difference.