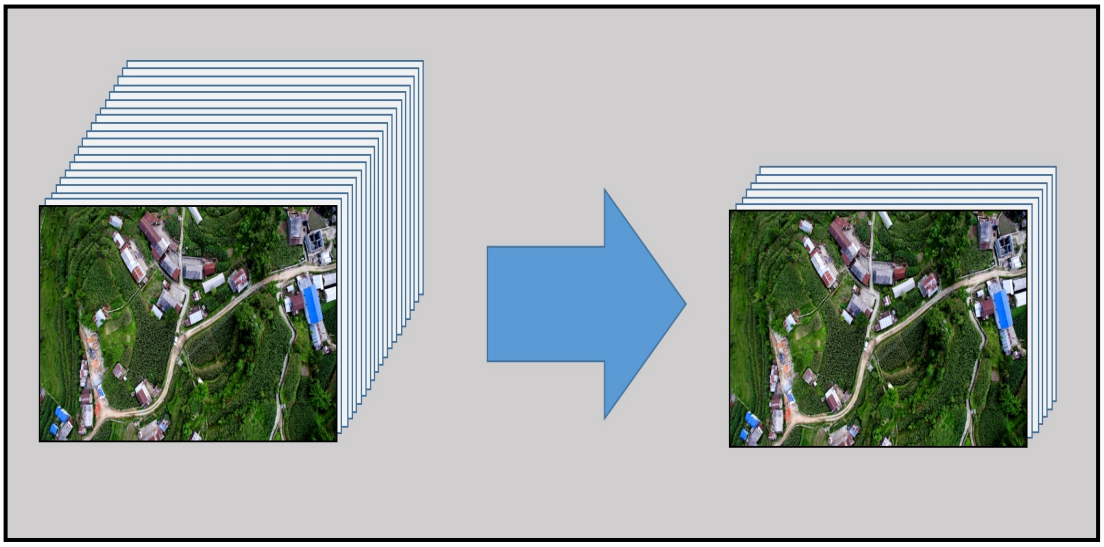
Band selection in hyperspectral images using artificial neural networks

Mateus Habermann

Par **Mateus HABERMANN**

Band selection in hyperspectral images using artificial neural networks

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 27 septembre 2018
Spécialité : Technologies de l'Information et des Systèmes : Unité de recherche Heudyasic (UMR-7253)

D2434

Thèse

présentée pour l'obtention du grade de

Docteur de l'Université de Technologie de Compiègne

Spécialité : Technologies de l'Information et des Systèmes

Band Selection in Hyperspectral Images using Artificial Neural Networks

Mateus Habermann

27 Septembre 2018

Band Selection in Hyperspectral Images using Artificial Neural Networks

Soutenue le 27 Septembre 2018 devant un jury composé de :

M. Yves Grandvalet	Directeur de recherche Laboratoire Heudiasyc, UTC ***Président***
M^{me} Veronique Berge-Cherfaoui	Professeur des universités Laboratoire Heudiasyc, UTC ***Examinatrice***
M. Antônio de Pádua Braga	Professeur Ecole d'Ingénieur de l'Université Fédérale de Minas Gerais, Belo Horizonte, Brésil ***Rapporteur***
M. Sébastien Lefèvre	Professeur Laboratoire IRISA, Vannes ***Rapporteur***
M. Vincent Frémont	Professeur Ecole Centrale Nantes ***Directeur de thèse***
M. Elcio Hideiti Shiguemori	Enseignant chercheur Institut d'Etudes Avancées, São José dos Campos, Brésil ***Directeur de thèse***

Band Selection in Hyperspectral Images using Artificial Neural Networks

To my parents.

Acknowledgements

I am deeply thankful for the support and help of my supervisors, namely, Professor Dr Vincent Frémont and Dr Elcio Hideiti Shiguemori.

This PhD thesis was carried out in the framework of the Labex MS2T and DIVINA challenge team, which were funded by the French Government, through the program "Investments for the Future" managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

I am also thankful for the support provided by the Brazilian government.

Résumé

Les images hyperspectrales (HSI) fournissent des informations spectrales détaillées sur les objets analysés. Étant donné que différents matériaux ont des signatures spectrales distinctes, les objets ayant des couleurs et des formes similaires peuvent être distingués dans le domaine spectral.

Toutefois, l'énorme quantité de données peut poser des problèmes en termes de stockage et de transmission des données. De plus, la haute dimensionnalité des images hyperspectrales peut entraîner un surajustement du classificateur en cas de données d'apprentissage insuffisantes. Une façon de résoudre de tels problèmes consiste à effectuer une sélection de bande (BS), car elle réduit la taille du jeu de données tout en conservant des informations utiles et originales.

Dans cette thèse, nous proposons trois méthodes de sélection de bande différentes. La première est supervisée, conçu pour utiliser seulement 20 % des données disponibles. Pour chaque classe du jeu de données, une classification binaire un contre tous utilisant un réseau de neurones est effectuée et les bandes liées aux poids le plus grand et le plus petit sont sélectionnées. Au cours de ce processus, les bandes les plus corrélées avec les bandes déjà sélectionnées sont rejetées. Par conséquent, la méthode proposée peut être considérée comme une approche de sélection de bande orientée par des classes.

La deuxième méthode que nous proposons est une version non supervisée du premier framework. Au lieu d'utiliser les informations de classe, l'algorithme K-Means est utilisé pour effectuer une classification binaire successive de l'ensemble de données. Pour chaque paire de grappes, un réseau de neurones à une seule couche est utilisé pour rechercher l'hyperplan de séparation, puis la sélection des bandes est effectuée comme décrit précédemment.

Pour la troisième méthode de BS proposée, nous tirons parti de la nature non supervisée des auto-encodeurs. Pendant la phase d'apprentissage, le vecteur d'entrée est soumis au bruit de masquage. Certaines positions de ce vecteur sont basculées de manière aléatoire sur zéro et l'erreur de reconstruction est calculée sur la base du vecteur d'entrée non corrompu. Plus l'erreur est importante, plus les fonctionnalités masquées sont importantes. Ainsi, à la fin, il est possible d'avoir un classement des bandes spectrales de l'ensemble de données.

Keywords: Sélection des bandes, images hyperspectrales, réseau de neurones.

Abstract

Hyperspectral images (HSIs) are capable of providing a detailed spectral information about scenes or objects under analysis. It is possible thanks to both numerous and contiguous bands contained in such images. Given that different materials have distinct spectral signatures, objects that have similar colors and shape can be distinguished in the spectral domain that goes beyond the visual range.

However, in a pattern recognition system, the huge amount of data contained in HSIs may pose problems in terms of data storage and transmission. Also, the high dimensionality of hyperspectral images can cause the overfitting of the classifier in case of insufficient training data. One way to solve such problems is to perform band selection (BS) in HSIs, because it decreases the size of the dataset while keeping both useful and original information.

In this thesis, we propose three different band selection frameworks. The first one is a supervised one, and it is designed to use only 20% of the available training data. For each class in the dataset, a binary one-versus-all classification using a single-layer neural network is performed, and the bands linked to the largest and smallest coefficients of the resulting hyperplane are selected. During this process, the most correlated bands with the bands already selected are automatically discarded, following a procedure also proposed in this thesis. Consequently, the proposed method may be seen as a class-oriented band selection approach, allowing a BS criterion that meets the needs of each class.

The second method we propose is an unsupervised version of the first framework. Instead of using the class information, the K-Means algorithm is used to perform successive binary clustering of the dataset. For each pair of clusters, a single-layer neural network is used to find the separating hyperplane, then the selection of bands is done as previously described.

For the third proposed BS framework, we take advantage of the unsupervised nature of autoencoders. During the training phase, the input vector is subjected to masking noise—some positions of this vector are randomly flipped to zero—and the reconstruction error is calculated based on the uncorrupted input vector. The bigger the error, the more important the masked features are. Thus, at the end, it is possible to have a ranking of the spectral bands of the dataset.

Keywords: Band selection, hyperspectral image, artificial neural network.

Contents

List of Figures

List of Tables

List of Acronyms

ANN	Artificial Neural Network
BS	Band Selection
CART	Classification and Regression Trees
CNN	Convolutional Neural Networks
DD	Distance Density
DT	Decision Trees
EA	Evolutionary Algorithm
EM	Electromagnetic
ExpM	Expectation Maximization
FE	Feature Extraction
FS	Feature Selection
GA	Genetic Algorithm
HSI	Hyperspectral Image
IP	Image Processing
JF	Jeffreys-Matusita
KNN	k-Nearest Neighbors
KSC	Kennedy Space Center (HSI)
LSR	Least Squares Regression
MA	Memetic Algorithm
MAF	Multiplicative Aggregation Function
MDPP	Multigraph Determinantal Point Process
ML	Machine Learning
MR	Manifold Ranking
NHHMC	Non-Homogeneous Hidden Markov Chain
PCA	Principal Components Analysis
PR	Pattern Recognition

PRS	Pattern Recognition System
RGB	Red Green Blue
RST	Rough Set Theory
RS	Remote Sensing
SLNN	Single-layer Neural Network
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
SWIR	Short Wavelength Infrared
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1 General context

Single tasks of our daily life such as recognizing a handwritten character, a face, or identifying a certain object in our pocket by feel can be considered Pattern Recognition (PR) acts (Duda et al., 2001). The ease with which we perform such acts belies the complex processes that must be followed when computers enter the scene to perform those same single tasks.

Before a computer is used in a PR application, it must undergo a learning process in order to be acquainted with the problem at hand. This learning process takes into account the information from training samples related to the domain of interest. Thus, new patterns —or objects—not seen during the learning phase may be correctly analyzed by the computer in later executions of the algorithm. Furthermore, learning is done by using some method for reducing the classification error on a set of training data. That is, the computer teaches itself how to recognize the patterns by adjusting the classifier's parameters during the learning —or training—phase of the classifier.

Depending on the dataset, the learning process can be of one out of three types:

- Supervised: Under the supervised learning, a teacher gives a class information for each data sample in the training set. Then, it seeks to reduce the sum of costs for those patterns;
- Unsupervised: In this case there is no teacher. In unsupervised learning, or clustering, the system finds clusters of the input data. Consequently, the data structure is taken into account; and
- Semi-supervised: In this case, both labeled and unlabeled data are used. Thus, it can be seen as a mixture of supervised and unsupervised learning techniques.

Normally, only few labeled samples are used, and the learning is based on the assumption that data samples which are close to each other in the feature space are more likely to have the same label.

The classifier along with its learning process are part of a Pattern Recognition System (PRS). The PRS diagram shown in Figure 1.1 is composed of five steps, which lie between the input and the final decision (Duda et al., 2001). Each step is briefly described as follows:

- Sensing: Normally, the input to a PR system is a camera or a microphone array. The difficulty of the problem is related to the limitations if this input device, such as bandwidth, resolution, distortion, signal-to-noise ratio etc;
- Segmentation: All the analysis is made on individual patterns. Thus, those patterns must be segmented from the background. It is indeed a difficult task, and in many processes which have an image as input data, this issue is solved by considering each pattern as a pixel or a patch of a certain size.
- Feature extraction/selection: The goal of the feature selection/extraction is to describe a pattern to be recognized by measurements whose values are very similar for patterns in the same class, and dissimilar for patterns that belong to different classes. Also, the representation yielded by an ideal feature selection/extraction process makes the job of the classifier trivial. Considering a hyperspectral image (HSI), for example, a *feature extraction* process makes a combination of the original bands to form new features, whereas under the *feature selection* approach the resulting features set is a selection of the original bands.
- Classification: The task of the classifier is to use the feature vector provided by the *feature extraction/selection* component and assign the pattern to a class. The degree of difficulty of the classification task is related to the variance of the feature values for patterns in the same class in relation to the difference amongst feature values for patterns in different classes. In Figure 1.2, there are two examples of feature space. In the first example, Figure 1.2 (a), the variance within the classes is small, thus a linear classifier can be used in order to separate the two hypothetical classes, considering the features $f1$ and $f2$. In Figure 1.2 (b), the variance within classes is so high that a linear classification is no longer possible. Normally, the classes present in a hyperspectral image, for example, belong to this second case, that is, there is a high variance within the classes, what makes the classification of such images a challenging task.
- Post-processing: The post-processing component takes into consideration the context of the problem. It takes as input the classification map provided by the *classification* component, and analyzes each data point label by taking into account

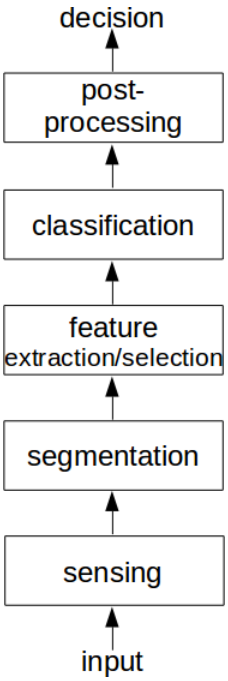


Figure 1.1: Components of a typical pattern recognition system.

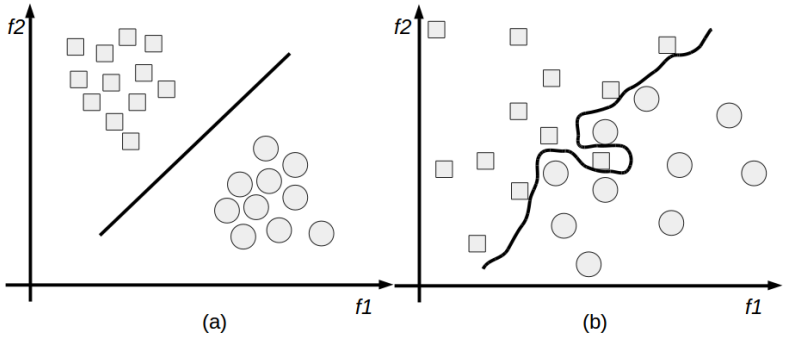


Figure 1.2: Two examples of feature space. In (a), the variance within the classes are small enough to permit a linear separation between the classes. In (b), there is a high variance within the classes, consequently a linear separation is no longer possible.

its neighbors in the image domain. If a label is not consistent with its neighbors, then the label is changed in order to match the label of the majority.

1.1.1 Sensing step issues

When the input to a pattern recognition system is a hyperspectral image, it is necessary to consider the implications of working with such a huge source of information. Compared with a RGB image, for example, a hyperspectral image (HSI) has much more information about the scene under analysis due to the fact that it may have hundreds of bands instead of only three (Pu, 2017), as shown in Figure 1.3. Consequently, the richness of details in a hyperspectral image may provide a good discrimination amongst

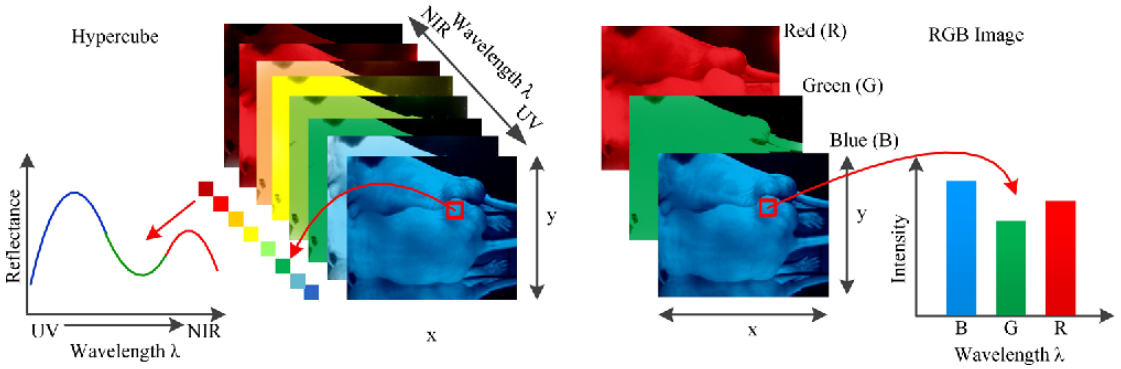


Figure 1.3: Comparison between HSI and RGB image. HSI is a three-dimensional dataset of a 2D image on each wavelength. On the the lower left: the reflectance curve, or spectral signature, of a pixel. The RGB image only has three image bands on red, green, and blue wavelengths, respectively. On the lower right: the intensity curve of a pixel in the RGB image (Lu and Fei, 2014).

patterns of different classes. However, the huge amount of data contained in HSIs may pose hardware problems, such as data storage (Molisch, 2005) and transmission (Wijitdechakul et al., 2016), which belong to the *sensing* step of a pattern recognition system.

Data storage and transmission issues become more evident when unmanned aerial vehicles (UAVs) are used, due to their small electronic components. UAVs-based technology, for example, is vital for the next-generation monitoring systems, because UAVs can obtain data at high altitudes (Zhang et al., 2018). Besides, due to the advantages of low cost, small consumption, convenience and safety the unmanned aerial vehicles are widely used in the military (Roberge et al., 2018) and civilian fields (Oliveira et al., 2018).

The UAV navigation can be performed either in an autonomous (da Silva et al., 2015; Braga et al., 2015, 2016; Kuroswiski et al., 2018) or in an externally controlled fashion. For the latter, it is possible to employ UAVs in a scenario where they are controlled by a ground station computer, which can perform heavy computations using Graphic Processing Units and parallelization (Silva et al., 2017), as shown in Figure 1.4. In this case, each UAV sends the collected data to a computer, which in turn makes computations with its more powerful resources and then sends information back to the UAV (Buyukyazi et al., 2013).

But, when hyperspectral data are used (Zhong et al., 2017; Freitas et al., 2018), the hardware limitations may pose a threat to the real-time nature of the mission. Thus, reducing the amount of transmitted data can be a good alternative.

Hyperspectral image data reduction may be done by means of band selection (BS) (Wang et al., 2018a). Briefly, BS methods seek to select some few bands, based on a predefined criterion such as classification accuracy—in supervised cases—, or data structure, when there is no class information.

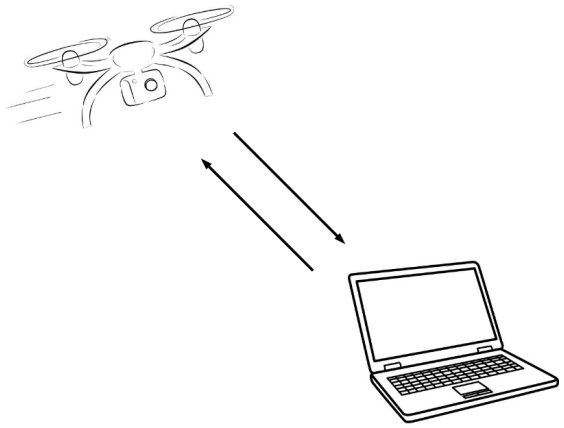


Figure 1.4: Example of data transmission between an UAV and a ground station computer.

1.1.2 Classification step issues

The resulting feature space after a BS operation has the same dimension as the number of selected bands. Therefore, the feature space dimension can be drastically reduced if only a small fraction of the original bands is selected.

By keeping the same quantity of training data, classifiers working in smaller dimensions are less likely to be affected by the curse of dimensionality (Bai et al., 2017; Habermann et al., 2017a), which hampers the classifiers' job. Consequently, the data reduction accomplished by band selection methods can also be beneficial to the classification step of the PRS.

1.2 Band selection challenges

1.2.1 Selection strategy

Numerically speaking, band selection seems to be—and indeed could be—a very long and computationally heavy process. For example, one could select 10 out of 150 spectral bands in

$$C(150, 10) = \frac{150!}{(150-10)!10!} \approx 1.17 \times 10^{15}$$

ways. Obviously, it is both a huge and discouraging number.

Fortunately, the band selection methods found in literature do not perform an exhaustive search on all possible band combinations. By adopting some criteria such as distance measures (Keshava, 2004), class separability measures (Cui et al., 2011), dependence

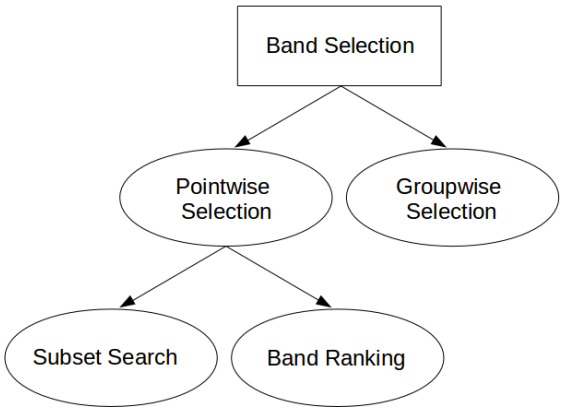


Figure 1.5: Band selection approaches.

(Camps-Valls et al., 2010), classification measures (Habermann et al., 2017b), it is possible to guide the band search and thus avoid an exhaustive search.

In general, as shown in Figure 1.5, BS methods can be divided into *pointwise* and *groupwise* approaches (Theodoridis and Koutroumbas, 2008). The latter separates all the spectral bands into several subsets. Then, the final band selection may done basically in two ways: *i)* all the bands in a certain subset are selected; or *ii)* a representative of each subset is selected. As for the pointwise methods, they can be subdivided into two groups:

- Subset search: the number of bands to be selected is previously determined. At each iteration, a band of interest is selected, and the algorithm stops when the desired number of bands is achieved; and
- Band ranking: during the iterations, each band receives pertinence values that are summed up. At the end, there is a ranking indicating the most important bands according to a certain criterion. In this case, the number of bands does not need to be defined *a priori*.

1.2.2 Paucity of the training data

Another important issue to be considered during the design of a BS framework is the paucity of the training data.

Some band selection methods perform their job using the classifier of the pattern recognition system, described in Section 1.1. This strategy—*wrapper* (Suto et al., 2016) approach—seems to be very appealing, because the classifier can select the bands that maximize its classification accuracy. However, some classifiers may have too many parameters to be adjusted, consequently, lots of training data are necessary, otherwise the classifier will be subjected to overfitting (Bilbao and Bilbao, 2017).

An alternative to this problem is the *filter*-based band selection (Adlakha and Chhikara, 2016). Under this approach, the band selection happens before the classification step of the PRS. Thus, BS algorithms simpler than the classifier can be used, thus less training data may be required.

1.3 Addressed Problems and Contributions

In the literature, different mathematical and statistical tools are used by the BS frameworks. We may cite data manifolds (Li and Hao, 2007), data information analysis (Yan et al., 2017), graph theory (Bollobás, 1998) and evolutionary computing (Eiben and Smith, 2015).

Machine learning (ML) (Haykin, 2009) is a branch of Artificial Intelligence that uses statistical tools to give machines and computers the capacity to learn. That is, a ML-based BS algorithm is capable of learning by itself the most appropriate spectral bands according to a certain criterion. Therefore, we chose to use ML in our proposed band selection methods.

Thus, in this thesis, we address the BS subject by proposing three filter-based frameworks. One of them is supervised and the other two are unsupervised. All of them follow a pointwise search approach, as a consequence of the architectures we chose. They are:

- *Supervised hyperspectral band selection using single-layer neural networks.*
- *Unsupervised hyperspectral band selection based on clustering and single-layer neural networks*; and
- *Unsupervised hyperspectral band selection using autoencoders.*

1.3.1 Supervised hyperspectral band selection using single-layer neural networks

We propose a supervised filter-based BS approach using single-layer neural networks. Thus, after the training phase, it is possible to draw conclusions about the spectral bands based on the neural net's weights.

For each class of the data set, a binary single-layer neural network finds a separating hyperplane between that class and the remainder of the data. Then, the bands related to the biggest and smallest hyperplane's coefficients are selected, so, one can say that the band selection process is class-oriented. This process iterates until a previously defined number of bands is selected.

By comparing with three state-of-the-art supervised band selection approaches, it is possible to see that our method yields better results in many situations even with greatly reduced training data size.

1.3.2 Unsupervised hyperspectral band selection based on clustering

We propose an unsupervised framework for band selection based on clustering and neural networks.

The proposed method starts with a binary clustering of the whole data set performed by k-Means algorithm. After that, a single-layer neural network is trained to perform a binary classification between the two clusters. Then, the bands related to the biggest and smallest separating hyperplane coefficients are selected. This process is repeated using as input data the newly generated clusters until a predefined number of bands is selected.

A comparison with four other BS methods shows the validity of our framework.

1.3.3 Unsupervised hyperspectral band selection using autoencoders

We propose an unsupervised approach for band selection using autoencoders, which are unsupervised neural networks that learn the data structure, and reconstruct the input vector at the output.

In sum, during the training phase of the autoencoder, some features of the data samples are turned to zero, by a masking noise transform. The subsequent reconstruction error is assigned to the indices that were subjected to the masking noise. We adopt the following criterion: The bigger the error, the greater the importance of the masked features. Then, the errors are summed up during the training phase. At the end, we select the bands with the biggest indices.

A comparison with four other BS frameworks shows that the our algorithm yields better results in some specific cases, and similar performance in other situations.

1.4 Publications

- **Problem-based Band Selection for Hyperspectral Images** (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), 2017 IEEE International Geoscience and Remote Sensing Symposium, IGARSS, (international conference paper)

- **Feature Selection for Hyperspectral Images Using Single-Layer Neural Networks**, (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), 8th International Conference on Pattern Recognition Systems, 2017, (international conference paper)
- **Unsupervised Band Selection in Hyperspectral Images Using Autoencoder**, (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), 9th International Conference on Pattern Recognition Systems, 2018, (international conference paper)
- **Clustering-based Unsupervised Hyperspectral Band Selection Using Single-Layer Neural Network**, (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), Conférence Française de Photogrammétrie et de Télédétection, CFPT 2018, (french conferencer paper)
- **Supervised Band Selection in Hyperspectral Images Using Single-Layer Neural Networks**, (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), International Journal of Remote Sensing, (journal paper, accepted)
- **Unsupervised Hyperspectral Band Selection using Clustering and Single-Layer Neural Network**, (*Mateus Habermann, Vincent Frémont, Elcio Hideiti Shiguemori*), Revue Française de Photogrammétrie et de Télédétection (under review)

1.5 Thesis Structure

In order to make the reading of this thesis more pleasant, we divide the text into seven parts. The first one, which you are now reading, is called *Introduction*. Here we presented the context and motivation for the proposed BS frameworks.

In Chapter 2, there are some basic concepts of Remote Sensing, as well as some information about the electromagnetic spectrum. Furthermore, we will see that a hyperspectral image has hundreds of bands, and that the resulting high dimensional feature space may pose sparsity-related problems, affecting the classifier's performance. Due to the high correlation amongst neighboring bands, it is possible to reduce the feature space dimensionality without losing much useful information. For this, we opt for band selection, whose state-of-the-art methods can be found in Chapter 3.

Then, in Chapter 4, we present our supervised band selection framework. Chapters 5 and 6 contain our two unsupervised BS frameworks. Finally, in Chapter 7 one can find the conclusion to this thesis.

Chapter 2

Fundamental Tools

This thesis addresses the feature selection (FS) subject. In order to be precise, it is more appropriate to speak of band selection, instead of FS, due to the fact that the data sets to be analyzed in this thesis are exclusively composed of hyperspectral images.

Hyperspectral band selection is a task that demands the knowledge of techniques that go beyond the Pattern Recognition sphere. It also requires a certain comprehension of other subjects, such as Remote Sensing and Spectroscopy, so that one can understand the image formation and the reason why adjacent spectral bands are highly correlated. By the way, because of this correlation, it is possible to perform hyperspectral band selection without discarding important information.

2.1 Fundamentals of Remote Sensing

Sensing something remotely, in its strict sense, is to measure it indirectly, *i.e.*, without a physical contact. In many applications, it is not possible to measure features of the objects of interest directly, therefore it is necessary to resort to the measurement of some other quantities related to the sought ones (Ustinov, 2015).

The idea of sensing an area or object without physical contact is rather old. Concerning aerial images, for example, the French photographer and balloonist Gaspard-Félix Tournachon[1] took the first known aerial photograph in the year of 1858. In 1903, a small light weight camera was attached to a pigeon of the Bavarian Pigeon Corps, in order to capture aerial images, as shown in Figure 2.1[2]. The aerospace industry improved the Remote Sensing (RS) technology, especially with the space exploration in the sixties. In our days, one can see spaceborne images on television during weather forecasts, for

[1] https://francearchives.fr/commemo/recueil-2010/39161
[2] https://www.pinterest.fr/pin/109353097178882074/?lp=true

Figure 2.1: Left: a pigeon of Bavarian Pigeon Corps, with a light weight camera. Center and right: some images acquired during the flight.

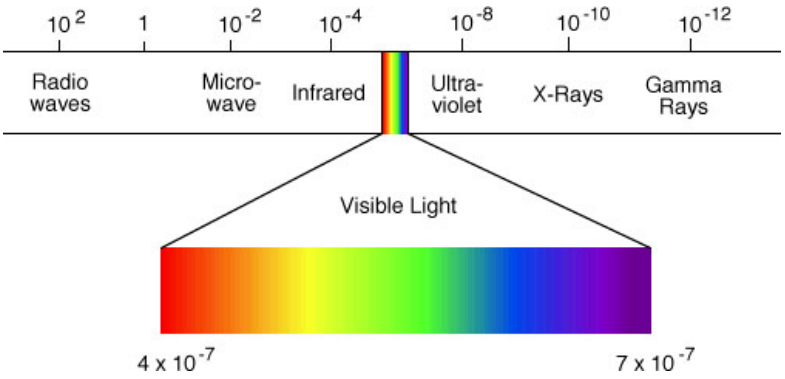


Figure 2.2: Electromagnetic spectrum with emphasis on visible light.

instance. The advances in technology allow for the unbounded increasing of RS applications, such as X-Rays and Magnetic Resonance Imaging. However, it is worth noting that the main driving force behind the RS technology is the military industry.

Remote Sensing, as a science, is a recognized inter-disciplinary field. In this thesis, the RS's facet we will explore is that of classification of objects on the surface of the Earth, by analyzing aerial images taken by satellites. In reality, this thesis' contribution is focused on a preprocessing step before the classification itself takes place. More precisely, we perform the selection of spectral bands, which are, in turn, the measurements of the electromagnetic energy that emanates from the targets. Thus, it is important to understand some principles of emission and reflectance (Khorram et al., 2016). Before that, though, let us remember some basic aspects of the electromagnetic spectrum.

2.1.1 The electromagnetic spectrum

In Figure 2.2, the electromagnetic (EM) spectrum is shown. It is normally characterized by the wavelength, which ranges from radio waves until gamma rays (Solimini, 2016). The visible light embraces only a small fraction of the spectrum, exposing the fact that there are much more information in the world than our eyes are able to see.

Some sensors are capable of capturing the electromagnetic energy beyond the visual spectrum, thus obtaining more information from the objects under scrutiny when compared

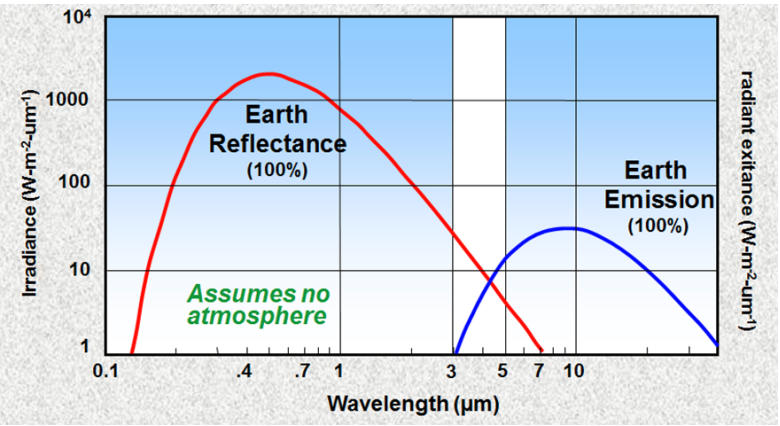


Figure 2.3: Earth emission and reflectance curves.

to RGB cameras, for instance. In fact, the interaction between matter and electromagnetic radiation is particular for each material, creating, consequently, a peculiar spectral signature (Schowengerdt, 2006).

2.1.2 Emission and reflectance

When the electromagnetic radiation comes into contact with the matter, there is an interaction between them. It can be: *absorption, scattering, reflection* or *emission* of the energy by the matter. There is also the *transmission* of the energy through the matter (Khorram et al., 2016). Normally, Remote Sensing deals with the reflected and emitted energy.

When it comes to Earth reflectance—see Figure 2.3[3]—, it is worth noting that its irradiance peak takes place in the spectrum region with wavelengths varying between 0.4 m^{-6} and 0.7 m^{-6}, which coincides with the spectrum range that humans can see.

When the incident radiation hits an object, a fraction of it is reflected and the remaining energy is absorbed. Thus,

$$P^{(i)} = P^{(r)} + P^{(a)}, \tag{2.1}$$

where $P^{(i)}$ is the incident irradiance, $P^{(r)}$ is the reflected irradiance, and $P^{(a)}$ is the energy absorbed by the material (Solimini, 2016). In Figure 2.4 we can see this relation.

In fact, what a sensor measures is the reflectance ρ_λ, which varies according to the wavelength λ. More precisely,

$$\rho_\lambda = \frac{P_\lambda^{(r)}}{P_\lambda^{(i)}}. \tag{2.2}$$

[3]http://www.markelowitz.com/Hyperspectral.html

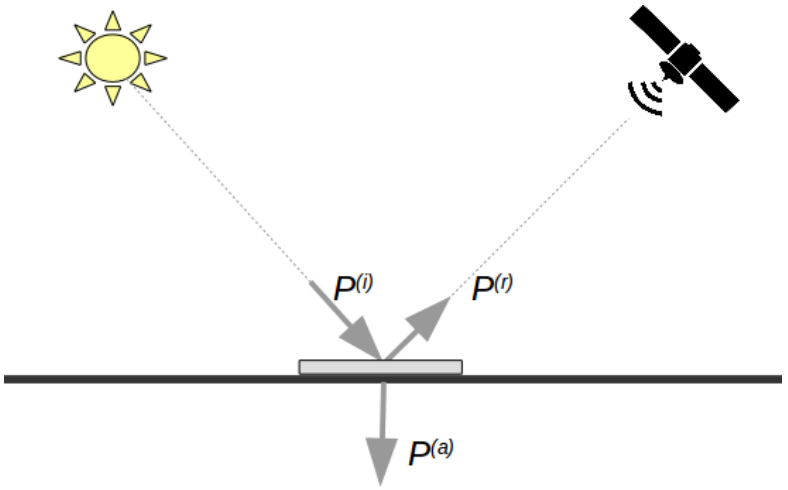


Figure 2.4: Reflected and absorbed energy.

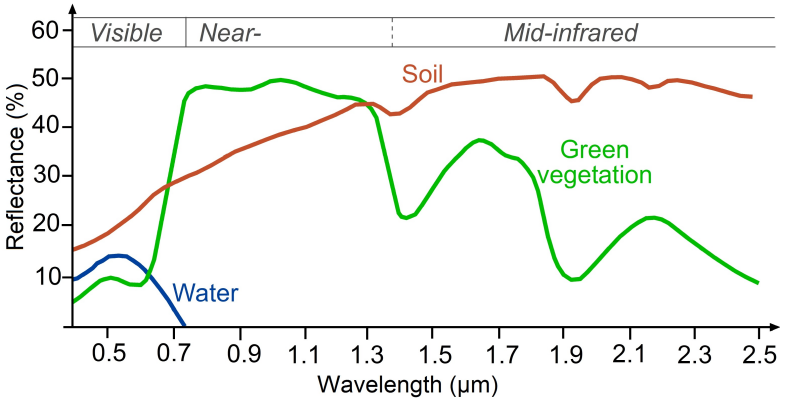


Figure 2.5: Spectral signatures of different elements. The shape of the signatures is defined by the absorption regions.

From (2.1) and (2.2), concerning the *reflected* energy, it is evident that:

- There must be a source of electromagnetic energy so that the reflectance of a target may be detected. It is the reason why a sensor that captures the reflected energy cannot work at night, for example; and
- The spectral signature of a target is modeled by the regions of energy absorption. This topic is addressed by Imaging Spectroscopy.

2.2 Imaging Spectroscopy

The main objective of Imaging Spectroscopy—also known as Hyperspectral Imaging—is to measure the chemical composition of the image content. This is possible due to the

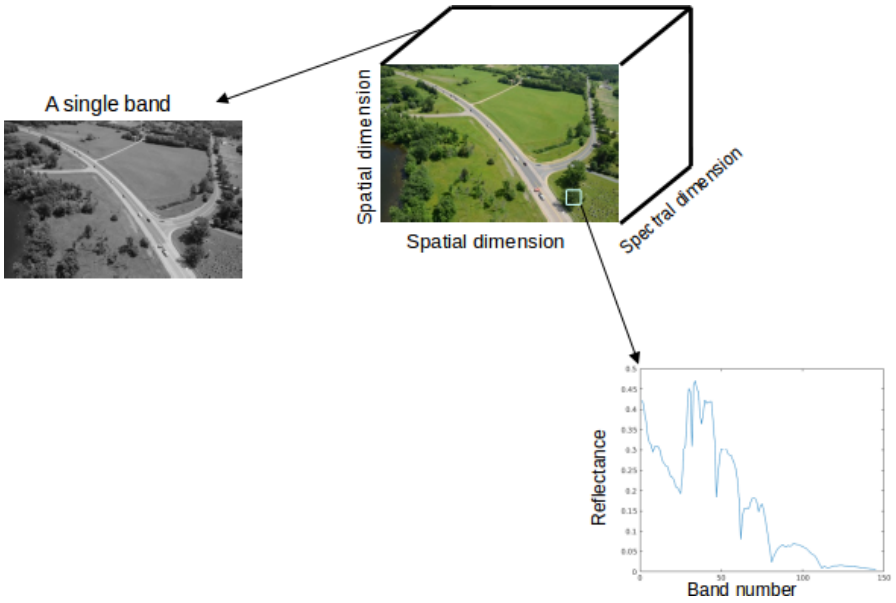


Figure 2.6: A stack of single-band images, also called hyperspectral cube. Consequently, each pixel is a vector containing a spectral signature.

energy absorption characteristics of each material (Antonio and Huynh, 2012). In Figure 2.5[4], it is possible to see that different materials have their own spectral signature.

Hyperspectral imaging systems employ sensors that mostly operate from visible up to SWIR (Short Wavelengths InfraRed). Also, those sensors can simultaneously acquire hundreds of both contiguous and narrow spectral channels. Consequently, each pixel of the resulting hyperspectral image (HSI) can be seen as a vector whose elements are measurements corresponding to specific wavelengths. The dimensionality of such vectors indicates the number of spectral bands of that image. In other words, HSIs can be seen as a three dimensional hyperspectral cube, as shown in Figure 2.6.

The detailed spectral information contained in HSI increases the possibility of correctly detecting specific targets and materials of interest, according to their spectral signature. Thus, many applications take advantage of using hyperspectral images:

- *Mineralogy*: A wide range of minerals can be identified by using hyperspectral data. Moreover, it is possible to investigate the effect of oil and gas leakages from natural wells and pipelines (Klima et al., 2014).
- *Precision agriculture*: Healthy green plants have a peculiar spectral signature. In the visible region of the spectrum, the curve shape is a consequence of absorption effects from chlorophyll and other leaf pigments. Chlorophyll has the property of absorbing visible light effectively, but it absorbs blue and red wavelengths more strongly than green, causing, as a consequence, the fact that healthy plants appear to us in green color. Reflectance value rises significantly across the limit between

[4]http://eumetrain.org/data/4/461/navmenu.php?tab=4&page=2.0.0

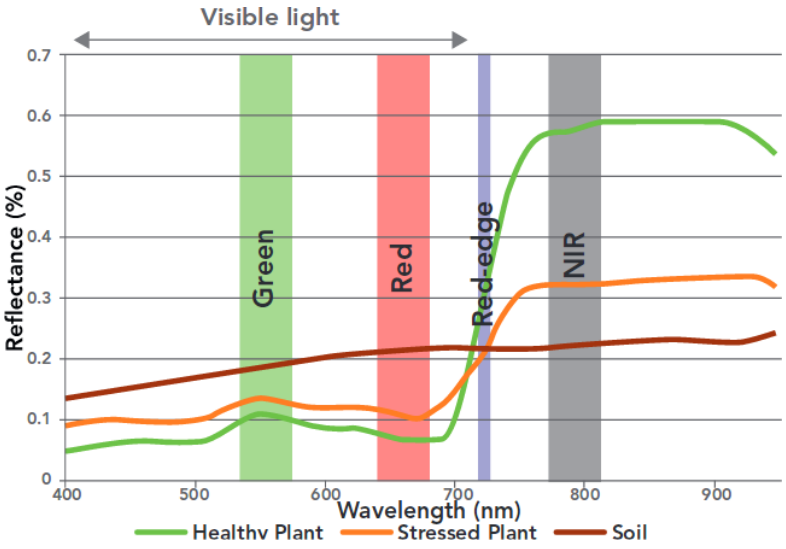


Figure 2.7: Spectral signatures of healthy plants, soil and stressed plants.

red and near infrared wavelengths, and this region is normally called *red edge*, as depicted in Figure 2.7 [5] At longer wavelengths, that is, in the *near infrared* region, water absorption prevails and the reflectance drops, and the spectral signature shape gives information on possible plant water stress. Consequently, HSIs provide the location of unhealthy crop spots, thus only those regions are subjected to specific treatments and measures, what can save both time and money (Murugan et al., 2016).

- *Geological science*: By using hyperspectral imagery, it is possible to recover physico-chemical mineral properties in terms of composition and abundance (Murphy et al., 2012).

- *Ecological science*: The biomass and carbon, and also the biodiversity in dense forest zones can be estimated by using hyperspectral images. Thus, land cover changes can be assessed (Lu et al., 2009).

- *Hydrological science*: Changes in wetland characteristics can be evaluated by means of HSIs. Furthermore, both water quality and coastal zones can also be analyzed by means of hyperspectral data (Klemas, 2014).

- *Military applications*: The rich spectral information provided by hyperspectral images can also be used for military purposes (Wang et al., 2018b). Camouflage, for instance, can deceit human eyes because a hidden target may end up having colors and texture that imitate its surroundings. However, since HSIs are able to sense wavelengths beyond the visual range, they may render camouflage useless (Hua et al., 2015).

[5] https://www.korecgroup.com/product/parrot-sequoia-sensor/

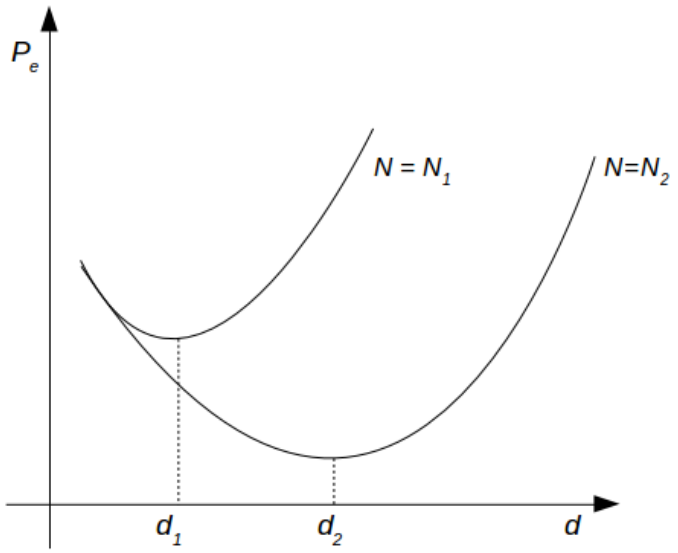


Figure 2.8: The peaking phenomenon (Theodoridis and Koutroumbas, 2008).

Thus, we can perceive that the spectral richness contained in hyperspectral images provides valuable information to be used in different science areas. This very spectral richness, on the other hand, also means that the feature space ends up having high dimensionality, and this can cause some negative consequences.

2.3 High Dimensional Data Issues

Hyperspectral images are known to have hundreds of bands, in some cases. When it comes to supervised classification approaches, for example, the number of labeled instances plays an important role in the performance of the classifier. More precisely, the number of training samples, N, must be large enough with respect to the number of bands, d, which is the dimensionality of the feature space.

Taking into account the accuracy of a classifier, for a given N, by increasing the number of spectral bands one gets an initial improvement in performance. But, after a critical value, a further increase of the quantity of bands yields an increase of the probability of error. This is called *peaking phenomenon*, or more popularly known as *curse of dimensionality* (Theodoridis and Koutroumbas, 2008). Figure 2.8 illustrates the general tendency we should expect by playing with the number of bands and the cardinality of the training data set. When $N_2 \gg N_1$, the error values corresponding to N_2 are lower than those of N_1. Besides, the peaking phenomenon occurs for $d_2 > d_1$. For each value of N, the probability of error P_e decreases when d gets bigger until a critical point, after which P_e increases.

One plausible explanation for the peaking phenomenon is the sparsity of data points in high dimensional feature spaces. It can be understood by using Geometry, and we can state the problem as follows:

As dimensionality increases, the volume of a hypercube concentrates in corners (Scott, 1992).

The volume of a hypersphere with radius r and dimension d is calculated by

$$V_s(r) = \frac{2r^d \pi^{d/2}}{\Gamma(\frac{d}{2})d}, \tag{2.3}$$

where Γ is the *gamma function*. The volume of a hypercube in $[-r, r]^d$ is given by

$$V_c(r) = (2r)^d. \tag{2.4}$$

The fraction f_{sc} between the volumes of a hypersphere inscribed in a hypercube is calculated according to

$$f_{sc} = \frac{V_s(r)}{V_c(r)} = \frac{\pi^{d/2}}{\Gamma(\frac{d}{2})(2^{d-1})d}. \tag{2.5}$$

It is easy to see that $\lim_{d\to\infty} f_{sc} = 0$. That is, as d increases, the volume of the hypercube—that may represent a feature space—increasingly concentrates in the corners. It is now obvious that high-dimensional spaces are completely different from 3-D ones. Thus, in such a situation, the data points would be so sparse that many different hyperplanes, for example, could provide a good separation amongst the classes. The resulting high variance of the hyperplane parameters reduces the statistical significance of the classifier's generalization power. The ratio N/d can be used as an indicator of this generalization (Theodoridis and Koutroumbas, 2008), because a large number of spectral bands can be translated into a large number of classifier parameters. Moreover, few training samples would make the data points so sparse in high dimensional feature spaces. So, the bigger this ratio, the better the classifier generalization.

Thus, one could conclude that increasing the number of training samples—to increase N/d—is a reasonable way to avoid high dimensionality-related problems. However, supervised approaches require labeled data during the training phase of the algorithm, and the labeling process of a hyperspectral image is done manually pixel by pixel, generally using some field measurements. It means that the collection of these labeled samples is both expensive and time consuming. Consequently, the number of available hyperspectral training data is normally limited, what causes serious issues in supervised classification.

Since it is expensive to have enough training data, we can resort to another way to increase the ratio N/d, namely, making d smaller. Decreasing d means decreasing the dimensionality of the feature space. Fortunately, there are, at least, two facts that support the HSI dimensionality reduction:

- Due to the high HSI dimensionality, the different classes present in the image may lie in manifolds embedded in subspaces of the original feature space. Thus, it is possible to explore the sparsity of the data set in order to find a more meaningful data representation (Bitar et al., 2017).
- The bands of a hyperspectral image are not only contiguous, but also have a very small bandwidth—10 nanometers or less. Consequently, neighboring bands have almost the same information, what is characterized by the high correlation that exists among them (Schowengerdt, 2006).

According to the HSI-related literature (Zhang et al., 2016; Cao et al., 2017a), there are two methods that perform dimensionality reduction, namely, *feature extraction* (FE) and *feature selection* (FS).

2.3.1 Feature extraction

According to the FE approach, new features are generated by linear, or non-linear, combinations of the original ones (Theodoridis and Koutroumbas, 2008). The new features have a lower dimension, and normally they still retain much of the original data variance.

In Pattern Recognition, one seeks to extract features that provide discrimination amongst classes. However, it is important to balance the dimensionality reduction and its consequent loss of information, which can impair the classifier's discriminating power (Webb and Copsey, 2011).

Feature extraction techniques can be divided into two branches, *supervised* and *unsupervised*. Supervised methods are used for data redundancy reduction, aiming for a better classification accuracy. Discriminant Analysis Feature Extraction (Luo et al., 2015) is commonly used for this purpose, and it takes into consideration within-class and between-class scatter matrices, that is, labeled data are needed. Unsupervised FE approaches are used for the purpose of data representation. When it comes to multi- or hyperspectral images, Minimum Noise Fraction (Wu et al., 2013) and Principal Component Analysis (PCA) (Silva et al., 2013; Uddin et al., 2017) are largely used for dimensionality reduction by means of feature extraction.

2.3.1.1 Principal Component Analysis

PCA is an unsupervised technique, which is used for dimensionality reduction, lossy data compression, feature extraction and data visualization. It is also known as Karhunen-Loève transform (Duda et al., 2001).

PCA can be defined as the orthogonal projection of the data onto a lower dimensional linear space, such that the variance of the projected data is maximized (Bishop, 2006).

Let $\mathbf{X}$ be a hyperspectral dataset with samples $\mathbf{x}_i \in \mathbb{R}^d$, where d is the number of spectral bands. One seeks to project the data onto a space with dimensionality $d^{'} < d$, while maximizing data variance.

Let us consider the projection onto a one-dimensional space, that is, $d^{'} = 1$. One can define the direction of this space using a d-dimensional vector $\mathbf{u}_1$, which, for convenience, is set to be a unit vector, resulting $\mathbf{u}_1^T\mathbf{u}_1 = 1$. Each data point is then projected onto a scalar value $\mathbf{u}_1^T\mathbf{x}_i$. The mean of the projected data is $\mathbf{u}_1^T\bar{\mathbf{x}}$, where $\bar{\mathbf{x}}$ is given by

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i,$$

where N is the total amount of data samples. The variance of the projected data is calculated by

$$\frac{1}{N}\sum_{i=1}^{N}\{\mathbf{u}_1^T\mathbf{x}_i - \mathbf{u}_1^T\bar{\mathbf{x}}\}^2 = \mathbf{u}_1^T\mathbf{S}\mathbf{u}_1,$$

where S is the covariance matrix given by

$$S = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T.$$

We seek a reduced $d^{'}$-dimensional feature space that keeps the most original data variance. For that, the variance $\mathbf{u}_1^T\mathbf{S}\mathbf{u}_1$ of the new feature space is maximized with respect to $\mathbf{u}_1$. Naturally, this has to be a constrained optimization in order to prevent $\|\mathbf{u}_1\| \to \infty$. So, we condition $\mathbf{u}_1^T\mathbf{u}_1 = 1$, by using a Lagrange multiplier λ_1. Then, we proceed to an unconstrained maximization of $\mathbf{u}_1^T\mathbf{S}\mathbf{u}_1 + \lambda_1(1 - \mathbf{u}_1^T\mathbf{u}_1)$.

We set the derivative with respect to u_1 equal to zero, then there will be a stationary point when

$$\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1.$$

Consequently, $\mathbf{u}_1$ must be an eigenvector of $\mathbf{S}$. Multiplying on the left by $\mathbf{u}_1^T$, the variance of the projection is given by

$$\mathbf{u}_1^T \mathbf{S} \mathbf{u}_1 = \lambda_1,$$

and the variance of the reduced feature space will be a maximum when $\mathbf{u}_1$ equals the eigenvector with the largest eigenvalue λ_1. This eigenvector is known as the first principal component.

Let $\mathbf{U}$ be a $d \times d$ matrix whose columns are the $\mathbf{u}_j$ eigenvectors of $\mathbf{S}$, with $j = 1, 2, ..., d$. The multiplication $\mathbf{X} \cdot \mathbf{U}$ projects the original data set onto a new set of orthogonal axes, which are the eigenvectors of $\mathbf{S}$, with variance $v = \sum_{j=1}^{d} \lambda_j$, which equals the variance of the original data set $\mathbf{X}$.

In practice, however, most of the variance of the projected data are concentrated on the first m principal components, and it is given by $v^{'} = \sum_{i=1}^{m} \lambda_i$. Thus, it is possible to have a $d \times m$ matrix $\mathbf{U}^{'}$ whose columns are the first m eigenvectors of $\mathbf{S}$, with variance $v^{'} < v$. The multiplication

$$\mathbf{X} \cdot \mathbf{U}^{'} \tag{2.6}$$

gives a $N \times m$ matrix of the projected data, with $m < d$, yielding, consequently, a reduction in the dataset dimensionality.

It is worth noting, however, that the Equation 2.6 performs a linear transform on the original data set $\mathbf{X}$. This means that the original information—reflectance values for each pixel—is lost.

Feature extraction techniques, as a whole, change the original data representation, and this changing renders the post-processing analysis unfeasible, when the physical meaning of individual bands needs to be maintained (Feng et al., 2017), (Li and Liu, 2017).

Feature selection methods, on the other hand, can also be used as a dimensionality reduction tool. The difference in relation to FE approaches lies in the fact that FS techniques keep the original information—or bands—of the hyperspectral images.

2.3.2 Feature selection

As already mentioned, feature or band selection seeks to reduce the dimensionality of a data set without changing the original information.

When it comes to classification tasks, the focus of a BS method is on the bands that provide a good class separability, when the class labels are provided. For data sets without

class information, unsupervised approaches must be used, and, in this case, the BS process is focused on bands that best preserve the original data structure. For unsupervised approaches, the band selection procedure can follow one out of three methods:

- *Ranking*: the bands are quantified according to their importance. At the end of the process, the top-ranked bands are selected (Emmanuel Arzuaga-Cruz, 2003). More formally, let A be the original set of spectral bands, with cardinality $|\mathbf{A}| = d$. The band selection process is based on the assignment of weights p_j to each band $a_i \in \mathbf{A}$, defining their relative importance. Some methods give a weighed linear order of bands, whereas other approaches yield a subset of the original bands, whose weights p_j are binary (Molina et al., 2002). Normally, ranking-based methods do not take into account the correlation amongst spectral bands (Sun et al., 2017);
- *Clustering*: under this approach, the correlation amongst bands are considered. The most appropriate bands are selected by clustering (Datta et al., 2012); and
- *Searching methods*: The searching scheme evolves a good solution of band selection by optimizing a given measure (Xia et al., 2013).

In both supervised and unsupervised cases, the band selection algorithms preserve the HSI original information, which makes the results more interpretable, as indicated by (Li and Liu, 2017). For a judicious BS process, only the relevant bands should be selected, and the relevance of a spectral band is thus defined:

- A relevant band should provide useful information, based on a given criterion (Molina et al., 2002);
- A relevant band should not be redundant in relation to others already selected (Monteiro and Murphy, 2011); and
- A relevant band provides a better class separation in the feature space, and, consequently, improves the generalization power of the classifier—for supervised approaches (Duda et al., 2001).

In Figure 2.9, there are two 2D feature spaces for a hypothetical classification problem, in which two classes—red circle and blue asterisk—are to be distinguished by a classifier. In *Feature space 1*, the bands a_1 and a_2 are highly correlated. Thus, the band a_1, for example, would suffice to carry out the classification. Therefore, a_2 can be considered redundant information and can be, consequently, discarded from the original data set. In *Feature space 2*, the band a_1 provides a good class separability between the classes. However, a_3 is not a discriminant band for this classification task, that is, a_3 has irrelevant information and can be discarded.

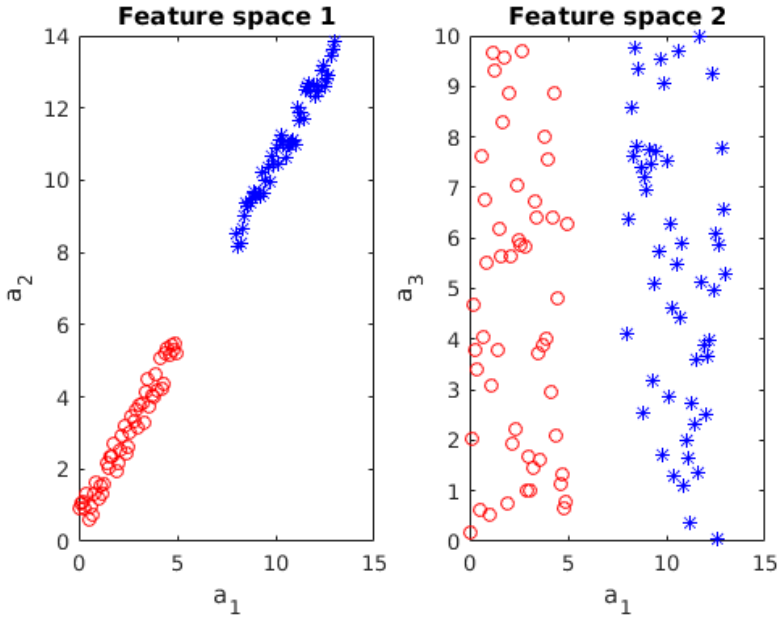


Figure 2.9: Two examples of feature spaces considering a binary classification. The band a_1 is relevant, whereas bands a_2 and a_3 are redundant and irrelevant, respectively.

2.3.2.1 Characterization

A BS method can be characterized under four aspects:

- Search organization;
- Generation of features subset;
- General schemes for feature selection ; and
- Evaluation measure.

Search organization: In relation to the number of bands that a method can analyze at a given instant, there are three possibilities:

- *Exponential search:* It can evaluate more than one band at a time. The optimal solution is achieved due to an exhaustive search. Sometimes, however, if the evaluation measure is monotonic, not all the possible combinations need be visited. In such a case, a branch and bound algorithm (Narendra and Fukunaga, 1977) may be used (Theodoridis and Koutroumbas, 2008);
- *Sequential search:* This method chooses one amongst all candidates to be the current state. It is an iterative method, so it is not possible to go back once one band is selected (Habermann et al., 2017b); and

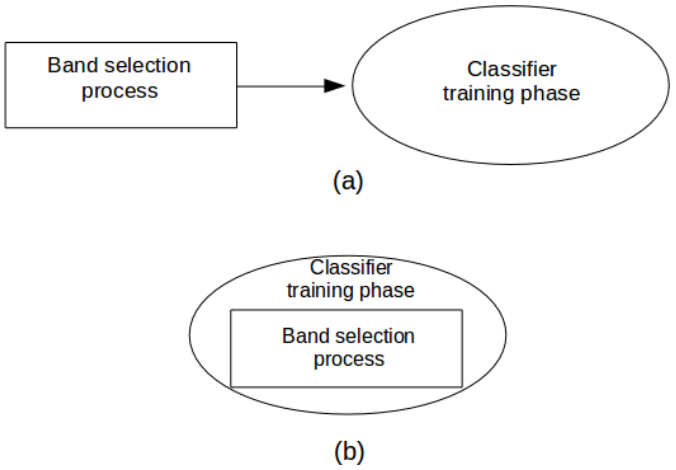


Figure 2.10: In (a): A flowchart of a typical filter-based band selection method. In (b): Wrapper approaches perform the band selection using the classifier.

- *Random search:* The intention is to use the randomness in order to avoid getting trapped in local minimum solution, and also to allow some movements towards states with suboptimal results (Kang et al., 2016).

Generation of features subset: The output of a band selection method is the subset $\mathbf{A}' \subset \mathbf{A}$, containing the selected spectral bands.

One approach starts with the original set $\mathbf{A}$ with d elements, and, at each step, one band is dropped out from $\mathbf{A}$ until the desired number of bands is achieved. This method is called *sequential backward selection.*

Another method, which is the reverse of the preceding procedure, is referred to as *sequential forward selection.* It starts from an empty set, and the best band—according to a specified criterion—is added to the set after each iteration. Sequential forward band selection is appropriate because in many cases the classifiers perform better with a small portion of the total number of available bands (Kohavi and John, 1997). Thus, it is faster to start with a empty set $\mathbf{A}^{'}$ and iteratively add bands to it, instead of discarding unwanted bands from the original data set.

The two afore-mentioned methods, however, suffer from the *nesting effect.* That is, once a spectral band is discarded or selected, depending on the method, it cannot be undone (Theodoridis and Koutroumbas, 2008). In order to mitigate such setbacks, *compound methods* make a combination of both forward and backward methods.

General schemes for features selection: For supervised approaches, the relationship between a band selection algorithm and the subsequent classifier can normally have two forms : *filter* and *wrapper* (Molina et al., 2002).

- *Filter:* Under this approach, the band selection takes place before the classifier training phase. That is, the BS method is used as a data preprocessing step. Consequently, the band selection method is independent of the classifier. The advantage of this strategy is its speed in relation to wrapper methods. Furthermore, filter-based band selection methods are most useful in situations in which wrapper algorithms may overfit due to small training data sets. The disadvantage of filter methods is that the band selection is not assisted by the classifier, what normally yields suboptimal results (Shahana and Preeja, 2016) (Molina et al., 2002). In Figure 2.10 (a), there is a simple flowchart depicting the relation between a filter-based band selection and a classifier. The horizontal arrow indicates that the training phase of the classifier starts after the band selection process is finished.

- *Wrapper:* In this case, the HSI bands are selected during the training phase of the classifier. For each feature a_i added to the bands subset $\mathbf{A}^{'}$—or discarded from it—, the classifier should be trained again in order to evaluate the bands in $\mathbf{A}^{'}$. Thus, the main disadvantage of this method is its heavy computational cost. Its advantage is the good overall classifier's accuracy (Shahana and Preeja, 2016) (Molina et al., 2002). A strong argument for wrapper approaches lies in the fact that the estimated accuracy of the classifier is the best available heuristic for measuring the degree of appropriateness of the selected bands. Moreover, different classifiers may yield different results using as input the same subset of selected bands. Thus, it is desirable that a classifier be able to select its own features (Kohavi and John, 1997). In Figure 2.10 (b), we see that both the band selection process and the classifier training phase are run at the same time.

Unsupervised approaches: As wrapper-based methods need the classes information to evaluate a given subset $\mathbf{A}^{'}$, there is no point talking about unsupervised wrapper-based algorithms, because, in principle, such a thing cannot exist. Thus, unsupervised band selection methods always follow a filter strategy.

Evaluation measures: There are some ways to assess how good a subset of selected bands is. Most evaluation measures such as *Divergence* and *Chernoff Bound and Bhattacharyya Distance* take into account the probability distribution of the classes (Molina et al., 2002). But, they are not easily computed.

For supervised approaches, a non-parametric measure called *Scatter Matrices* can be adopted (Theodoridis and Koutroumbas, 2008).

Scatter Matrices measure how well the data samples are scattered in the feature space. For this purpose, the following three matrices are defined:

- *Within-class scatter matrix:*

$$\mathbf{M}_w = \sum_{k=1}^{q} P_k \Sigma_k, \tag{2.7}$$

where q is the number of classes, Σ_k stands for the covariance matrix for class k, and P_k is the *a priori* probability of class k;

- *Between-class scatter matrix:*

$$\mathbf{M}_b = \sum_{k=1}^{q} P_k (\mu_k - \mu_0)(\mu_k - \mu_0)^T, \tag{2.8}$$

where μ_k is the mean vector of class k, and μ_0 stands for the mean vector of the whole data set; and

- *Mixture scatter matrix:* it is given by

$$\mathbf{M}_m = \mathbf{M}_w + \mathbf{M}_b. \tag{2.9}$$

The trace of M_m represents the sum of the features' variances around their respective global mean. Thus,

$$J = \frac{trace(\mathbf{M}_m)}{trace(\mathbf{M}_w)} \tag{2.10}$$

has larger values when:

- The data samples are well clustered inside their respective classes; and
- The clusters of different classes are well separated.

By using the index J, it is possible to have an early idea about the classifier's performance. Indeed, indices with large values indicate that the classes are well separated in the feature space. Consequently, the classifier will not have difficulties in finding appropriate separating boundaries amongst classes. In general, a good feature selection process is the first step towards a successful Pattern Recognition framework.

The use of the index J is, though, restricted to supervised approaches. When it comes to unsupervised BS methods, clustering algorithms may be used to compare the clustering results between $\mathbf{A}$ and $\mathbf{A}^{'}$, using, for example, the Adjusted Rand Index (Rand, 1971).

Most band selection works found in the literature, however, do not use any sort of index nor clustering methods to assess their proposed BS framework. They just compare their results with other approaches by using the same classifiers.

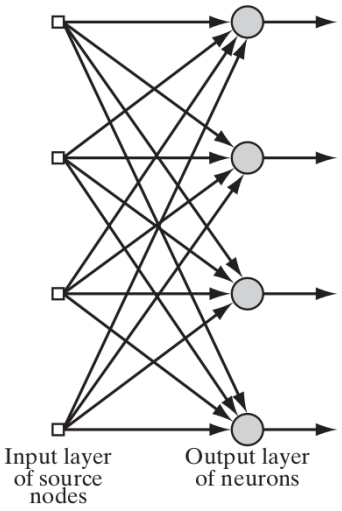


Figure 2.11: A single-layer neural network (Haykin, 2009).

2.4 Machine Learning Basics

Machine learning is a multidisciplinary research field that embraces different areas such as computer science, probability and statistics, psychology and brain science (Liu et al., 2018). Basically, the aim of machine learning is to automatically discover and acquire knowledge from datasets like humans do.

As an example of ML algorithm, artificial neural networks (ANN) imitate the functioning of the human brain, which is known for its highly complex, nonlinear and parallel computing power (Haykin, 2009).

In the sequel, we present the two ANN-based algorithms to be used in this thesis.

2.4.1 Single-Layer Neural Network

A single-layer neural network (SLNN) is the simplest neural net framework.

There is an input layer of source nodes that projects directly to an output layer of neurons, *i.e.*, it is a strictly feedforward neural network. It is illustrated in Figure 2.11 depicting a four-node case in both the input and output layers. Such an architecture is called a single-layer network, with the designation *single-layer* referring to the output layer of computation neurons. That is, one does not count the input layer due to the fact that no computation is performed there.

2.4.1.1 Positive aspect

One advantage of the SLNN is that it has less parameters to be adjusted in relation to a neural network with hidden layers. This characteristic is specially important when one

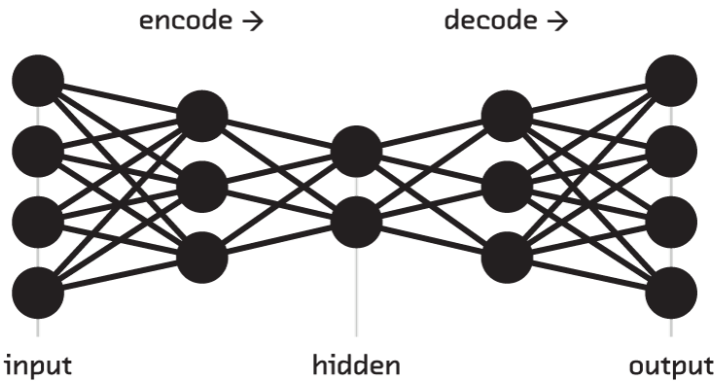


Figure 2.12: Example of an autoencoder architecture for dimensionality reduction.

works with hyperspectral data following a supervised approach, because assigning labels to such data is expensive. Thus, the less data needed, the better.

2.4.1.2 Negative aspect

Since the decision boundaries in the feature space are linear—a single or a set of hyperplanes—, it is important that the different classes be linearly separable.

2.4.2 Autoencoders

Autoencoders are a type of artificial neural network. They are used to learn efficient data codings in an unsupervised manner, because the output is the same as the input, thus, the class information is not needed.

The objective of an autoencoder is to learn a representation for a dataset, and it is oftentimes used for the purpose of dimensionality reduction. In this case, the hidden layers size must be smaller than that of the input layer, as illustrated in Figure 2.12.

2.4.2.1 Positive aspect

One advantage of the autoencoders is their unsupervised nature. The focus is done on the dataset structure, thus it is possible to draw conclusion about the importance of the input features—or spectral bands—, for example.

2.5 Hyperspectral Images Used

In this thesis, all the results and analyses are made by taking into account four hyperspectral datasets[6]. They are also used in several other scientific papers, thus it is possible to compare results.

1. **Botswana:** This image has a spatial resolution of 30 m. It has 145 bands covering the 0.4-2.5 μm range with a spectral resolution of 10 nanometer. The Botswana image comprises 1476 $\times$ 256 pixels, with 14 classes to be classified, as shown in Figure 2.13.

2. **Kennedy Space Center (KSC):** The spatial resolution of this image is 18 m. The KSC image comprises 512 $\times$ 614 pixels and has 176 spectral bands. There 13 classes to be classified, as illustrated in Figure 2.14.

3. **Indian Pines:** This scene was acquired by the AVIRIS sensor over the Indian Pines test site in north-western Indiana. It consists of 145 $\times$ 145 pixels and 224 spectral reflectance bands in the 0.4-2.5 μm wavelength range. The image contains two-thirds agriculture, and one-third forest or other natural perennial vegetation. Regarding the ground truth, there are 16 classes, as shown in Figure 2.15.

4. **Pavia University:** The Pavia University image has 103 spectral bands. It has 9 classes, as shown in Figure 2.16.

[6] http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

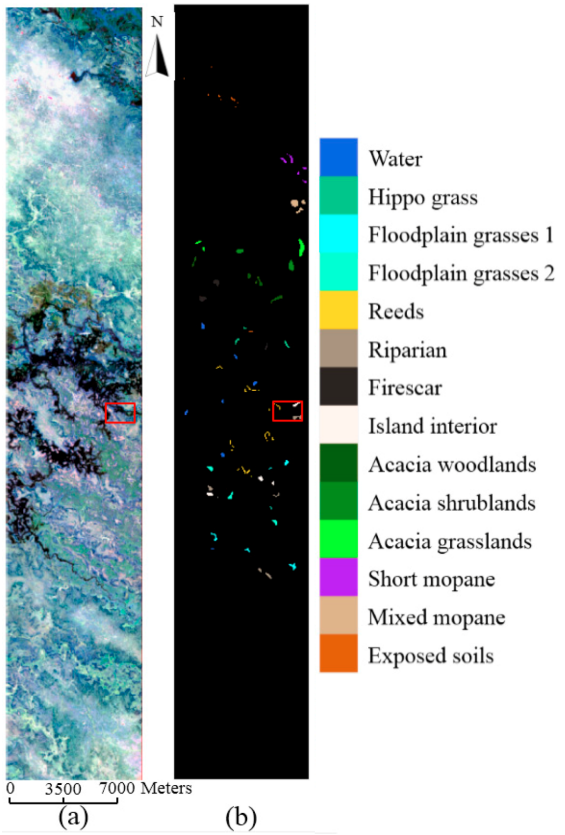


Figure 2.13: The Botswana image and its ground truth. In (a) a color composition, and in (b) the ground-truth classification map.

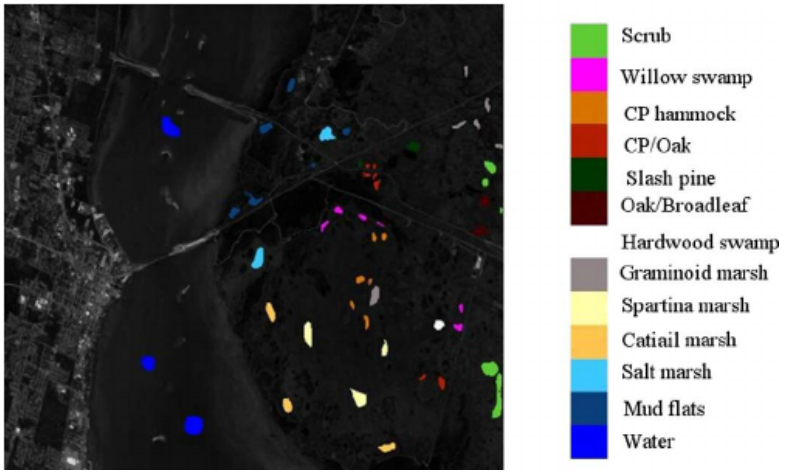


Figure 2.14: The KSC hyperspectral image and its classes.

Figure 2.15: The Indian Pines image and its classes. In (a), a color composition, and in (b) the ground-truth classification map.

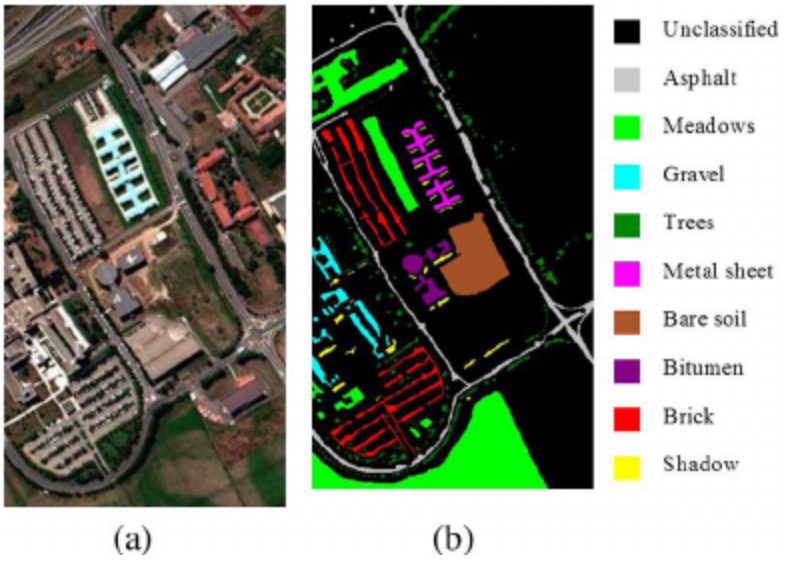


Figure 2.16: The Pavia University dataset. In (a) a color composition. In (b), the ground-truth.

Chapter 3

Literature Review on Hyperspectral Band Selection Methods

In this thesis, we propose three hyperspectral band selection frameworks. Two of them are unsupervised, and one is supervised.

As explained in Chapter 2.3.2, the strategy followed by supervised-based frameworks are different from that used by unsupervised approaches. Therefore, we present the BS state-of-the-art methods in two separate sections:

- Supervised approaches; and
- Unsupervised approaches.

3.1 Supervised approaches

According to Chapter 2.3.2.1, supervised band selection methods can be divided into *filter* and *wrapper* approaches.

3.1.1 Wrapper-based band selection frameworks

Wrapper methods perform the selection of bands based on the accuracy of the classifier. For example, (Monteiro and Murphy, 2011) proposes a band selection framework for hyperspectral images using boosted decision trees (DT). Several DTs are generated and the most recurrent features are selected.

In (Fauvel et al., 2015), it is proposed a method that iteratively selects spectral bands that will be assessed by a Gaussian Mixture Model classifier. The selection of bands is made by a method called nonlinear parsimonious feature selection. One positive aspect of the proposed framework is the selection of few bands—about 5% of the total amount.

The authors propose, in (Cao et al., 2016b), a BS framework based on local spatial information. Initially, the subset of selected bands is empty, and at each iteration a band is added to it. Then, this subset is used as input to a Markov Random Field-based classifier. Based on the local smoothness, the last inserted band is accepted or discarded. Due to the paucity of the training data, this method could not achieved better results than its competitors.

A framework based on Convolutional Neural Networks (CNN) and Distance Density (DD) is proposed in (Zhan et al., 2017). In this case, DD is used, instead of random search, for the selection of the candidate bands, which are assessed by a CNN classifier. Experiments show that the DD-based BS is faster than its random counterpart.

In (Bris et al., 2014), the authors address the problem of designing superspectral cameras dedicated to specific applications. Thus, they seek to find the best number of bands and the most useful spectrum regions suitable for their necessities. For this, they use two different band selection methods, namely, Sequential Forward Floating Search and Genetic Algorithm-based approach. The classifier used is SVM.

The authors proposed a wrapper-based framework using extreme learning machine as classifier (Su et al., 2017). Both the selection of bands and optimization of classifier's parameters are performed by an evolutionary optimization algorithm called Firefly.

In (Ma et al., 2017), it is proposed a framework that measure the band importance by means of gain ratio, then the bands subset is evaluated by polygon-based algorithm with SVM. The authors use not only spectral data, but also other types of features.

3.1.2 Filter-based band selection frameworks

When it comes to filter-based approaches, there are some different criteria for the band selection, such as, distances measures (Keshava, 2004), class separability measures (Cui et al., 2011), information, dependence (Camps-Valls et al., 2010), correlation, searching strategies (Jahanshahi, 2016) (Su et al., 2016) and classification measures (Habermann et al., 2017b).

For example, in (Damodaran et al., 2017), the authors propose a class separability-based approach. To be more precise, a new class separability measure based on surrogate kernel and Hilbert space independence criterion in the kernel Hilbert space is devised. Then, the proposed class separability is used as a objective function using LASSO optimization

(Hastie et al., 2015). The authors claim that this framework allows the selection of spectral bands to increase the class separability, thus avoiding an intensive subset search.

In (Jahanshahi, 2016), the author proposes a framework for hyperspectral band selection based on an evolutionary algorithm to perform the band selection, and then he uses a SVM classifier to assess the selected band subsets. The BS step is performed by Multi-Objective Particle Swarm Optimization, which ranks the bands according to the relevance between each band and the ground-truth information.

In this paper, the Firefly algorithm is used for the selection of bands (Su et al., 2016). The bands subsets found during the search are evaluated by Jeffreys-Matusita distance.

In (Monteiro and Murphy, 2011), the authors list some of the pros and cons of wrapper and filter methods for feature selection, and propose a filter-based forward selection algorithm that shares some common features with the wrapper method. The proposed framework uses boosted decision stumps. Over series of iterations, the features that correctly predict the classes' labels are chosen.

Rough Set theory (RST) (Pawlak, 1992), which has already been applied in image classification tasks (Pessoa et al., 2011), is used in a BS framework proposed in (Patra et al., 2015). The authors propose a BS framework based on RST, which is a paradigm to deal with vagueness, incompleteness and uncertainty of data. Firstly, informative bands are selected by RST, based on relevance and significance. Comparison of classification results shows that this method outperforms its competitors when a small number of bands is selected.

Normally, researchers working on the supervised band selection area have to devise methods capable of handling few training data. It may be a challenging issue for filter-based methods based on classification and class separability measures. For wrapper approaches, the paucity of training data poses an even worse problem, since such methods rely exclusively on classifiers to generate results. So, in order to alleviate this inconvenience, unlabeled instances are added to the training data, constituting, thus, a semisupervised model.

In (Bai et al., 2015b), the authors propose a framework based on spectral-spatial hypergraph model. Firstly, the method builds a hypergraph model using all data to measure the similarity amongst pixels. Then, a semisupervised learning algorithm is used in order to assign class labels to unlabeled samples. After that, the selection of bands is performed by a linear regression model that uses group sparsity constraint. Finally, the selected data are used to train a SVM classifier. This method has the advantage of using the spatial information of pixels.

In another paper, the authors assert that most band selection methods do their job taking into consideration all the classes at the same time, and this could result in sub-optimal band subset choice (Bai et al., 2015a). It is, then, proposed a framework that

selects bands for each class in a pairwise fashion. Initially, the Expectation-Maximization (ExpM) algorithm is used to measure the mean vectors and covariance matrices of each class. Then, for each pair of classes, Bhattacharyya distances are calculated and the best subset of bands are chosen. After that, a binary classifier is embedded into the ExpM process in order to get the posterior probabilities of instances, based on the selected bands. Finally, all the binary classifiers are fused.

Based on affinity propagation, which is an exemplar-based clustering method, the authors propose a semisupervised framework for the selection of bands (Jiao et al., 2015). Band correlation and band preference are also taken into account. In the paper, a new normalized trivariable mutual information is devised to measure band correlation. Due to the noisy bands the clustering step is disturbed, so a new method based on Statistics is devised, that is, the mean value of the neighboring bands correlation is compared to the correlation between two contiguous bands in order to find bands bearing low information. Finally, the framework is capable of selecting informative bands, whereas it can discard redundant ones.

3.2 Unsupervised approaches

In the literature, it is possible to find lots of BS works following many different perspectives and methodologies, such as *data manifold*, *data information analysis*, *graph theory*, *evolutionary computation* and *clustering*.

3.2.1 Data manifold

Due to the high HSI dimensionality, the different classes present in the image may lie in manifolds embedded in subspaces of the original feature space. Furthermore, it is also possible to explore the sparsity of the data set in order to find a more meaningful data representation. For example, in (Wang et al., 2016), the authors propose a new method in which they look for salient bands. The number σ of selected bands is user-defined. Then, the band selection algorithm has two steps. Firstly, β bands are selected by means of clone selection algorithm, which seeks to minimize the Euclidean distance amongst elements of the same class, whereas maximizing the distance of elements from different classes. After that, if $\beta < \sigma$, those β bands already chosen will serve as seeds to a Manifold Ranking (MR) algorithm. MR sorts the remaining bands, and the most dissimilar band is added to the β group. This step is repeated until $\beta = \sigma$.

In (Wang et al., 2017), the authors propose a BS framework based on sparsity. Initially, the most representative bands are obtained according to the correlation matrix, whereas the block-diagonal structure is measured to segment bands into subspaces. Then, a method for band selection based on trace LASSO and spectral clustering is used.

The authors of the paper (Sun et al., 2015) propose a method that initially represents data instances as sparse coefficient vectors by solving a L2-norm optimization using the least squares regression (LSR) algorithm. Then, a correct segmentation of band vectors is made using the resulting LSR matrix with sparse and block-diagonal structure. After that, a similarity matrix is constructed by angular similarity measurement, and then the size of the band subset is calculated by the distribution compactness plot algorithm.

In (Gan et al., 2017), the authors state that all HSI bands can be represented by a band subset. Thus, they propose a sparse representation of bands with row-sparsity constraint. Besides, a dissimilarity-weighted regularization term is integrated with the self-representation model, to avoid contiguous bands. The problem is solved by the alternating direction method of multipliers, and the representative bands can be chosen.

A fast and robust self-representation framework to select a band subset is proposed (Sun et al., 2017). It is assumed the separability structure of the spectral bands, thus the problem may be seen as non-negative matrix factorization. After that, an optimizing convex problem is addressed and augmented Lagrangian multipliers are used to select the band subset.

The authors in (Zhu et al., 2017) propose a BS framework that can capture the inter-band redundancy through low-rank modeling. Then, by using an affinity matrix and concepts of data quality, the most representative bands are selected.

In (Wang et al., 2015), a BS method based on column subset selection is proposed. By means of column subset selection problem, it is possible to select some bands maximizing the volume of the selected subset of columns. The high dimensionality decreases the contrast amongst bands, thus Manhattan distance is used to get a higher quality in the BS process.

As a last example on data manifold, in (Cao et al., 2016a), the authors propose a framework that removes low-discriminating bands that normally need to be discarded manually. Based on the spatial structure of the data set, it is possible to determine which bands have low-discriminating power. Then, a new clustering algorithm is proposed in order to define the optimum number of bands to be selected.

3.2.2 Data information analysis

Another criterion that can be used in BS strategies is the HSI data information analysis. For example, in (Sui et al., 2015), the authors propose a framework that integrates both the overall accuracy and redundancy. Thus, an optimization problem using adaptive balance parameter is devised to handle the trade-off between the overall accuracy and redundancy. Furthermore, an unsupervised overall accuracy prediction method was adopted.

In (Sun et al., 2014), the authors propose a framework that merges the concept of noise-adjusted principal components with maximum determinant of covariance matrix. A new index to measure the HSI quality is also proposed, taking into account signal-to-noise ratios (SNR) and correlation of bands. Based on the new index, the authors devise an unsupervised band selection method, which considers the quality of the data set as selection criterion. It selects bands with both high SNR and low correlation.

The authors in (dos Santos et al., 2015) propose a BS method based on the dissimilarity amongst neighboring bands. They use an intermediary representation named spectral rhythm, which can take advantage of a pixel sampling strategy, what ends up improving its efficiency without reducing the selected bands quality.

Another BS framework is based on information-assisted density peak index (Luo et al., 2017). It takes into account the intraband information entropy into the local density and intercluster distance to ensure cluster centers with a high quality. Besides, the channel proximity and band distance are integrated to control the local density compactness. The bands with top-ranked scores may get clear global distinction, good local density and also high informative quality.

In (Chang et al., 2017), the authors formulate the BS as a channel capacity problem. After constructing a band channel with the original bands. Then, some bands are selected by Blahut's algorithm, which iteratively finds a feature space that provides the best channel capacity. Thus, neither band prioritization nor interband decorrelation are required. Two iterative methods are devised to find the best band subset, which avoid an exhaustive search.

3.2.3 Graph theory

Using graph theory, in (Yuan et al., 2017) the authors propose a multigraph determinantal point process (MDPP). The aim is to capture the structure amongst bands and find the optimal band subset. For this, multiple graphs are designed to capture the intrinsic relationship amongst bands. Besides, the proposed MDPP is used to model the multiple dependencies in graphs, providing an efficient search strategy for the BS process.

3.2.4 Evolutionary computation

Evolutionary computation with optimization have been largely used by BS methods. For example, in (Xu et al., 2017), the authors propose an incorporated rank-based multiobjective band selection framework, to avoid conflicting objective functions, such as Jeffreys-Matusita (JF) and Bhattacharyya distances. During the processing, the spectral bands are transformed into binary vectors, whose elements are subjected to flipping with a certain probability.

In this approach, the authors propose a framework that handles two conflicting objective functions (Gong et al., 2016). One function is designed to represent the information contained in the selected bands, by means of entropy. The other function is set as the number of selected bands. Both objective functions are optimized simultaneously by a multiobjective evolutionary algorithm.

The authors of the paper (Su et al., 2014) propose a framework for band selection which employs two objective functions using JF. During the search process, the spectral bands are treated as firefly variables.

In (Zhang et al., 2017a), a framework for band selection based on fuzzy clustering and swarm optimization in proposed. The authors devise a modified fuzzy clustering method for band selection, whose drawbacks are alleviated by swarm optimization.

Memetic algorithms (MA) are also used in BS frameworks (Zhang et al., 2017b). Firstly, MA is used to select a subset of spectral bands. Also, a objective function is designed to select bands considering both bands information and redundancy deduction. The authors claim that this method is not only computationally faster than exhaustive search approaches, but also has comparable performances.

3.2.5 Clustering

Finally, clustering techniques can also be used in band selection methods. For instance, in (Datta et al., 2012) the authors propose a framework that removes redundancy amongst bands by means of clustering. Then, from each cluster one representative band is selected. After that, the bands are ranked according to their classification capabilities.

A framework based on dual clustering that takes into account the contextual information is also proposed (Yuan et al., 2016). For this, a novel descriptor that reveals the image context is devised, in order to select the representatives of each cluster, taking into consideration the mutual effects of each cluster.

Chapter 4

Supervised Band Selection using Single-Layer Neural Network

In this chapter, the hyperspectral band selection problem will be addressed under a *supervised* approach.

This means that we need to have the class information of the training samples in order to perform the BS process. As already explained in Section 2.3, assigning class labels to pixels is not only expensive, but also highly time consuming. Thus, we devised a band selection algorithm that works well with very few training data.

The supervised framework we propose is a new BS method based on single-layer neural networks, acting as a binary linear classifier. The band selection is done in a class-wise fashion, that is, the selection of bands is based on the separability of each class in relation to the remainder of the data set. As a result, the most representative bands of each class can be selected.

By comparing the proposed method with other three BS frameworks it is possible to see that our framework has a good performance even with greatly reduced training data size.

4.1 Motivation

There are some benchmark hyperspectral images available on internet[1]. It is worth noting that not all the pixels have their respective class information, but it is still possible to perform supervised methods in general—be it for classification or feature selection/extraction—, since the ratio N/d be appropriate for the application at hand.

[1] http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

The proposed method is based on single-layer neural networks with only one neuron in the output layer, *i.e.*, it's a very simple neural net architecture. Thus, it is supposed to work well even when there are few training data available.

4.2 Proposed Framework

4.2.1 Definitions

Let $\mathbf{X}$ be the data set corresponding to a hyperspectral image, where each element of $\mathbf{X}$ is a tuple $(\mathbf{x}_i, y_i)$, and $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ is a vector containing a spectral signature and $y_i \in \{1, 2, ..., q\}$ is its corresponding class—or label; where q is the number of classes c_j, with $j = 1, 2, ..., q$, and d is the dimensionality of the feature space $\mathbf{F}$.

Let $\mathbf{S}$ be the set of selected bands, and $\mathbf{G}$ the set containing bands highly correlated to those in $\mathbf{S}$. Let $\mathbf{A}$ be the set containing the original spectral bands a_k, with $k = 1, 2, ..., d_0$; where d_0 is the original quantity of bands. And let γ be the previously determined number of bands to be selected.

Finally, let $f : \mathbf{F} \longrightarrow t$ be a single-layer neural network, where $t = \{0, 1\}$, and the feature space $\mathbf{F}$ initially equals $\mathbf{A}$ and is updated by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$ after each iteration. The input to f is a vector $\mathbf{x}$ and its output is a scalar given by

$$\hat{t} = f(z) = \frac{1}{1 + e^{-z}}, \tag{4.1}$$

with $z = \mathbf{w}^T\mathbf{x} + b$, where $\mathbf{w} \in \mathbb{R}^{d\times 1}$ and b are the weights and bias of the neural network, respectively.

According to Equation 4.1, $\hat{t} \in [0, 1]$, and in order to assign a binary value to it, the following criteria are adopted:

$$\text{If } z < 0 \implies f < 0.5 \implies \hat{t} \leftarrow 0, \text{ and} \tag{4.2}$$

$$\text{if } z \geq 0 \implies f \geq 0.5 \implies \hat{t} \leftarrow 1. \tag{4.3}$$

From Equations (4.2) and (4.3), it is clear that the sign of z determines whether an input vector is to be assigned to class 0 or to class 1.

As the input data is normalized into $[0, 1]$, the coefficients $w_l \in \mathbf{w}$, with $l = 1, ..., d$, in the hyperplane equation

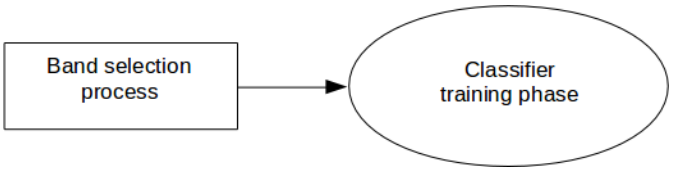


Figure 4.1: Flowchart of a filter approach. The band selection takes place before the training phase of the classifier.

$$z_i = x_i^1 w_1 + x_i^2 w_2 + ... + x_i^d w_d + b \tag{4.4}$$

play an important role in determining the signal of z, and, as a consequence, the estimate $\hat{t}_i$ for $\mathbf{x}_i$.

The cost function of this single-layer network is quadratic, and the training is done by stochastic gradient descent, using the back-propagation algorithm (Haykin, 2009).

4.2.2 Description

The proposed method follows a filter approach, that is, it takes place before the classifier[2] undergoes its training phase, as illustrated in Figure 4.1.

Our framework is also based on a *sequential forward selection* approach, meaning that it starts with an empty subset, *i.e.*, $\mathbf{S} = \emptyset$, to which the bands selected from $\mathbf{A}$ will be added. As it is based on single-layer neural networks, we shall call it `SLN`, whose characteristics are described below.

4.2.2.1 Iterations

`SLN` is an iterative class-oriented band selection method that starts at class c_1 and ends at the last class, that is, c_q. At each iteration a binary classification problem is to be solved by the function f. At iteration j, for $j = 1, 2, ..., q$, two groups, *class j-vs-all*, are to be separated by a hyperplane defined by $\mathbf{w}$ and b, where the *class j* is composed of all $\mathbf{x}_i \in \mathbf{X}$ with $y_i = j$, and the remainder of the data is a balanced composition of all $\mathbf{x}_i \in \mathbf{X}$ whose $y_i \neq j$. That is, there are two groups of data samples whose cardinalities are the same. The total amount of iterations is always q.

[2]Naturally, a single-layer neural network is a classifier. However, in this work it is used during the band selection process. The *classifier* referred to by the filter approach is used to perform the final data classification using the selected bands as input. In this work, KNN and CART are used as classifiers, as we shall see later on.

4.2.2.2 Selection of bands

After the training of the single-layer network, it is possible to assign degrees of pertinence to all $a_k \in \mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$. Since every element x^l of $\mathbf{x}$ is directly linked to w_l, the value of w_l is a token for the band a_l. This is the reason why we choose a single-layer neural network for BS. Deeper architectures would create more complex relationships among weights and spectral bands, thus, the consequent band selection based on weights magnitudes would not be a straightforward task. Besides, architectures with hidden layers have more parameters to be adjusted, and it would demand more training data.

In a one-vs-all scheme, in many cases the two groups are linearly separable. There are other situations in which a hyperplane cannot separate the two classes, however it may still provide a reasonable separation. To illustrate this, in Figure 4.2, all the 14 classes present in the Botswana image are displayed in a one-vs-all fashion. We reduced the data dimensionality by using the first two principal components from Principal Components Analysis. The data samples in green color represent the class j under analysis, for $j = 1, ..., 14$, and the blue points stand for a balanced composition of the remaining classes. That is, the number of green and blue points are the same. Thus, for each frame in Figure 4.2, the number of green and blue points is practically the same. The red line segment in each frame represents the separating boundary provided by a single-layer neural network. In Figure 4.3, the same analysis is done, using the Indian Pines dataset.

Note that our interest is not on the hyperplane defined by z in Equation 4.4, but on how the weights affect the sign of z, as in Equations 4.2 and 4.3. The signal of z ends up determining the binary class of an input vector $\mathbf{x}$. Thus, our focus is not on the classification itself, but on the *behavior* of the features.

In Equation 4.4, the largest and the smallest—the negative value with the biggest magnitude—weights make the most important contributions to the sign of z. For this reason the bands corresponding to these weights are also considered the most important, and, consequently these bands are added to the set $\mathbf{S}$. This strategy to select bands has, at least, two advantages:

- This method selects the most discriminant bands for the one-vs-all cases; and
- It is possible to assign either 0 or 1 to the class of interest during the training of the single-layer neural network.

After each iteration, the feature space $\mathbf{F}$ is updated by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$, and this procedure is repeated until the last class is reached. In this way, each class *chooses* its most discriminant bands.

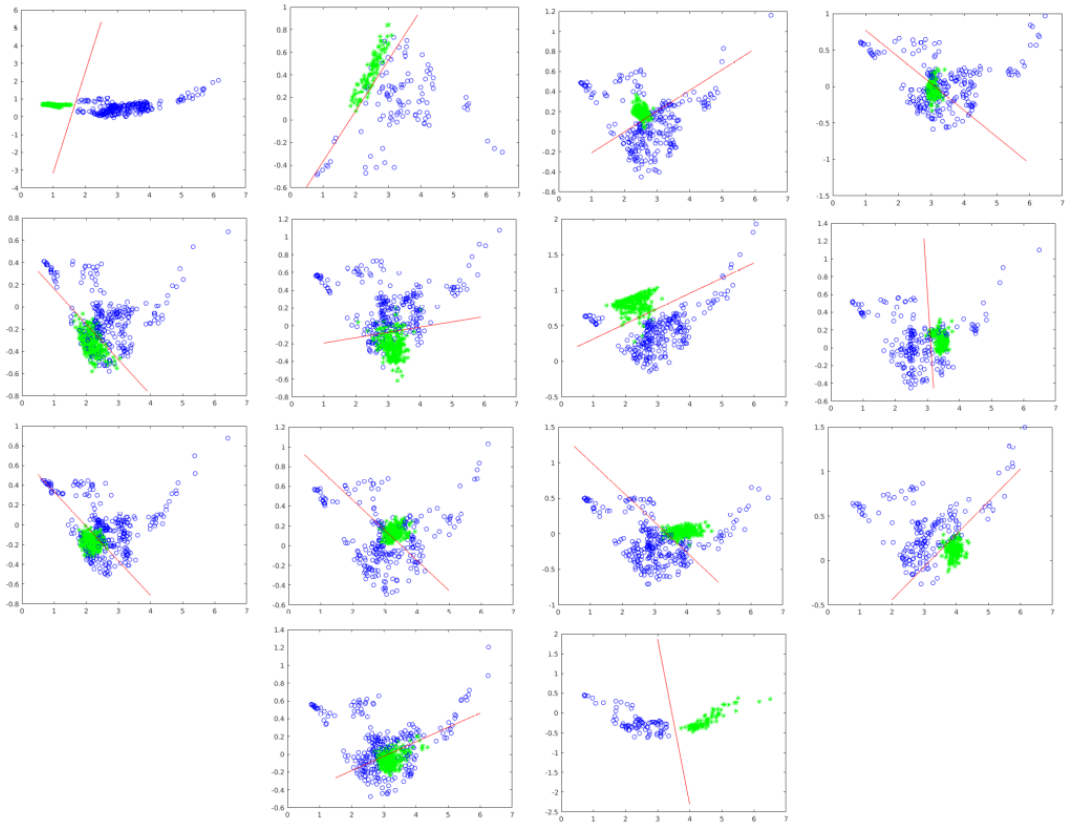


Figure 4.2: One-vs-all illustration for each class of the Botswana image. In each frame, the horizontal and vertical axis are, respectively, the first and the second principal components. The green dots represent the class under scrutiny, whereas the blue ones stand for data samples of the remaining classes. The red line segment is given by a single-layer neural network.

4.2.2.3 Avoiding highly correlated bands

By definition, the bands of a hyperspectral image are contiguous, which implies a high correlation between neighboring bands (Schowengerdt, 2006). In Figure 4.4 this fact is depicted, emphasizing the high correlation amongst neighboring bands, taking the band 72 as reference.

Based on this fact, it is possible to devise a method to avoid the selection of highly correlated bands. Thus, for each band $a_k \in \mathbf{F}$ we build in a off-line fashion a vector $\mathbf{v}_k$, in such a way that its elements are the bands indices in a descending order in relation to the correlation to the band a_k. That is, $\mathbf{v}_k(1)$ is the index of the band the most correlated to a_k.

Finally, the following procedure is adopted:

- At a given iteration, some band a_k will be selected, so $\mathbf{S} \leftarrow a_k$;
- $\mathbf{G} \leftarrow a_{\mathbf{v}_k(1)}$, where $\mathbf{G}$ is initially an empty set; and

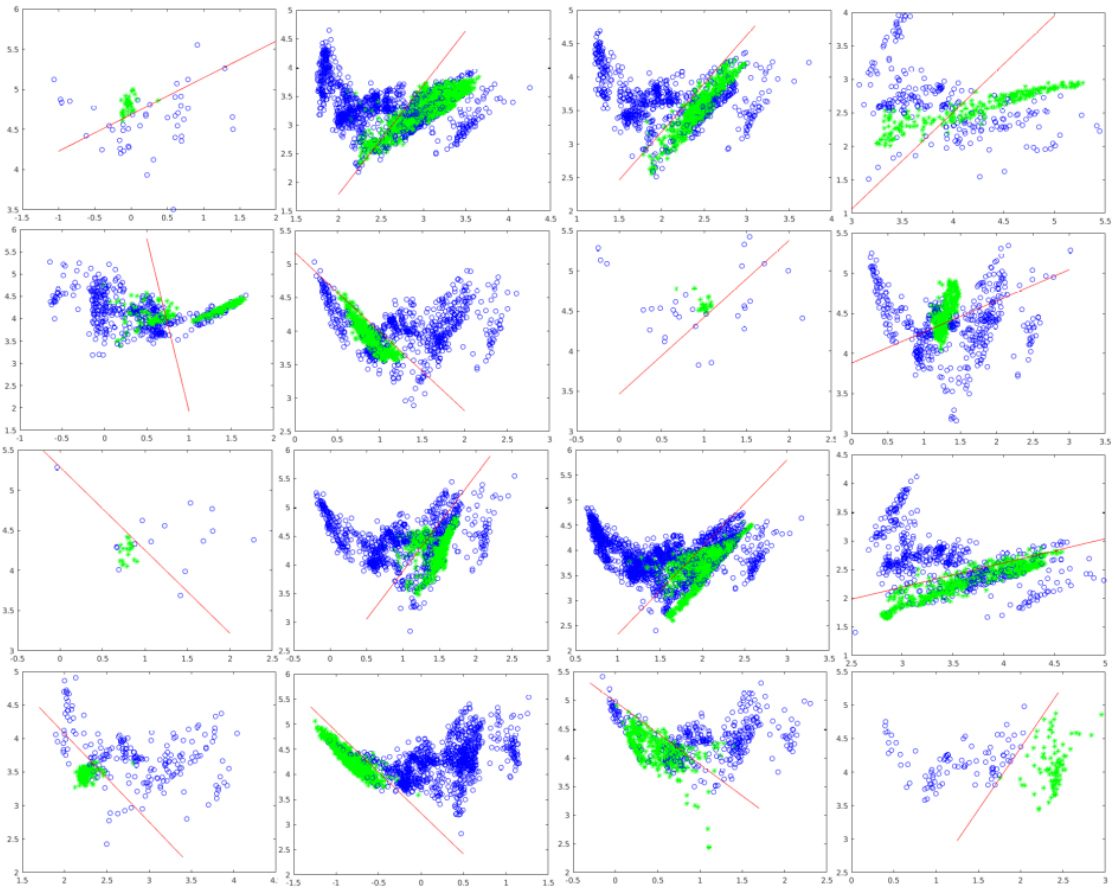


Figure 4.3: One-vs-all illustration for each class of the Indian Pines dataset. In each frame, the horizontal and vertical axis are, respectively, the first and the second principal components. The green dots represent the class under scrutiny, whereas the blue ones stand for data samples of the remaining classes. The red line segment is calculated by a single-layer neural network.

- After the iteration, the feature space $\mathbf{F}$ is updated by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$.

It is worth noting that only $a_k \in \mathbf{S}$ will be the selected bands, and the bands in $\mathbf{G}$ are discarded.

4.2.2.4 Number of bands selected

The number γ of selected bands is user-defined[3]. Thus, for each class c_j, with $j = 1, 2, ..., q$, $round(\gamma/q)$ bands can be selected, where $round()$ is an operator that rounds up the value of its argument to the next integer. Sometimes, at the end of the BS process, $|\mathbf{S}| > \gamma$. In such a case, it is possible to use the k-Means algorithm (Su et al., 2011) to select the γ sought bands.

[3]Note that for the purposes of this thesis, we assume that the user has the appropriate knowledge to determine the number of bands to be selected.

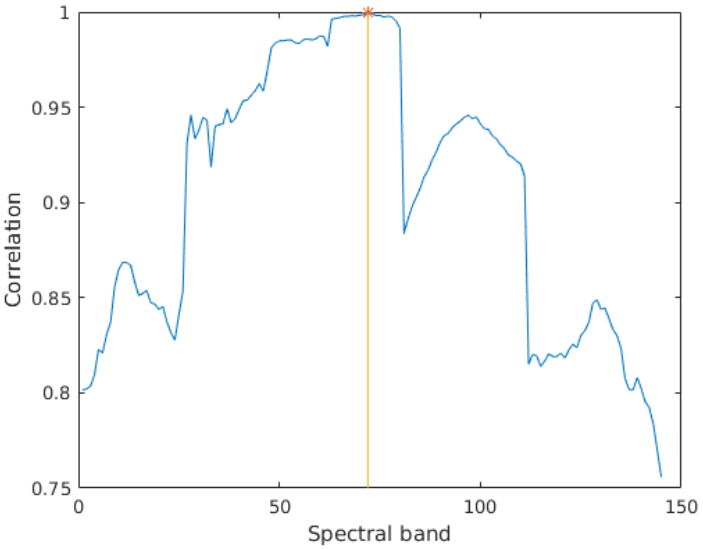


Figure 4.4: Correlation values of spectral bands in relation to the band 72 of the Botswana image. It is evident the higher degree of correlation amongst neighboring bands compared to more distant ones.

It is worth noting that, at the end of the proposed band selection process, $|\mathbf{S}| = |\mathbf{G}|$. Thus, $|\mathbf{S}|+|\mathbf{G}| \leq d_0$, and, consequently, $|\mathbf{S}| \leq d_0/2$ is a requirement that must be met. In other words, this means that *the maximum amount of bands that the proposed method is capable of selecting is the half of the total amount of original bands.* In practice, however, this limitation is not supposed to impair a BS process due to, at least, two reasons: *i)* the high correlation amongst neighboring bands, permitting a certain band to bear its neighbors's information; and *ii)* in order to avoid either heavy processing burden or Hughes phenomenon (Sun et al., 2016), it is desirable to greatly decrease the dimension of the input data.

There is no minimum limit of bands to be selected. However, when $\gamma < q$, not all the classes can contribute to the band selection. In this case, suboptimal results may be achieved.

Algorithm 1 summarizes the steps followed by our `SLN` approach.

Figure 4.5 depicts the proposed method. For each class c_j, Figure 4.5 (a), a binary one-vs-all classification is performed between class c_j and the remainder of the data set. In Figure 4.5 (b), the bands a_k corresponding to the largest and smallest weights are then added to set $\mathbf{S}$, and, in Figure 4.5 (c), the highly correlated bands $a_{\mathbf{v}_k(1)}$ are added to set $\mathbf{G}$. Finally, in Figure 4.5 (d), the feature space $\mathbf{F}$ is updated to $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$. This process iterates from the first class, c_1, until the last class, c_q.

Algorithm 1 Proposed band selection framework

Input : $\mathbf{X}$, $\mathbf{F} = \mathbf{A}$, $\mathbf{S} = \emptyset$, $\mathbf{G} = \emptyset$, q and γ.
for $r = 1 : q$ **do**
    Assign the value 1 to samples that belong to class c_r, and the value 0 to a balanced composition of the remaining classes
    Use $f : \mathbf{F} \longrightarrow \{0, 1\}$ to find a separating hyperplane z between class c_r and the remaining classes of the data set
    Identify the $round(\gamma/q)$ bands $a \in \mathbf{F}$ related to the largest and smallest $w \in \mathbf{w}$, and insert their indices in the temporary set $\mathbf{S}_0$
    for k=1:$round(\gamma/q)$ **do**
        $\mathbf{S} \leftarrow a_{S_0(k)}$, and $\mathbf{G} \leftarrow a_{\mathbf{v}_{S_0(k)}(1)}$
    end for
    $\mathbf{S}_0 = \emptyset$
    $\mathbf{F} = \mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$
end for
if $|\mathbf{S}| > \gamma$ **then**
    Use k-Means algorithm to select γ bands
end if
Return: $\mathbf{S}$

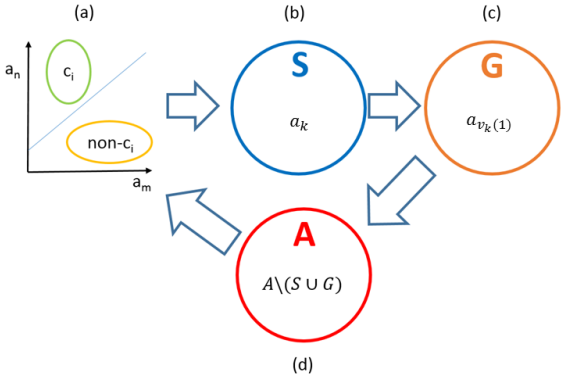


Figure 4.5: Flowchart of the proposed SLN method.

4.3 Experiments and Results

In this Section, the bands selected by the proposed method and their subsequent classification accuracies by two classifiers are shown and analyzed.

Before that, the data sets used in this thesis were presented. Also, the classifiers used to obtain the results will be shortly described, as well as the three competitors used to compare results.

Three hyperspectral images will be used: *i)* Botswana; *ii)* Indian Pines; and *iii)* Kennedy Space Center.

4.3.1 Classifiers

One way to compare the output of the different band selection methods is to perform a classification of the datasets using their respective selected bands as input.

We do not seek to find the best classifiers for a given task and dataset (Damodaran et al., 2016). Since the focus of this thesis is on the relative comparison amongst different BS methods, we restrict the analysis to only two classifiers that are largely used in hyperspectral images classification. They are k-Nearest Neighbors (KNN) and Classification and Regression Trees (CART) (Theodoridis and Koutroumbas, 2008), (Duda et al., 2001).

4.3.1.1 KNN

k-Nearest Neighbors is a nonparametric classifier. It takes into consideration the spatial relationship amongst data points. Each new entry is classified according to its k nearest neighbors in the feature space, being assigned the label of the majority. Different k values lead to different outcomes, so, in order to find the most suitable number of neighbors, for each $k_n = n$, with $n \in \{1, 2, ...15\}$, the KNN classification using the three images described in Section 4.3 was performed. The mean results are shown in Figure 4.6. All the spectral bands were used in order not to favor any BS method compared in this work. The best accuracy was achieved with $k = 13$ using Euclidean distance, thus we will keep this setting throughout this chapter.

4.3.1.2 CART

Classification and Regression Trees is a nonparametric classifier based on Decision Trees. Basically, it defines features thresholds in order to split the feature space into homogeneous regions. For the classification of a new entry, its features are analyzed according to the previously learned thresholds, and its label will be assigned according to the feature space region this entry falls into.

4.3.2 Related Works for Comparison

Results obtained by the method proposed in this chapter are compared with results from three other supervised band selection approaches. The framework we propose is based on Machine Learning, so, for the sake of a more diverse comparison, we chose methods

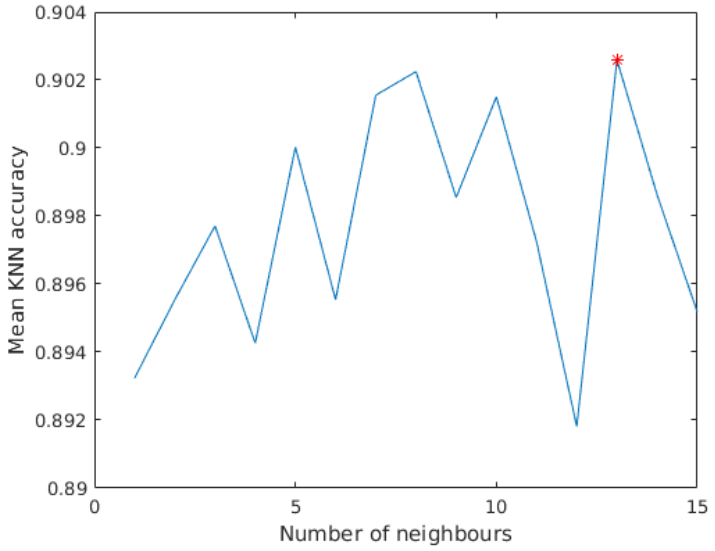


Figure 4.6: KNN mean classification accuracies using the three images with different numbers of neighbors.

from three different branches, namely Statistical Models, Evolutionary Algorithms (EA) and Image Processing (IP).

4.3.2.1 Statistical Models-based Approach

In the first method (Feng et al., 2017), the authors propose a framework that uses non-homogeneous hidden Markov chains (NHHMC) and wavelet transform as tools for band selection. Spectral signatures are first processed by wavelet transform, which is capable of encoding compactly the locations and scales at which the signal structure is present. A zero-mean Gaussian mixture model is then used to provide discrete values for the wavelet coefficients. The more Gaussian components used, the greater the detail in the descriptors generated. However, the authors demonstrate that the accuracy obtained using just two Gaussian components is usually only 1% less than where multiple Gaussian components are used and therefore in this thesis, we have limited ourselves to two in order to reduce computational load. Since wavelet coefficient properties can be accurately modeled by a NHHMC, this hidden Markov Chain is also used. Processing yields a set of candidate bands, among which those with the highest score in terms of correlation form the final output of this framework. The authors use the Support Vector Machine (SVM) classifier to measure the accuracy of the resulting bands.

This is a filter-based method, because the selection of bands is done before the classification is performed by SVM. It is also a sequential search algorithm, performing a sequential forward selection.

4.3.2.2 EA-based Approach

The second method is based on EA (Saqui et al., 2016). More precisely, it uses a Genetic Algorithm (GenA). Normally, GenA methods use three operators: selection, crossover and mutation of individuals. Each element of the population is a binary vector $\mathbf{v}_i \in \mathbb{N}^{1\times d_0}$ —also called chromosome —where d_0 is the number of spectral bands in the image. Each component, or gene, v_i^k in $\mathbf{v}_i$ indicates the presence of the k^{th} band when $v_i^k = 1$. At each generation of the algorithm, the population is evaluated by the fitness function, which is a Gaussian Maximum Likelihood Classifier. This classifier classifies the image using the bands indicated by the different vectors $\mathbf{v}_i$, and the classification accuracy is used as the fitness of the chromosome. After each iteration, the best chromosomes are retained, following which they are subjected to crossover and mutations. The whole process is repeated until a predefined number of generations is reached. At the end, the selected fittest chromosome is the one with the selected bands.

Clearly, this is a wrapper-based method, *i.e.*, the process of selecting features is embedded in the classifier. It also performs a random search, in virtue of the intrinsic characteristics of EA-based methods.

4.3.2.3 Image Processing-based Approach

The third competitor (Cao et al., 2017b) is a semi-supervised method that uses Image Processing tools in its wrapper-based band selection framework. Firstly, it trains a SVM classifier based on labeled instances, then this classifier assigns class label to unlabeled data, which end up having wrong labels —or pseudo ground-truth. After that, the resulting classification map is improved by an IP-based edge-preserving filter. At this point, there are two data sets: one with the original ground-truth information, and other with calculated pseudo ground-truth information. Then, for each combination of candidate bands to be selected, another SVM is trained using the data with original ground-truth, and its accuracy is assessed by the data set with pseudo ground-truth. By testing several band combinations, it is possible to select the one with the highest classifier accuracy.

4.3.3 Results

Supervised approaches rely on labeled training data to obtain their results. As already stated, the assignment of labels to pixels is an expensive task. So, in this thesis, the proposed BS framework uses only a small percentage of the available training data to get its results.

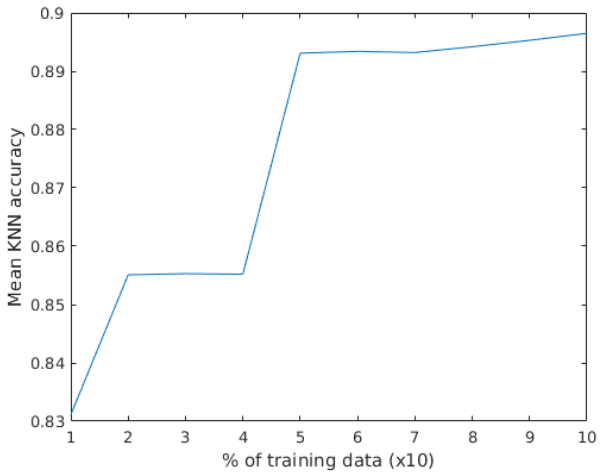


Figure 4.7: KNN accuracies with different percentages of Botswana training data.

4.3.3.1 Percentage of the training data used

We ran the proposed band selection algorithm ten times, with different fractions of the available training data. For each percentage $p_s = s/100$, with $s \in \{10, 20, ..., 100\}$, the proposed BS method has been used to select bands, and the cardinality of its training set was $|\mathbf{X}| \times p_s$. Thus, for each p_s there is a set S_s of selected bands. Each $\mathbf{S}_s$ has 50 selected bands.

Using all the three images, Figure 4.7 shows how the KNN classifier's mean accuracies change with different quantities of training data. In general, there is a tendency of getting higher accuracies as the amount of training data increases. With 20% of the available training data, the proposed algorithm had an accuracy similar to that of 30% and 40%. As it is desirable to work only a small fraction of the available training data, we chose to use only 20% of the data to select bands using the proposed method.

4.3.3.2 Methods Comparison

The bands selected by each competitor will be compared. We measure the validity of each subset of selected bands using two classifiers, namely, KNN and CART, which are largely used in classification of hyperspectral images (Wang et al., 2017, 2016; Zhu et al., 2017; Zhang et al., 2017b).

The classifiers are run using MATLAB. For the KNN classifier, we used `fitcknn` command, from Statistics and Machine Learning toolbox. For CART, `fitctree` was used, also from Statistics and Machine Learning toolbox. The proposed BS framework was also implemented in MATLAB, and we used the `trainSoftmaxLayer` command, from Neural Network Toolbox, for single-layer neural network, with 2000 training epochs—normally

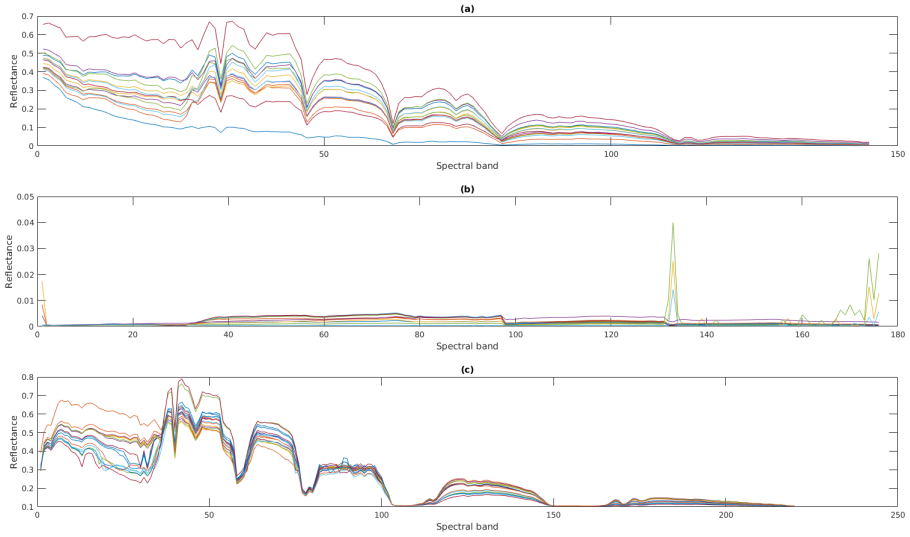


Figure 4.8: (a) Mean spectral signatures for the Botswana image. (b) Mean spectral signatures for the KSC image. (c) Mean spectral signatures for the Indian Pines image.

the training phase stopped before this, so other training epochs quantities were not tested.

First, it is important to analyze the data set to have an idea of the complexity of the problem. Figure 4.8 shows the mean spectral signature of each class. In Figure 4.8 (a), for example, it can be seen that in some regions the spectral signatures are further apart than in other regions of the electromagnetic spectrum. Since each spectral signature corresponds to a class, one might conclude intuitively that the bands where the curves are more spread out will provide a better class separability. In Figure 4.8 (b), the spectral signatures of classes are practically juxtaposed, except in a handful of regions. This can prevent the classifier from achieving a good outcome.

The competitor described in Section 4.3.2.1 will be called `NHMC`. The method described in Section 4.3.2.2 will be called `GA`. Finally, the algorithm described in Section 4.3.2.3 will be referred to as `ICM`.

All the classifiers results that will be exhibited in this chapter are the mean values of 10 runs. Standard-deviation values are also calculated.

Botswana Table 4.1 shows the bands selected by the proposed method, `SLN`. In Table 4.2, which exhibits the KNN mean accuracies and their respective standard-deviation for Botswana image, we see that the `SLN` method outperformed its competitors with 50 bands. Figure 4.9 gives a plot of the results of the four methods. It is then possible to see the advantage of the proposed method with 50 selected bands.

Table 4.3 shows the CART mean accuracies and standard-deviations for Botswana image. The proposed `SLN` method got the best results with 10, 20 and 50 bands. Figure 4.10 gives a visual perspective about the results, and it is possible to notice the superior results.

Table 4.1: Selected bands for the Botswana image.

10 bands	1 3 20 27 32 37 43 50 54 68
20 bands	1 4 7 16 20 21 24 26 31 35 37 44 47 50 57 59 62 69 93 98
30 bands	1 4 6 10 13 16 21 24 29 33 35 38 41 43 47 49 50 55 59 61 67 71 75 84 89 93 107 113 122 125
40 bands	1 4 6 10 11 13 16 21 23 24 25 27 29 32 33 35 38 40 41 43 47 49 50 52 54 55 59 61 67 71 74 75 84 89 93 106 107 112 122 125
50 bands	1 2 6 13 15 19 20 23 24 26 27 29 32 34 35 41 43 44 47 49 50 52 53 54 55 61 63 64 65 66 69 71 73 74 75 79 84 87 94 96 98 103 106 107 112 115 117 122 125 131

Table 4.2: KNN results for Botswana image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
`SLN`	88.50%	1.19%	88.19%	1.34%	89.73%	0.76%	90.55%	0.76%	**90.35%**	0.73%
`NHMC`	89.32%	1.17%	89.49%	0.57%	**90.38%**	1.86%	89.22%	0.37%	90.14%	0.37%
`GA`	**89.43%**	0.74%	**89.97%**	0.93%	89.97%	0.49%	**90.93%**	1.10%	89.53%	0.10%
`ICM`	89.05%	0.51%	89.08%	0.21%	89.15%	0.31%	88.60%	0.53%	88.40%	0.36%

Table 4.3: CART results for Botswana image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
`SLN`	**85.63%**	1.36%	**85.22%**	1.26%	83.16%	1.39%	84.80%	1.20%	**85.83%**	1.43%
`NHMC`	84.46%	0.63%	83.61%	2.12%	85.08%	1.40%	83.37%	0.72%	85.25%	1.37%
`GA`	84.80%	0.18%	84.67%	0.51%	**85.56%**	1.40%	**85.63%**	0.62%	85.69%	1.47%
`ICM`	85.39%	0.16%	83.78%	1.82%	84.70%	0. 36%	85.63%	0.74%	84.70%	0.98%

KSC In Table 4.4 all the bands selected by the `SLN` approach are displayed.

Table 4.5 shows the KNN mean accuracies and standard-deviation for KSC image. The proposed method `SLN` got the best results with 30, 40 and 50 bands. Figure 4.11 shows the results.

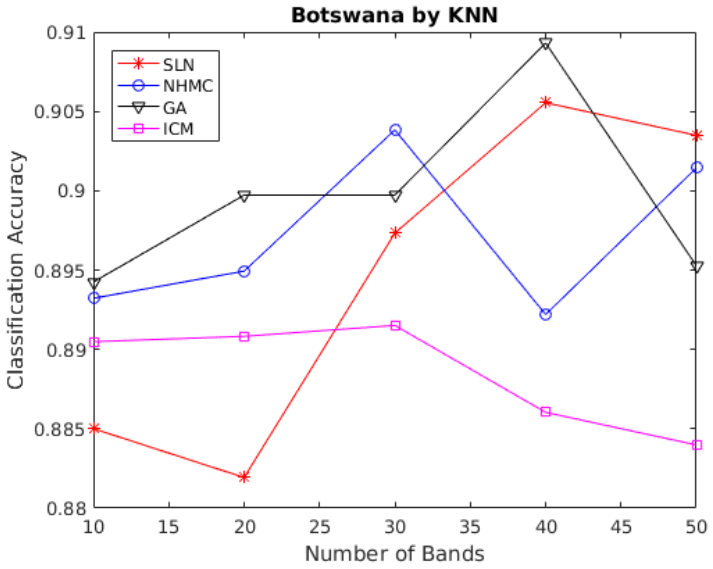


Figure 4.9: KNN classification results using the Botswana dataset.

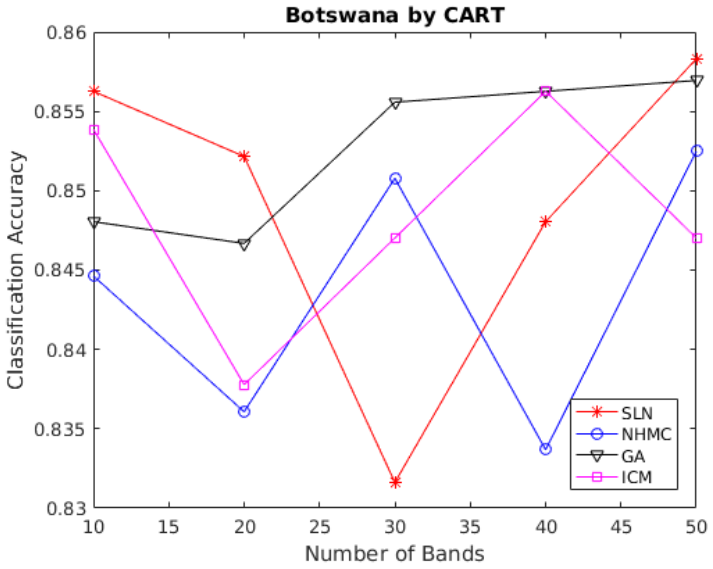


Figure 4.10: The Botswana image under the CART classification.

In Table 4.6, the CART classifier results are exhibited. The same results are depicted in Figure 4.12. The proposed method got the best results again with 30, 40 and 50 bands.

Indian Pines The bands selected by our approach are shown in Table 4.7.

According to Table 4.8, the proposed method got the best result with 30 bands, using the KNN classifier with Indian Pines image. In Figure 4.13, the results are displayed.

In Table 4.9, the proposed `SLN` method has the best result with 20 and 30 bands. The results can be visualized in Figure 4.14.

Table 4.4: Selected bands for KSC image.

10 bands	1 17 34 37 48 74 98 134 161 175
20 bands	1 6 8 19 25 28 33 48 53 72 76 95 133 139 143 150 163 168 173 176
30 bands	1 2 9 15 19 28 31 34 37 41 44 47 48 72 74 75 96 97 101 110 124 133 135 139 143 147 160 163 167 175
40 bands	1 3 7 16 19 26 28 30 32 34 35 39 40 43 49 51 53 56 71 73 77 93 94 95 96 101 104 125 133 134 142 145 150 153 159 162 167 169 171 174
50 bands	1 3 7 9 18 19 26 28 33 34 37 38 39 41 42 47 48 51 53 59 68 69 70 71 72 73 78 96 97 100 101 107 110 111 120 121 125 127 131 133 135 137 140 142 143 159 162 171 173 175

Table 4.5: KNN results for KSC image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
SLN	92.80%	0.23%	93.83%	0.68%	**94.88%**	0.45%	**95.02%**	0.36%	**94.62%**	0.50%
NHMC	92.39%	0.81%	93.51%	0.59%	93.25%	1.04%	94.02%	0.14%	93.54%	0.63%
GA	92.64%	0.18%	94.05%	0.27%	93.70%	0.23%	93.92%	0.09%	94.18%	0.09%
ICM	**93.09%**	0.63%	**94.69%**	0.36%	93.89%	0.59%	94.72%	0.14%	94.43%	0.27%

Table 4.6: CART results for KSC image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
SLN	85.64%	0.14%	84.96%	1.27%	**87.88%**	0.41%	**88.11%**	1.18%	**87.94%**	1.49%
NHMC	85.25%	0.86%	85.22%	0.90%	85.00%	2.31%	85.73%	0.45%	85.96%	0.50%
GA	**86.18%**	0.36%	87.08%	2.08%	86.02%	0.68%	87.27%	0.54%	86.76%	0.72%
ICM	85.96%	1.58%	**87.91%**	0.18%	87.30%	0.41%	88.04%	0.72%	87.40%	0.63%

4.3.4 Different training data sizes

As stated in Section 4.3.3.1, all the results shown in Section 4.3.3.2 are achieved by using only 20% of the available training data.

One question that naturally arises is

What happens if we use more than 20% *of the available training data?*

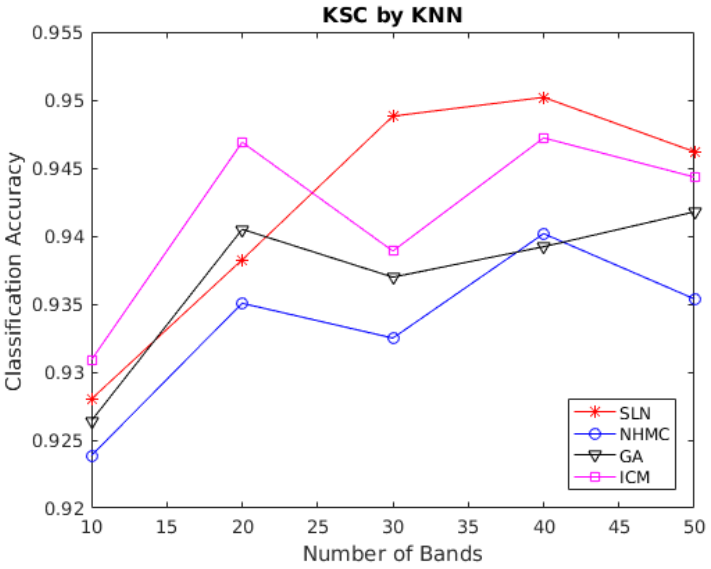


Figure 4.11: The KSC image classified by KNN.

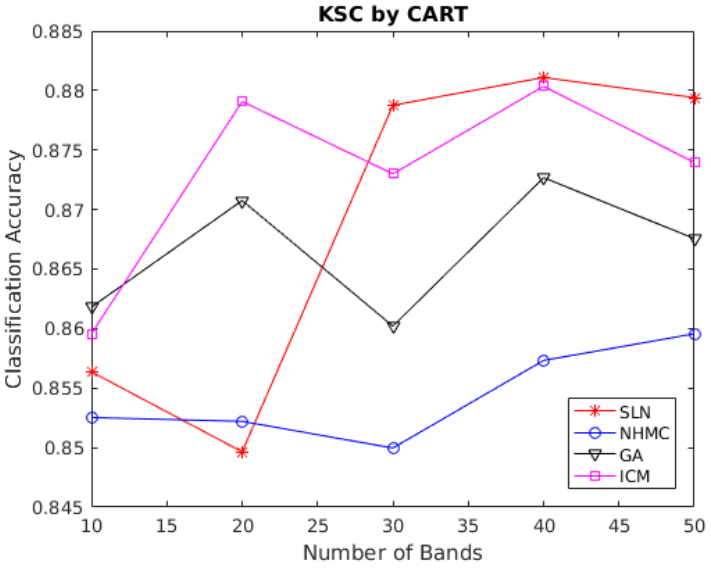


Figure 4.12: The KSC dataset classified by CART.

In order to answer this question, it is necessary to run the Algorithm 1 for each different amount of training data $\mathbf{X}$. We compare the results using from 30% to 100% of the available training data.

Botswana: For the Botswana dataset, the mean classification accuracies achieved by KNN and CART classifiers are shown in Figure 4.15. It is possible to see that using more training data does not necessarily increase the classification accuracy.

In Figure 4.16, the mean processing time for band selection using from 20% until 100% of the available training data is shown. For each percentage of training data, we measure

Table 4.7: Selected bands for Indian Pines image.

10 bands	10 25 32 39 42 48 63 75 91 98
20 bands	5 11 18 23 25 29 36 44 52 56 60 64 75 91 94 98 106 117 132 168
30 bands	5 6 11 16 18 20 23 25 29 30 31 36 38 44 47 52 54 56 60 62 64 74 75 91 94 98 106 117 132 168
40 bands	1 8 10 12 15 19 20 23 27 30 32 36 39 41 49 51 54 56 58 60 64 66 71 74 75 80 91 94 98 101 113 117 149 156 167 170 173 178 206 218
50 bands	4 8 11 14 18 20 22 27 30 32 36 38 41 44 46 48 52 58 60 62 64 66 68 71 75 77 81 84 86 88 90 94 95 98 100 109 117 124 137 140 149 153 156 167 170 172 174 178 202 216

Table 4.8: KNN results for Indian Pines image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
SLN	72.49%	2.37%	72.20%	2.16%	**72.81%**	2.60%	73.07%	0.78%	74.50%	0.67%
NHMC	75.27%	0.85%	76.98%	0.51%	71.66%	0.71%	**73.37%**	0.97%	**74.89%**	0.09%
GA	69.22%	1.03%	65.12%	0.02%	66.16%	0.30%	67.25%	0.14%	67.32%	0.23%
ICM	**78.70%**	0.09%	**78.54%**	0.87%	68.10%	0.64%	68.44%	0.53%	67.32%	1.38%

Table 4.9: CART results for Indian Pines image.

	10 bands		20 bands		30 bands		40 bands		50 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std
SLN	69.69%	1.63%	**73.27%**	0.90%	**74.50%**	1.77%	72.16%	0.11%	74.05%	1.45%
NHMC	70.73%	0.60%	70.47%	0.23%	72.11%	0.99%	73.28%	0.76%	**74.88%**	0.34%
GA	68.18%	1.22%	71.35%	0.32%	74.21%	1.43%	**74.37%**	0.14%	73.56%	0.60%
ICM	**74.72%**	0.80%	73.17%	0.64%	73.61%	0.48%	73.19%	0.11%	74.31%	0.41%

the mean values taking into account the processing time for 10, 20, 30, 40 and 50 selected bands. The mean processing time for each quantity of selected bands is exhibited in Figure 4.17.

KSC: As for the KSC image, in Figure 4.18 we see the mean accuracies for each percentage of available training data, from 20% until 100%, using the KNN and CART classifiers. Again, increasing the amount of training data does not necessarily increases the accuracies.

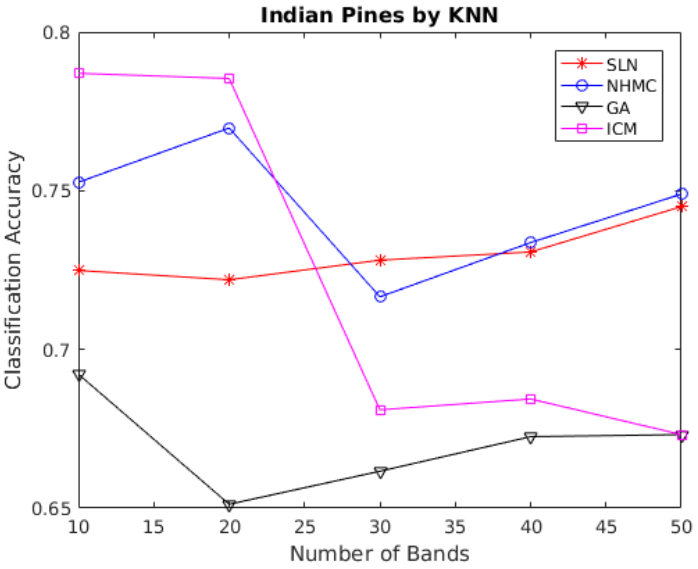


Figure 4.13: Indian Pines classification using KNN.

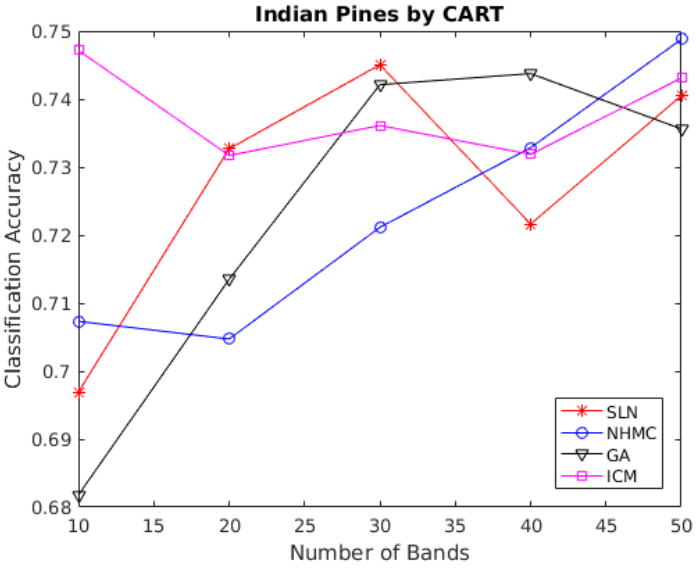


Figure 4.14: Indian Pines classification using CART.

In Figure 4.19, the mean processing time for each amount of available training data is shown. The mean values are measured by taking into consideration the processing time for 10, 20, 30, 40 and 50 selected bands. Figure 4.20 shows the mean processing time for each number of selected bands, using the KSC dataset.

Indin Pines: Considering the Indian Pines dataset, the mean classification accuracies achieved by KNN and CART classifiers are shown in Figure 4.21. Using more training data does not necessarily increase the classification accuracy.

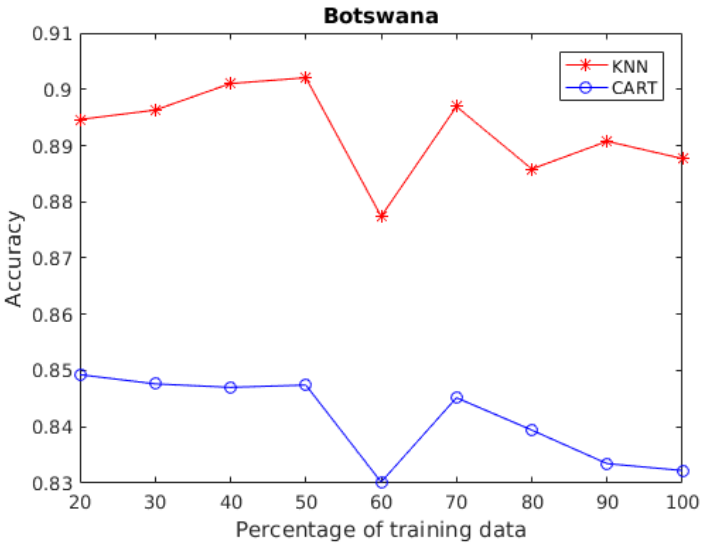


Figure 4.15: Mean accuracies by KNN and CART classifiers using from 20% to 100% of the available training data, for Botswana image.

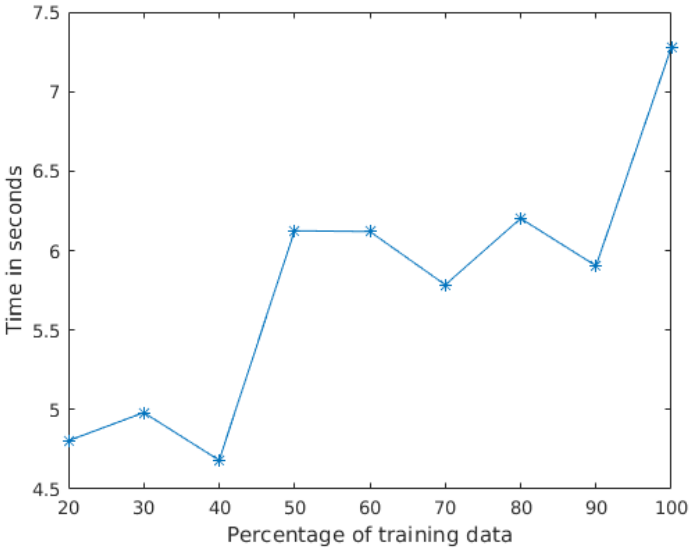


Figure 4.16: Mean processing time for band selection considering 10, 20, 30, 40 and 50 bands, for the Botswana image.

In Figure 4.22, the mean processing time for band selection varying the available training data from 20% until 100% is shown. For each percentage, one measures the mean values taking into consideration the processing time for 10, 20, 30, 40 and 50 selected bands. The mean processing time for each quantity of selected bands is shown in Figure 4.23.

4.3.5 J index

As described in Section 2.3.2.1, the J index—see Equation 2.10—, which is based on Scatter Matrices, can measure how well the the classes are separated from each other in

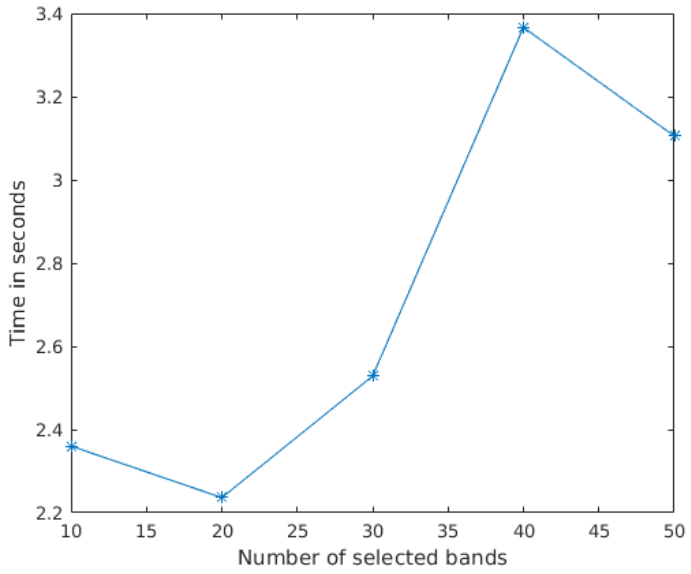


Figure 4.17: Processing time for each number of selected bands, using the Botswana dataset.

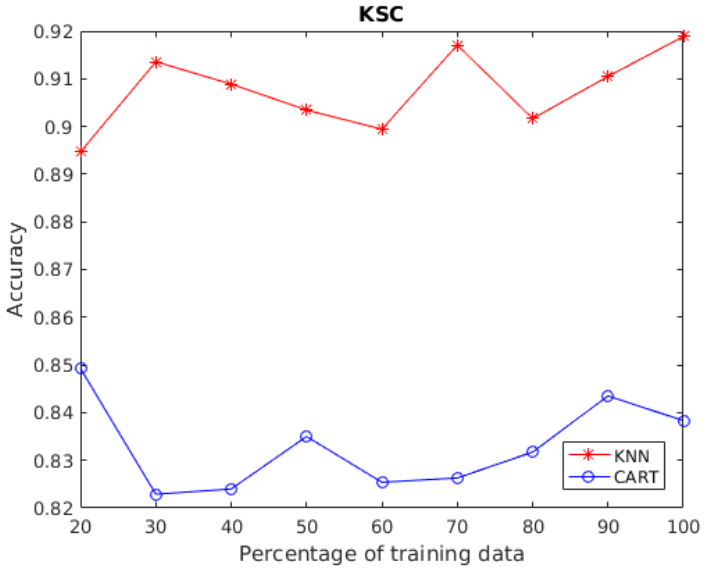


Figure 4.18: Mean accuracies by KNN and CART classifiers using from 20% to 100% of the available training data, for KSC image.

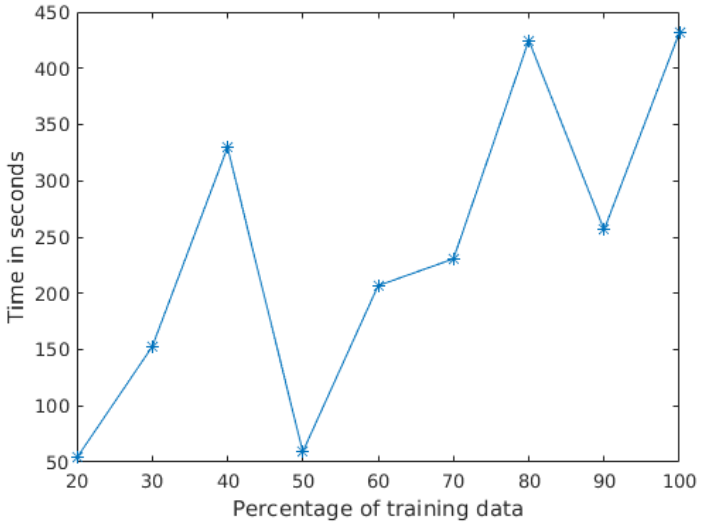


Figure 4.19: Mean processing time for band selection considering 10, 20, 30, 40 and 50 bands, for the KSC image.

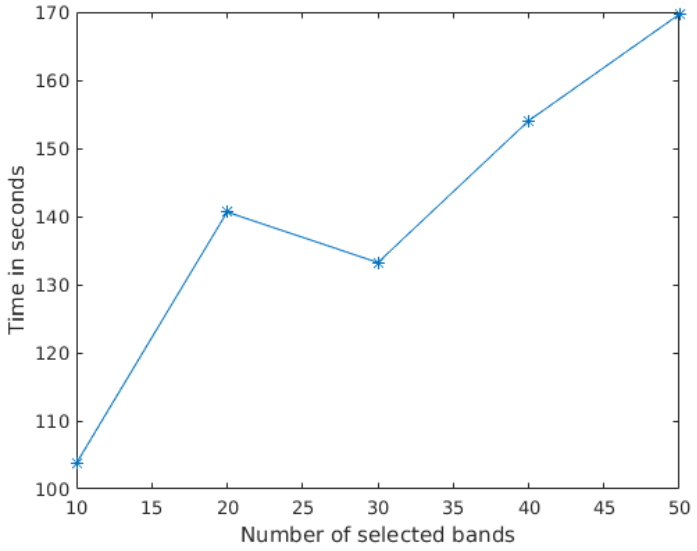


Figure 4.20: Processing time for each number of selected bands, using the KSC dataset

the feature space.

For the calculation of J, it is necessary to have the class information of the data samples. Thus, this analysis is only possible in supervised problems.

In Table 4.10, the J indices of all BS methods are shown, taking as input the Botswana dataset. We see that the proposed method `SLN` has the best result with 50 bands. In Figure **??**, it is possible to see the J results.

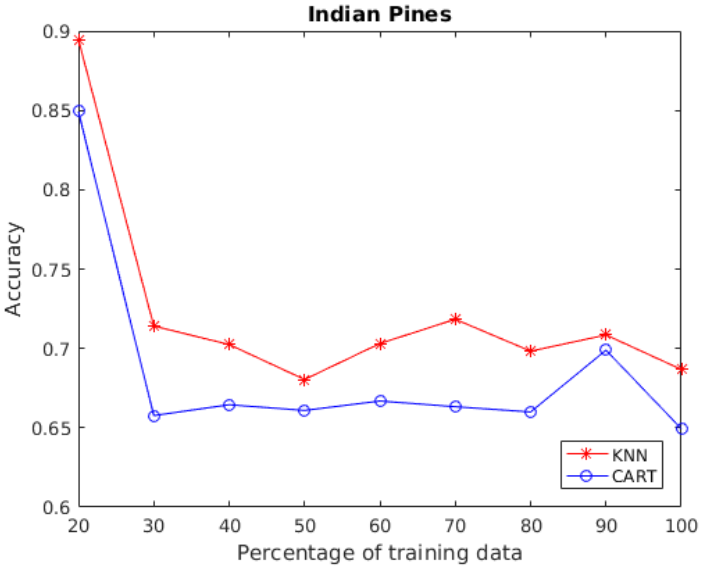


Figure 4.21: Mean accuracies by KNN and CART classifiers using from 20% to 100% of the available training data, for Indian Pines image.

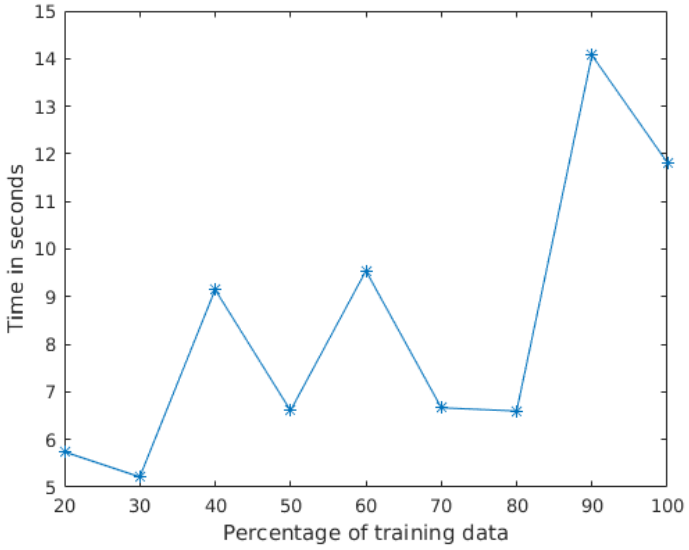


Figure 4.22: Mean processing time for band selection considering 10, 20, 30, 40 and 50 bands, for the Indian Pines image.

Table 4.10: J indices for the Botswana image.

Method	10 bands	20 bands	30 bands	40 bands	50 bands
SLN	182.0879	181.1352	184.0382	183.5925	**190.3920**
NHMC	168.4621	192.3552	**198.0085**	190.6596	189.5664
GA	**250.9020**	**243.7681**	109.7085	**199.6179**	109.7085
ICM	147.7290	150.1814	153.9822	154.1625	155.6518

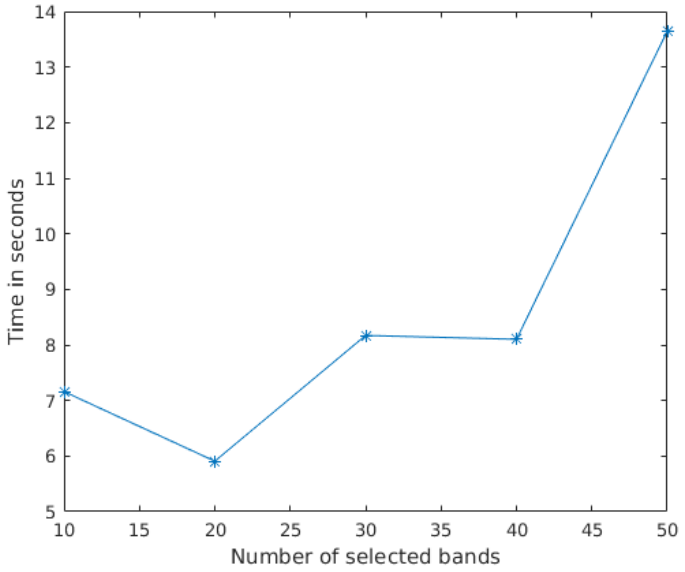


Figure 4.23: Processing time for each number of selected bands, using the Indian Pines dataset

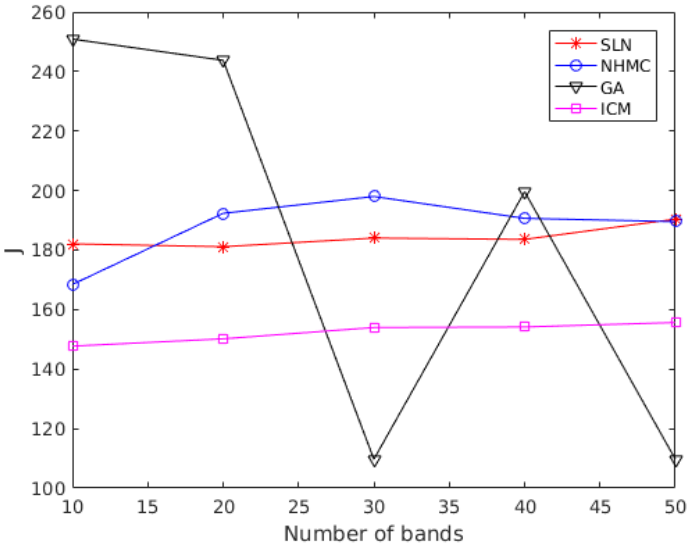


Figure 4.24: J indices for the Botswana image.

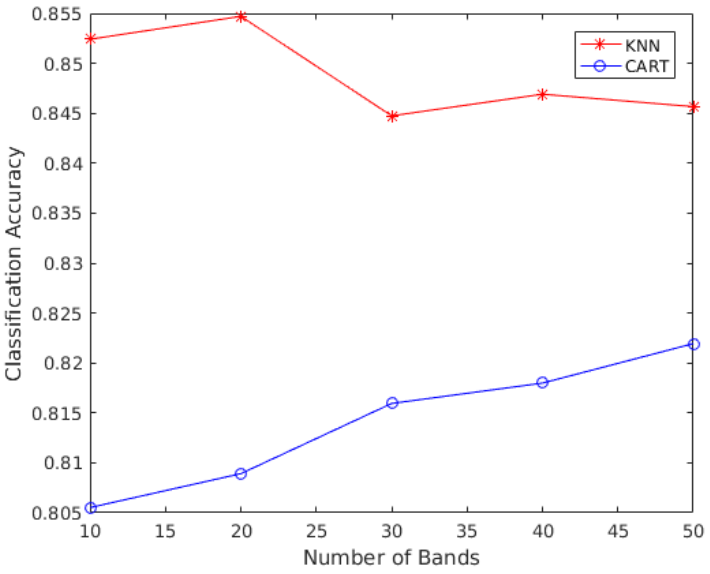


Figure 4.25: Overall results considering all methods compared in this chapter, using all the three images.

4.3.6 Remarks about the results

4.3.6.1 KNN versus CART

Both classifiers used in this chapter are nonparametric, that is, they do not assume any hypothesis about data distribution nor about its parameters. Yet, they share more dissimilarities than characteristics in common. To illustrate this, one can notice the differences between the overall accuracies of the two classifiers, taking into account all the methods compared in this chapter using the three hyperspectral images: For KNN, the mean accuracies of all results are 84.89%, whereas CART has 81.41% of mean accuracy. One possible explanation may be related to the highly nonlinearity of the classes boundaries. For example, let h_l be a homogeneous region of the feature space defined by CART, whose a new entry $\mathbf{x}_i$ will be classified as c_l, even if it belongs to class c_j. This, obviously, is a classification error. KNN classifier, in this situation, would inquire the k nearest neighbors of $\mathbf{x}_i$, and eventually assign the c_j label to it.

The overall results for each number of selected bands can be seen in Figure 4.25. For the CART classifier, the classification accuracy increased as the number of bands increased. For the KNN classifier, there was an opposite effect. This indicates that CART is not susceptible to the curse of dimensionality, at least in the dimensions and with the images analyzed. In general, however, one can see that there is a general improvement in results as the number bands increases from 10 to 50, as shown in Figure 4.27.

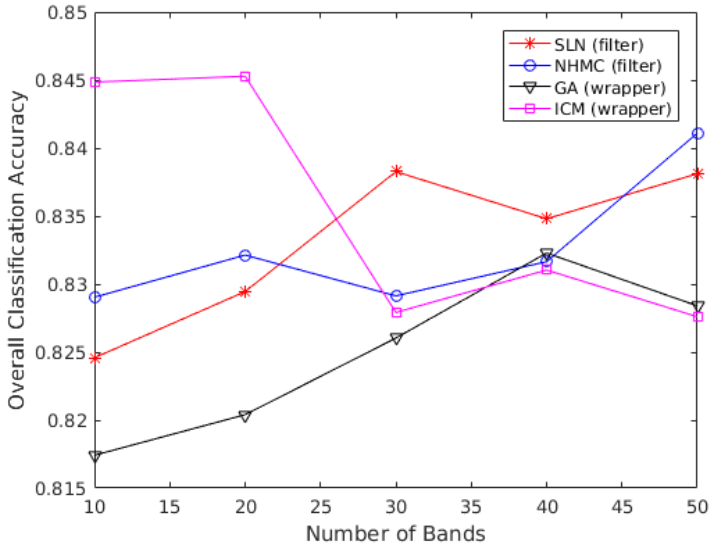


Figure 4.26: Mean results of each method, using all images and both classifiers.

4.3.6.2 Filter versus Wrapper

It is frequently stated in the literature that wrapper-based methods are superior in performance to filter approaches (Cao et al., 2017b; Theodoridis and Koutroumbas, 2008; Shahana and Preeja, 2016; Molina et al., 2002; Cao et al., 2016b; Ma et al., 2017). Concerning the methods compared in this chapter, `SLN` and `NHMC` are filter-based approaches; and `GA` and `ICM` are wrapper frameworks, using, respectively, Gaussian Maximum Likelihood and SVM classifiers in their frameworks.

In Figure 4.26, the mean results of all methods are displayed. It uses all the three images with both classifiers. In sum, Figure 4.26 shows the mean values of the Tables 4.2, 4.3, 4.5, 4.6, 4.8 and 4.9. It is evident that the wrapper methods are not necessarily better than filter approaches. More precisely, the wrapper methods yield better results in only two situations—with 10 and 20 bands—, and in the remaining cases filter methods have a superior performance.

It is worth noting that a wrapper-based method proceeds to the band selection by using a certain classifier, and this classifier is supposed to be used during the subsequent classification process. It was not the case here. That is, the two wrapper competitors selected bands using one classifier they are and used in this thesis with another one, and this fact may explain why those two methods could not outperform the filter-based frameworks. On the other hand, filter methods perform the BS task without any relation with the classifier, which makes them more versatile compared to wrapper approaches.

4.3.6.3 Methods comparison

In general, as we can see in Figures 4.9, 4.10, 4.11, 4.12, 4.13 and 4.14 all the four methods had their best and worst results in different situations. Thus, pointing out the best framework would not be an easy task.

In Figure 4.26, we can see the mean results of the four methods. The proposed method, `SLN`, has the best mean results using 30 and 40 spectral bands. If we take the mean value of all results—using the three images, both classifiers and also all bands—, the results are thus:

1. `ICM`: 83.53%;
2. `SLN`: 83.30%;
3. `NHMC`: 83.26%; and
4. `GA`: 82.49%.

Basically, `ICM` framework has the best overall result because it gets very high accuracies with less spectral bands. However, it does not achieve good results with more bands. Thus, `ICM` has a instable behavior as the number of bands change.

For the sake of a fair comparison, we will count how many times each method yields the best results—values in bold in Tables 4.2, 4.3, 4.5, 4.6, 4.8 and 4.9. In this case, we have the following outcome:

1. `SLN`: 13;
2. `GA`: 7;
3. `ICM`: 6; and
4. `NHMC`: 4.

Consequently, one can infer that our proposed method has a stable good outcome, achieving the best results in 43.33% of the tests.

4.3.6.4 Remarks about the proposed method

As shown in Figure 4.26, the proposed method has a tendency of achieving better results as the number of selected bands increases. In fact, this is an expected behavior. In order to see that, refer to Figure 4.27, which shows the overall results of all methods together,

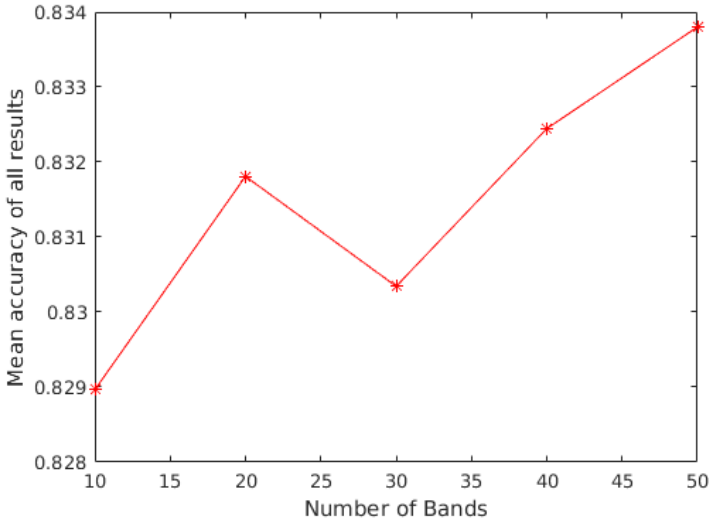


Figure 4.27: Mean accuracies of all results by both classifiers in relation to the number of selected bands. All the three images are used.

using the three images and both classifiers. In general, the accuracies increase if more bands are employed.

Concerning the J index, our method has a stable behavior as the number of bands increases, reaching the best mark with 50 bands. This stability shows the reliability of the proposed method, due to the fact that one can select more bands without compromising the separability amongst classes. For example, the `GA` method has the second place in terms of best results. However, as shown in Figure 4.24, its J indices vary abruptly as the number of bands increases, indicating a weak reliability. More precisely, as the number of bands changed from 20 to 30, the separability of classes sharply decreased, meaning that not so good bands were chosen.

As already said in Section 4.2.2.4, there is not a minimum limit of bands to be selected by `SLN`. However, Figure 4.26 shows that `SLN` achieves below-average results with 10 and 20 bands. When it comes to the 10-bands case, we may conclude that its poor results—compared to its competitors—may be due to the fact that `SLN` does not inquiry all the classes of band selection when $\gamma < q$.

Using 30, 40 and 50 bands, the proposed method gets above-average results, as shown in Figure 4.26. In those cases, all the classes were inquired during the BS process. Consequently, better results were achieved.

Finally, we should bear in mind that all the proposed method's results were attained by using only 20% of the available training data. Besides, the simplicity of the proposed method, in terms of implementation, makes it a good choice in the feature selection area.

4.3.6.5 What happens when more training data are used?

As shown in Figures 4.15 and 4.18, concerning the Botswana and KSC datasets, respectively, there is not much difference in accuracies when the amount of available training data is increased. This indicates that the proposed BS framework is really capable of doing its job with few data. In other words, if the classes are well represented even with few data samples, the proposed BS method can select good discriminating bands. In Figure 4.21, there is a positive peak with 20% of training data, which is somewhat unexpected. One possible explanation is that the single-layer neural net used with 20% of training data might have had a very good weights and bias initialization, permitting thus the framework to select very good spectral bands.

When more data are used, the training process takes generally longer, as shown in Figures 4.16, 4.19 and 4.22. Some disturbances in those curves may happen due to the random initialization of the neural network used during the band selection process.

As shown in Figures 4.17, 4.20 and 4.23, there is an increase in the processing time as more bands are selected. At first, it seems strange, because after the step 4 in Algorithm 1, $round(\gamma/q)$ bands are identified and selected, and, in computational terms, it does not matter whether 4 or 10 bands, for instance, are selected. It happens, however, that the bands selected at iteration r are removed from the feature space, consequently at iteration $r+1$ the BS framework does not count on them to find the separating hyperplane. Thus, if discriminating bands are removed from the feature space, it becomes harder—and longer—for the single-layer net to converge to a good solution in subsequent iterations. Furthermore, the more bands are selected, that is, the bigger the $round(\gamma/q)$, the more evident this issue becomes.

4.3.6.6 Visual inspection of the selected bands

In Figure 4.8, the mean values of the spectral signatures for each class are shown, considering the three datasets analyzed. In the regions of the spectrum where the signatures are more separated it is easier to find separating boundaries between the classes. Also, the bands that lie in those regions are the most discriminating ones, and, thus, they should be selected.

Figure 4.28 shows the mean values of the classes spectral signature in the Botswana dataset. The red vertical lines indicate the positions of the 10 bands selected by the proposed method, as shown Table 4.1. It is possible to notice that the selected bands are located in the regions where the spectral signatures are more separated. Consequently, one may infer that the proposed BS method is capable of selecting the bands that provide a good separability amongst the classes.

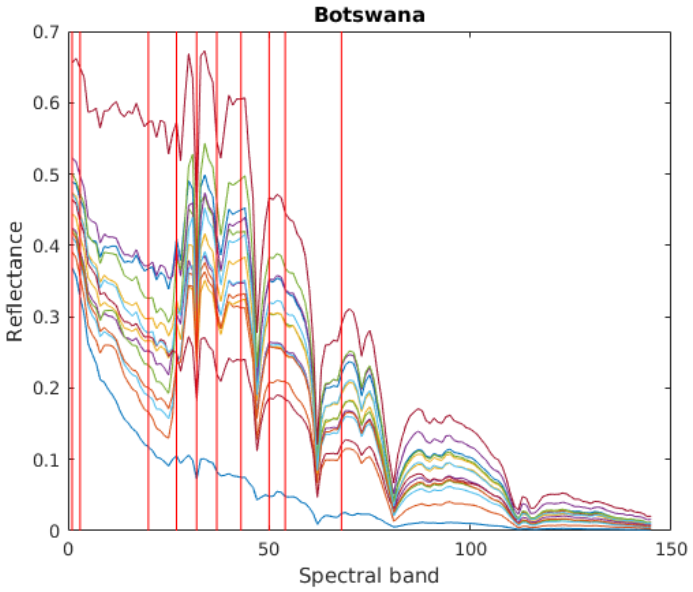


Figure 4.28: Mean spectral signature values of the Botswana image classes. The vertical lines indicate the location of the 10 bands selected by the proposed method.

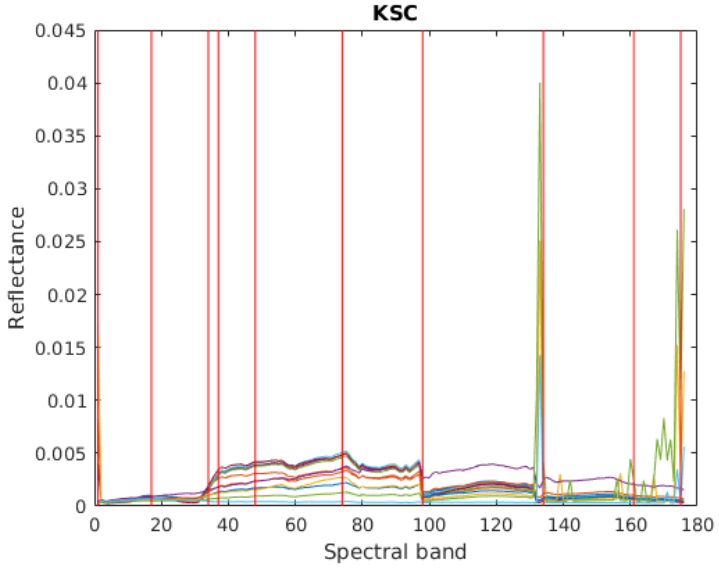


Figure 4.29: Mean spectral signature values of the KSC image classes. The vertical lines indicate the location of the 10 bands selected by the proposed method

In Figure 4.29, the mean values of the spectral signatures KSC's classes are exhibited. In general, except for the band 17, the vertical lines show the positions where the spectral signatures are more separated. The bands are enlisted in Table 4.4. Consequently, we see that the proposed framework can select good bands.

As for the Indian Pines dataset, in Figure 6.6 we see the spectral signatures mean values of the classes. The vertical lines, which indicate the location of the 10 bands selected by

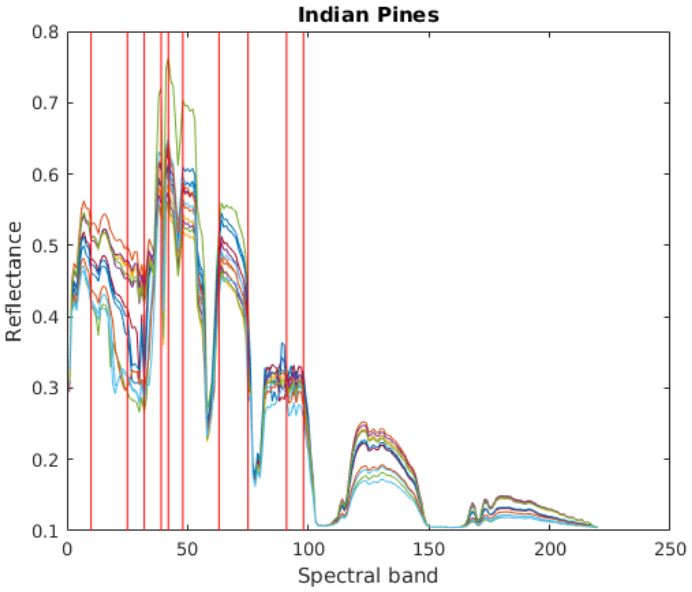


Figure 4.30: Mean spectral signature values of the Indian Pines image classes. The vertical lines indicate the location of the 10 bands selected by the proposed method.

our method —see Table 4.7—, are located in the spectrum regions where the signatures are more separated, which indicates the validity of the proposed method.

Chapter 5

Unsupervised Clustering-based Band Selection using Single-Layer Neural Network

An expected and useful improvement of a *supervised* band selection framework is the capacity to tackle unlabeled data. In other words, an evolution in its operating power marches towards the *unsupervised* approach.

Naturally, in terms of Pattern Recognition, the unsupervised issue poses a more challenging problem, because, without the class information, a plausible solution is to resort to clustering (Duda et al., 2001), which is oftentimes a difficult task due to the fact that one is supposed to define an appropriate measure for similarity between two feature vectors. Besides, the algorithmic scheme that will cluster the vectors must be carefully chosen (Theodoridis and Koutroumbas, 2008). As for the band selection approaches, the lack of class information drives the BS methods to take into account the structure of the dataset. That is, the relevant bands are those that maintain the original dataset structure, while keeping low correlation amongst them.

Thus, in this chapter we propose a BS framework which is an unsupervised version of the method proposed in Chapter 4. In sum, at each iteration, we cluster the data samples in two groups, and then a single-layer neural network finds a separating hyperplane between the two clusters. The bands related to the biggest and smallest hyperplane parameters are selected, and this process iterates until the desired number of bands is achieved. The most important difference in relation to the method proposed in Chapter 4 is the clustering step, which gives the framework its unsupervised characteristic.

Our method belongs to this last group. It uses bisecting k-Means to generate clusters according to intrinsic structure of the data set in the feature space, taking into account only the spectral information. Between each pair of clusters, a single-layer neural

network is used. Thus, we propose a very easily implementable framework for band selection.

5.1 Proposed Framework

5.1.1 Definitions

Let $\mathbf{C}^{(0)}$ be the whole data set corresponding to a hyperspectral image, whose elements are vectors $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ that contain spectral signatures, where d is the number of bands.

Finally, let $\mathbf{C}_g^{(l)}$ be a cluster of $\mathbf{C}^{(0)}$, where l is the partition level, and $\mathbf{C}_1^{(l)} \cup \mathbf{C}_2^{(l)} \cup \cdots \cup \mathbf{C}_g^{(l)} = \mathbf{C}^{(0)}$, and $\mathbf{C}_p^{(l)} \cap \mathbf{C}_q^{(l)} = \emptyset, \forall p \neq q$. There may be several levels, that is, $l = 1, 2, 3, \ldots$, and for each level the number of partitions g is given by $g = 2^l$.

5.1.2 Description

5.1.2.1 General view

The proposed framework begins with an empty subset of selected bands, that is, $\mathbf{S} = \emptyset$, to which the bands selected from $\mathbf{A}$ will be added. At the first iteration, $\mathbf{C}^{(0)}$ is split by the K-Means algorithm into two partitions, $\mathbf{C}_1^{(1)}$ and $\mathbf{C}_2^{(1)}$, following a bisecting K-Means approach (Banerjee et al., 2015). Since we consider that all the features have the same variance —to simplify the problem—, all the samples fall in equal-size hyperspherical clusters. Thus, the resulting discriminant function for a two-class case will be linear (Duda et al., 2001). Consequently, the assignment of an input vector to a class will be given by the Euclidean distance, which is used by K-Means.

Then, we use a single-layer neural network to find a hyperplane that separates those two partitions. After that, two bands are selected, and consequently discarded from the feature space $\mathbf{F}$. If more bands are needed, we keep repeating this procedure in deeper levels.

As the proposed method is based on both clustering and single-layer neural networks, we shall call it `CSLN`. Its characteristics are described below.

5.1.2.2 Iterations

`CSLN` is an iterative band selection method. At each iteration, a binary classification problem between k-Means-generated partitions $\mathbf{C}_p^{(l)}$ and $\mathbf{C}_q^{(l)}$ is to be solved by the function f. Since two bands are selected at each iteration, one needs to repeat the

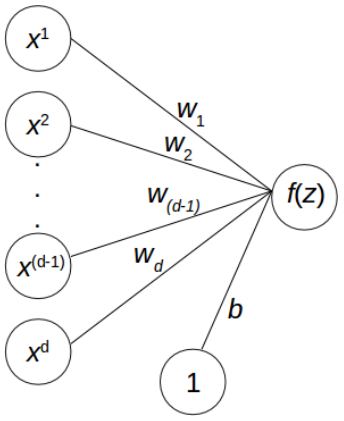


Figure 5.1: An example of the single-layer neural network used in this thesis. This architecture permits that each band $\mathbf{x}^i$ be linked to only one weight w_i.

process until the desired number of bands γ is attained. The following criteria are adopted:

- If γ is even, one needs $\gamma/2$ iterations; and
- If γ is an odd number, $(\gamma + 1)/2$ iterations are necessary, and the first γ selected bands are kept.

At each iteration, the neural net f is trained from scratch due to two reasons:

- In order to avoid possible local minimum regions from previous clusters; and
- After each iteration the feature space $\mathbf{F}$ decreases. Consequently, the sizes of $\mathbf{w}$ and b also change. So, the whole architecture must be trained again.

5.1.2.3 Selection of bands

After the training of the neural network, it is possible to give degrees of importance to all $a \in \mathbf{F}$. As every element $x^l \in \mathbf{x}$ is directly linked to w_l—see Figure 5.1—, for $l = 1, ..., p$, the magnitude of w_l is a indicator for the band a_l. As already seen in Equation 4.4, the largest and the smallest weights constitute the most important contributions to the sign of z. Thus, the bands linked to those weights are also considered the most important, and, consequently they are added to the set $\mathbf{S}$. The feature space $\mathbf{F}$ is then updated by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$.

5.1.2.4 Avoiding highly correlated bands

The bands of a hyperspectral image are contiguous, which causes a high correlation among neighboring bands.

Bearing this in mind, we adopt a method that avoids the selection of highly correlated bands. For each band $a_k \in \mathbf{F}$ we construct a vector $\mathbf{v}_k$, whose elements are the bands indices in a descending order in relation to the correlation to the band a_k. That is, $\mathbf{v}_k(1)$ is the index of the band $a_{\mathbf{v}_k(1)}$, which is the band with the highest correlation with a_k. The correlation ρ between two bands a_α and a_β is given by

$$\rho_{(a_\alpha, a_\beta)} = \frac{cov(a_\alpha, a_\beta)}{\sigma_{a_\alpha}\sigma_{a_\beta}},$$

where $cov()$ is the covariance and σ is the standard-deviation.

Thus, the following procedure is adopted:

- At a certain iteration, a band a_k is selected, so $\mathbf{S} \leftarrow a_k$;
- $\mathbf{G} \leftarrow a_{\mathbf{v}_k(1)}$, and $\mathbf{G}$ is, at the beginning, an empty set;
- After this iteration, the feature space $\mathbf{F}$ is updated by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$.

We emphasize that only $a_k \in \mathbf{S}$ are the selected bands. The bands $a_{\mathbf{v}_k(1)} \in \mathbf{G}$ are discarded.

Algorithm 2 gives the steps followed by the proposed `CSLN` framework.

Algorithm 2 Proposed band selection framework

1: Input : $\mathbf{C}^{(0)}$, $\mathbf{A}$, γ, $\mathbf{S} = \emptyset$ and $\mathbf{G} = \emptyset$
2: **for** $r = 1 :$ maxIterations **do**
3: Train a single-layer neural network f to find a hyperplane that separates $\mathbf{C}_p^{(l)}$ and $\mathbf{C}_q^{(l)}$ clustered by K-Means
4: Select the bands $a_k \in \mathbf{F}$ related to the largest and smallest $w \in \mathbf{w}$
5: $\mathbf{S} \leftarrow a_k$, and $\mathbf{G} \leftarrow a_{\mathbf{v}_k(1)}$
6: Update the feature space $\mathbf{F}$ by $\mathbf{A} \setminus (\mathbf{S} \cup \mathbf{G})$
7: **end for**
8: Return: $\mathbf{S}$

Fig. 5.2 depicts the proposed framework. Initially, the whole data set is split into two clusters by K-Means. Then, a single-layer neural network f is used to find a hyperplane that separates the clusters. This process is repeated until the desired number of bands is selected.

5.2 Results

The results of the proposed method are exhibited in this Section. They are compared with other band selection approaches by considering the accuracy of supervised classifiers.

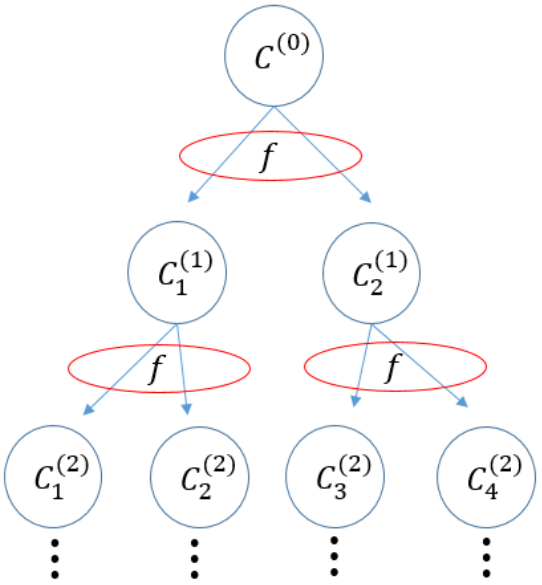


Figure 5.2: A general view of the proposed BS framework. At each binary clustering, a single-layer neural net f is used to select the bands.

5.2.1 Datasets and classifiers

In this chapter, we use two hyperspectral images, which are Indian Pines and Pavia University, already described in Section 2.5.

As classifiers, for the the Indian Pines dataset we use Classification and Regression Trees and k-Nearest Neighbors are used. For the Pavia University image, we use KNN, CART and Support Vector Machine (Theodoridis and Koutroumbas, 2008).

The classifiers are run in MATLAB. For KNN, we use the `fitcknn` command, $k = 3$ in all experiments. For CART, `fitctree` is used, and to run SVM for multiple classes, we use the command `fitcecoc`, with polynomial kernel. All those commands belong to the Statistics and Machine Learning toolbox.

The proposed band selection framework is also implemented in MATLAB, and we use the `trainSoftmaxLayer` command, from Neural Network Toolbox, for single-layer neural network, with 2000 training epochs—normally the training phase stopped before this, so other training epochs quantities were not tested. For the k-Means algorithm, we use the `kmeans` command, from the Statistics and Machine Learning toolbox.

The input data are the images with the selected bands. Furthermore, the number of bands selected by our method are for comparison purposes. Therefore, it does not mean they are the optimum number for any given application.

For the classification process, each dataset is divided into two subsets. The first subset is used during the training phase of the classifier, with 70% of the total data. The remaining

30% are used during the test phase of the classifier, yielding the results shown in Section 5.2.4.

5.2.2 Competitors

The performance of the proposed method is compared with six other BS approaches.

Four of them are used with the Indian Pines image:

- This method is also clustering-based (Martinez-Uso et al., 2007), and it will be referred to as `WaLuDi`;
- This approach uses both ranking and clustering for band selection (Datta et al., 2015), and we will call it `CR`;
- This competitor relies on information divergence, and this method will be called `ID` (Chang and Wang, 2006); and
- This framework resorts to band elimination with partitioned image correlation (Datta et al., 2014), and it will be referred to as `EM`.

For the Pavia University image, there are two competitors:

- The authors propose a framework that handles two conflicting objective functions. One function is designed to represent the information contained in the selected bands, by means of entropy (Gong et al., 2016). It will be called `MOBS`;
- In this approach, the authors construct a band channel with the original bands. Then, some bands are selected by Blahut's algorithm, which iteratively finds a feature space that provides the best channel capacity (Chang et al., 2017). This competitor will be referred to as `CC`.

As already stated in Section 5.1.2.1, our proposed method will be referred to as `CSLN`.

5.2.3 Selected bands

The bands selected by the proposed framework are displayed in Table 5.1. For Indian Pines, we have only the first 18 best-ranked bands of our competitors, thus the analyses of results are restricted to this number of bands. For the Pavia University image, we select 21 bands for the same reason.

The bands in Table 5.1 are sorted according to the order they were selected. For example, at the first iteration, the bands 2 and 42 were selected for Indian Pines.

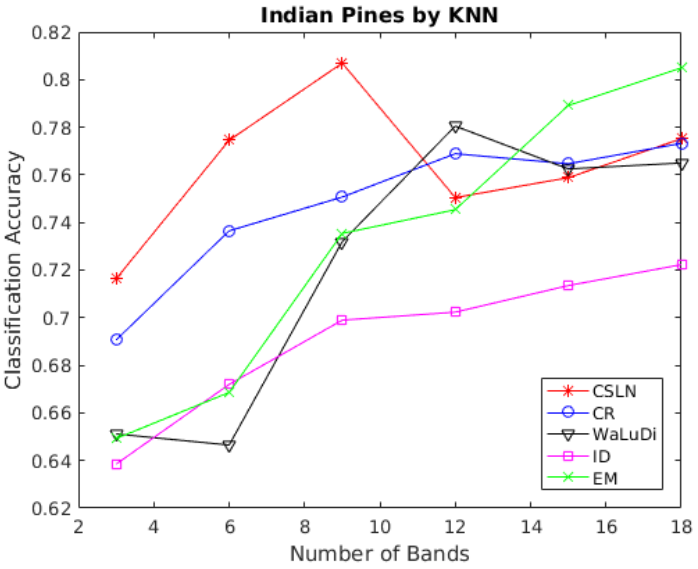


Figure 5.3: The Indian Pines dataset under the KNN classification.

The results comparisons are made with different quantities γ of selected bands, that is, $\gamma_s = s \times 3$, with $s = 1, 2, 3, 4, 5, 6$, for Indian Pines. Thus, for $\gamma_2 = 6$, for example, the first six bands of Table 5.1 are used. For the Pavia University image, $s = 1, 2, 3, 4, 5, 6, 7$.

Table 5.1: The selected bands according to the order of selection by the proposed method.

Selected bands for Indian Pines	2, 42, 6, 39, 22, 58, 25, 62, 71, 101, 94, 151, 111, 203, 156, 183, 171, 215.
Selected bands for Pavia University	69, 1, 68, 3, 92, 6, 77,18, 101, 24, 99, 51, 14, 74, 9, 75, 29, 97, 95, 8, 103.

5.2.4 Results comparison

The classification results exhibited throughout this chapter are the mean values over ten runs.

In Table 5.2, the results for the Indian Pines image are shown. Under KNN classification, the proposed method `CSLN` has the best results using 3, 6 and 9 bands. It is illustrated in Fig. 5.3.

The overall results achieved by the CART classifier using Indian Pines are also exhibited Table 5.2. The proposed framework achieves the best results with 3 and 9 bands. Fig. 5.4 provides a visual perspective of the results.

Table 5.2: Classification results for Indian Pines.

KNN results

	3 bands	6 bands	9 bands	12 bands	15 bands	18 bands
Method	acc.	acc.	acc.	acc.	acc.	acc.
CSLN	**71.63%**	**77.45%**	**80.69%**	75.05%	75.88%	77.52%
WaLuDi	65.12%	64.65%	73.19%	**78.05%**	76.25%	76.50%
CR	69.06%	73.65%	75.07%	76.89%	76.47%	77.32%
EM	64.92%	66.86%	73.54%	74.54%	**78.92%**	**80.50%**
ID	63.85%	67.20%	69.90%	70.23%	71.35%	72.23%

CART results

	3 bands	6 bands	9 bands	12 bands	15 bands	18 bands
Method	acc.	acc.	acc.	acc.	acc.	acc.
CSLN	**53.51%**	64.49%	**68.56%**	68.36%	69.41%	70.85%
WaLuDi	45.62%	53.71%	65.55%	**68.68%**	69.68%	70.96%
CR	52.03%	**65.66%**	66.93%	68.29%	70.46%	72.25%
EM	44.72%	55.72%	66.28%	66.57%	**71.33%**	**73.12%**
ID	49.07%	53.16%	58.85%	62.43%	63.37%	67.06%

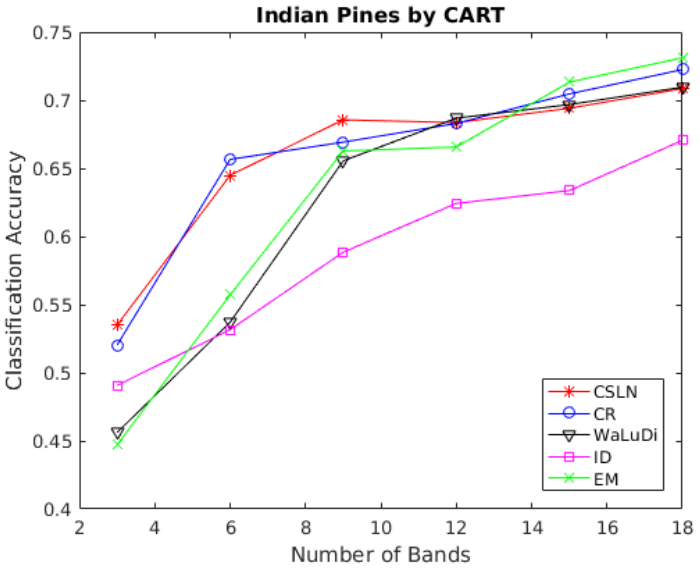


Figure 5.4: The Indian Pines image classified by CART.

Table 5.3: Classification results for Pavia University.

KNN results

	3 bands	6 bands	9 bands	12 bands	15 bands	18 bands	21 bands
Method	acc.	acc.	acc.	acc.	acc.	acc.	acc.
CSLN	**86.66%**	**90.63%**	**91.32%**	**92.11%**	**92.64%**	91.83%	91.82%
MOBS	70.30%	77.26%	87.68%	90.19%	91.15%	**92.73%**	**93.21%**
CC	85.98%	85.98%	86.32%	87.14%	87.91%	90.37%	91.00%

CART results

	3 bands	6 bands	9 bands	12 bands	15 bands	18 bands	21 bands
Method	acc.	acc.	acc.	acc.	acc.	acc.	acc.
CSLN	**72.96%**	**81.91%**	**83.00%**	**84.37%**	84.66%	84.54%	84.45%
MOBS	50.28%	60.13%	77.65%	81.64%	**85.37%**	**88.19%**	**88.07%**
CC	72.87%	73.36%	74.17%	75.18%	76.05%	81.92%	83.77%

SVM results

	3 bands	6 bands	9 bands	12 bands	15 bands	18 bands	21 bands
Method	acc.	acc.	acc.	acc.	acc.	acc.	acc.
CSLN	**78.81%**	**87.17%**	**90.33%**	**93.03%**	**94.36%**	83.76%	92.58%
MOBS	61.30%	71.50%	85.96%	91.65%	91.50%	**98.23%**	**98.88%**
CC	77.99%	79.47%	80.71%	82.33%	83.75%	84.63%	91.49%

As for the Pavia University image, the accuracy results are shown in Table 5.3. Using the KNN classifier, the proposed method has the best results with 3, 6, 9, 12 and 15 bands. This is illustrated in Fig. 5.5.

For CART, our method achieves the best results with 3, 6, 9 and 12 bands. Fig. 5.6 shows it.

Using the SVM classifier, the proposed CSLN method has the best results with 3, 6, 9, 12 and 15 bands, which can be seen in Fig. 5.7.

5.2.4.1 Visual inspection of the selected bands

The spectral signatures of the different classes give us an idea of the features—or bands—that provide a good separation amongst classes. The more the signatures are far from one another, the better it is for the classifier.

Figures 5.8 and 5.9 show the mean spectral signatures of the classes present in Indian Pines and Pavia University datasets, respectively. In order to avoid excessive visual information, the location of only the first 6 selected bands is displayed, in vertical lines.

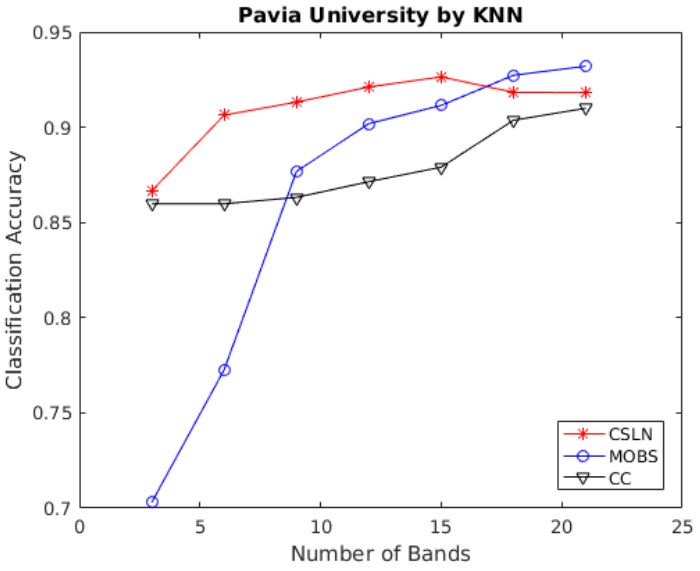


Figure 5.5: The Pavia University dataset classified by KNN.

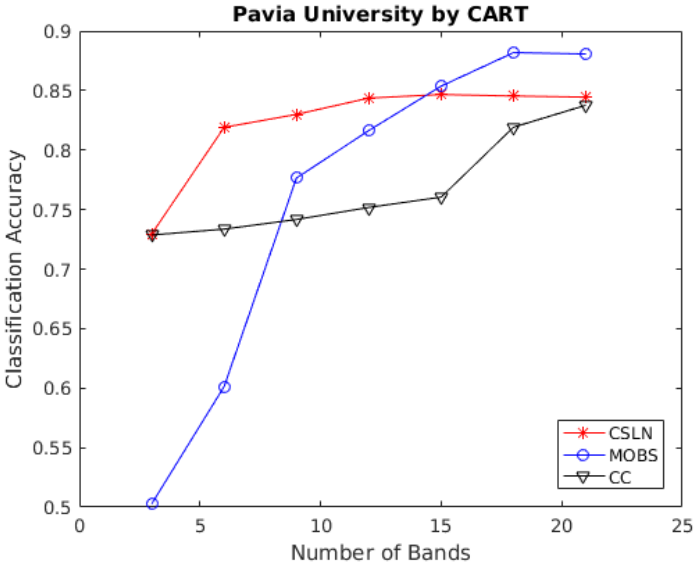


Figure 5.6: The Pavia University image classified by CART.

We notice that in both cases the selected bands fall in regions where the spectral signatures are far from one another. It denotes that our BS framework proposed in this chapter is capable of selecting appropriate spectral bands.

5.2.4.2 Considerations about the single-layer neural net choice

As already stated in Section 5.1.2.3, our rationale for the band selection is based on Equation **??**, which, in turn, is the separating hyperplane calculated by the single-layer neural network used in this chapter.

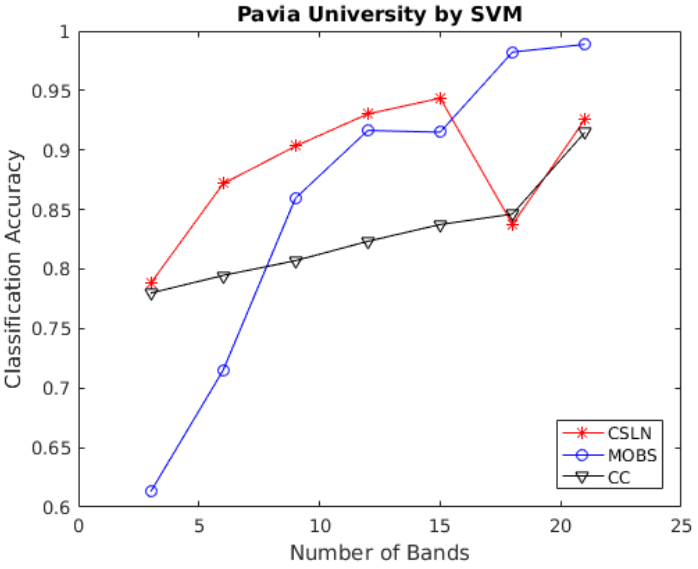


Figure 5.7: The Pavia University dataset under SVM classification.

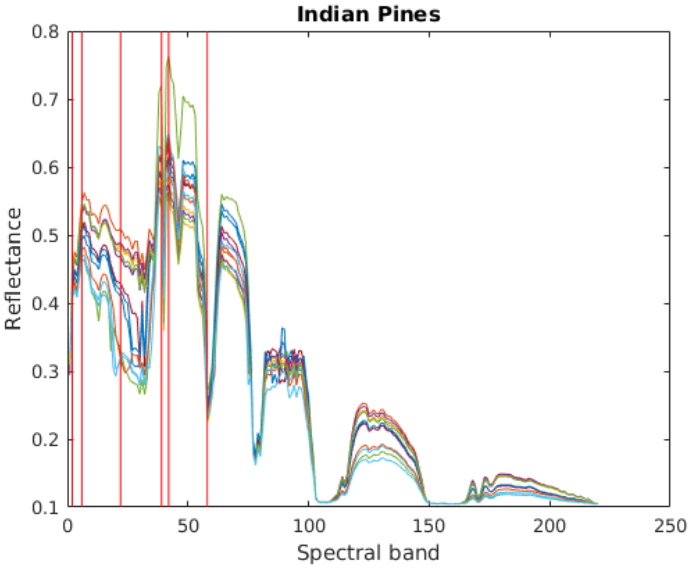


Figure 5.8: Mean spectral signature values of the Indian Pines image classes. The vertical lines indicate the location of the first 6 bands selected by the proposed `CSLN` method.

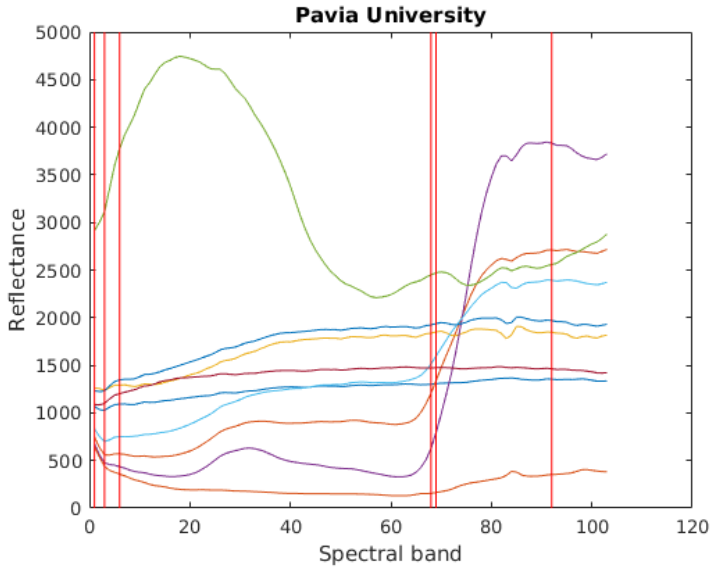


Figure 5.9: Mean spectral signature values of the Pavia University image classes. The vertical lines indicate the location of the first 6 bands selected by the proposed `CSLN` method.

If the neural net does not converge to a good local minimum, or if its architecture is not appropriate for the problem at hand, no conclusion based on Equation 4.4 would be reliable.

According to the framework proposed in this chapter, the binary classification problem addressed by the single-layer neural network comes from the clustering performed by the k-Means algorithm. Thus, the two groups are linearly separable, as shown in Figures 5.10 and 5.11, where the straight lines that separate the two clusters are calculated using a single-layer neural network. The dimension of the datasets was reduced by the Principal Components Analysis, whose the first two principal components—PC1 and PC2, respectively—are kept, for a 2D illustration.

It is worth-noting that the two groups are linearly separable not only in a sparse feature space—Figure 5.10—, but also in a more dense situation, such as in Figure 5.11. Consequently, it is reasonable to use a single-layer neural network in such situations.

5.2.5 Remarks about the results

5.2.5.1 KNN versus CART

For the Indian Pines image, KNN results are, in general, superior than that of CART: 73.26% and 63.13%, respectively. This may be attributed to the fact that CART splits the feature space into regions that correspond to the classes. Therefore, if $\mathbf{x}_i$ is found in a region corresponding to a class α, for example, it will be classified as α, even if it

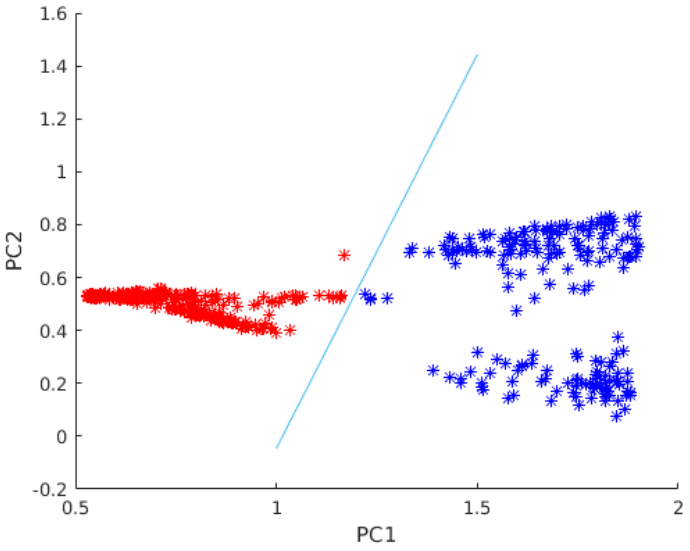


Figure 5.10: Two Indian Pines clusters. The straight line is calculated by a single-layer neural network.

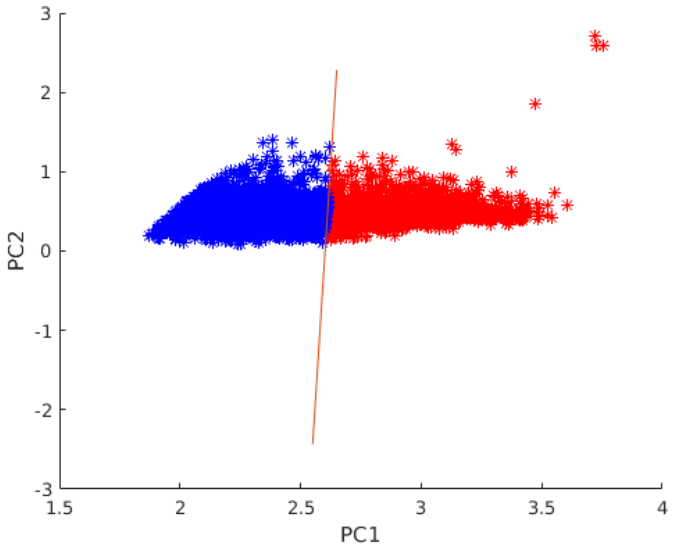


Figure 5.11: Two clusters from the Pavia University image. The straight line that separates the groups is calculated by a single-layer neural network.

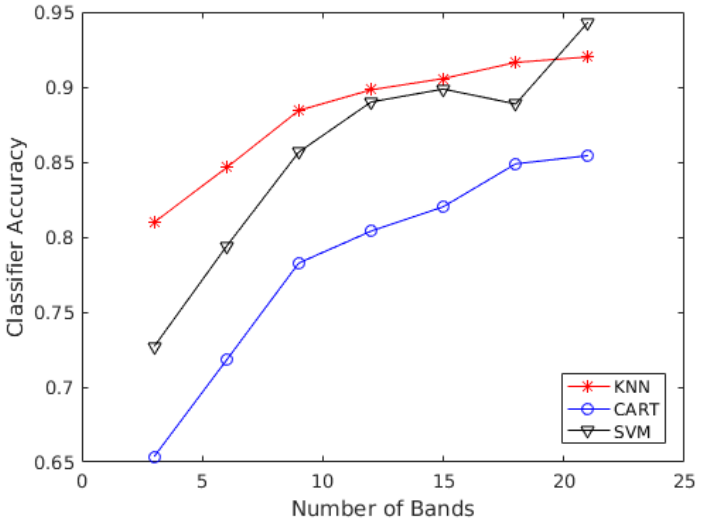


Figure 5.12: Mean results of the three classifiers used for the Pavia University image.

belongs to class β. Whereas, in this same situation, KNN would analyze the k nearest neighbors of $\mathbf{x}_i$ before assigning it a label. Consequently, KNN outperforms CART when the class boundaries are highly non-linear.

5.2.5.2 KNN, CART and SVM

Concerning the Pavia University image, the mean results for the KNN, CART and SVM are, respectively, 88.30%, 78.31% and 85.69%. This shows a slight superiority of KNN in relation to SVM, what is somehow unexpected in high dimensional feature spaces, as shown in Fig. 5.12.

It is worth mentioning that our objective in this thesis is the classification comparison amongst different BS methods, and not the best attainable classification result. For this, further studies on the classifiers hyperparameters would be necessary.

5.2.5.3 Band selection methods

As for the band selection methods, using the Indian Pines image, the proposed BS framework achieves the best results in 5 out of 12 experiments, whereas the competitor have 4/12, 2/12, 1/12 and 0/12.

For the Pavia University image, our method gets 14/21, and the competitors 7/21 and 0/21.

The `CSLN` framework not only gets superior results than its competitors, but it is also easily implementable. Thus, one can conclude that it is a good method for band selection.

Chapter 6

Unsupervised Band Selection using Autoencoder

Like Chapter 5, in this chapter we insist on the unsupervised issue. Indeed, methods that do not need the data class information are more likely to be used in several occasions, because unlabeled data are more abundant. Furthermore, as already stated in Chapter 2.3, giving labels to hyperspectral data is an expensive task.

Consequently, in this chapter a new unsupervised BS is proposed. It takes advantage of the intrinsic unsupervised nature of autoencoders, which also explore the structure of the dataset.

Basically, our method inserts a masking noise in the input samples during the training phase of the autoencoder. The reconstruction error is measured taking into account the uncorrupted input vector, thus it is possible to assess the importance of the missing information. The correlation amongst adjacent bands is softened in the hidden layer of the autoencoder.

The BS framework proposed in this chapter gives as output the ranking of all the spectral bands. The bigger the rank, the more important the band is.

6.1 Proposed method

In this section, the proposed unsupervised band selection approach is described.

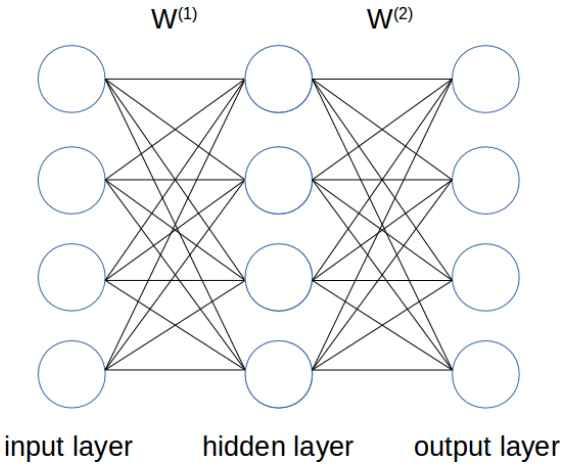


Figure 6.1: Example in reduced size of the autoencoder used in the proposed framework. All the layers have d neurons. $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$ are the sets of weights.

6.1.1 Definitions

Let $\mathbf{X}$ be the hyperspectral dataset whose elements $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ represent the pixels of this image, with $i = 1, 2, ..., n$, where n is the cardinality of $\mathbf{X}$ and d is quantity of spectral bands.

Let $N : \mathbb{R}^{d\times 1} \rightarrow \mathbb{R}^{d\times 1}$ be an autoencoder whose hidden layer has d neurons. Its cost function is quadratic. And let $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$ be the matrices of weights between the input and hidden layers and between the hidden and output layers, respectively, as shown in Fig. 6.1. And let $\mathbf{b}^{(1)}$ and $\mathbf{b}^{(2)}$ be the vectors of biases of the hidden and output layers, respectively. Let $o_s \in \mathbf{o}$ be the output of the s^{th} neuron of the hidden layer, with $s = 1, 2, \ldots, d$, and $\mathbf{o} \in \mathbb{R}^{d\times 1}$. Let $\mathbf{A}$ be the set containing the original d spectral bands. And let $\mathbf{r} \in \mathbb{R}^{d\times 1}$ be a vector, whose element r_h is the ranking of the band $a_h \in \mathbf{A}$, with $h = 1, 2, \ldots, d$.

Let $\mathbf{y}_i \in \mathbb{R}^{d\times 1}$ be the output of the autoencoder Θ.

Finally, let $sig()$ be the sigmoid function,

$$sig(z) = \frac{1}{1 + e^{-z}}, \tag{6.1}$$

which will be used as activation function of the autoencoder. Since every $x_i^k \in \mathbf{x}_i$ belongs to the $[0, 1]$ interval, we adopted the sigmoid as the activation function, because the range of this function is also contained in the $[0, 1]$ interval.

6.1.2 Description

Autoencoders seek to reconstruct in their output layer, the encoded information of the input vector.

Therefore, autoencoders are composed of two parts:

- *Encoder:* The encoding of the input vector takes place in the hidden layer; and
- *Decoder:* The output layer performs the decoding process.

Mathematically, the output $\mathbf{y}_i = \Theta(\mathbf{x}_i)$ is defined as

$$\mathbf{y}_i = sig(\mathbf{W}^{(2)}\mathbf{o} + \mathbf{b}^{(2)}), \tag{6.2}$$

where the vector $\mathbf{o}$ is given by

$$\mathbf{o} = sig(\mathbf{m} + \mathbf{b}^{(1)}), \tag{6.3}$$

where $\mathbf{m}$ is a vector containing d multiplicative aggregation functions $m_k()$.

6.1.2.1 Multiplicative aggregation function

The multiplicative aggregation function (MAF) is an important component of the proposed framework, and this function is used to soften the redundancy present in the dataset. MAFs are placed in the hidden layer, and in order to exploit the correlation amongst all bands, the input and hidden layers must have the same size. Thus, less redundant information will be fed towards the output layer.

In this thesis, we propose a MAF simpler than the one proposed in (Chandra and Sharma, 2015). It also yields simpler equations for the back-propagation algorithm.

For each neuron of the hidden layer, there is an associated multiplicative aggregation function $m_k : \mathbb{R}^{d\times 1} \rightarrow \mathbb{R}$, with $k \in \{1, 2, \ldots, d\}$, given by

$$m_k(\mathbf{x}_i) = x_i^k (w_{(1)}^{kk})^2 (1 + \sum_{l\neq k} -2\rho_{lk}^2 w_{(1)}^{kl} x_i^l), \tag{6.4}$$

where ρ_{lk} is the correlation between the bands l and k, and the weights $w_{(1)} \in \mathbf{W}^{(1)}$.

Finally, the output $o_k \in \mathbf{o}$ of each hidden neuron is

$$o_k = sig(m_k(\mathbf{x}_i) + b_{(1)}^k), \tag{6.5}$$

where $b_{(1)}^k \in B^{(1)}$.

According to Equation (6.4), the negative summation makes m_k smaller. More precisely, the bigger the correlation amongst band k and the other bands, the smaller the value of m_k, and consequently, the smaller the magnitude of o_k.

6.1.2.2 Spectral bands ranking

The outcome of the proposed band selection framework is the ranking of all spectral bands. At the end of the whole processing, for each band $a_h \in \mathbf{A}$ there will be a correspondent $r_h \in \mathbf{r}$ indicating its ranking.

During the training of the autoencoder Θ, every input data sample $\mathbf{x}_i$ is subjected to the masking noise transform t, which has the following properties:

- each $x_i^k \in \mathbf{x}_i$ has equal probability p to be masked; and
- no position x_i^k is masked in two consecutive iterations.

Let $\mathbf{c} \in \mathbb{N}^{d \times 1}$ be a vector that is initially zero. Each time a position k of the input vector is masked, that is, $\tilde{\mathbf{x}}_i^k = 0$,

$$c_k \leftarrow c_k + 1. \tag{6.6}$$

Thus, the vector $\mathbf{c}$ counts how many times each feature is masked during the training phase of the autoencoder.

In Equation 6.7, there is an example of an input vector $\mathbf{x}_i$ along with its correspondent corrupted version $\tilde{\mathbf{x}}_i$ subjected to the masking noise. Note that in $\tilde{\mathbf{x}}_i$ some positions are flipped to zero. The bigger the probability p, the more features are masked.

$$\mathbf{x}_i = \begin{bmatrix} x_i^1 \\ x_i^2 \\ x_i^3 \\ x_i^4 \\ x_i^5 \\ \vdots \\ x_i^d \end{bmatrix} \qquad \tilde{\mathbf{x}}_i = \begin{bmatrix} x_i^1 \\ 0 \\ x_i^3 \\ x_i^4 \\ 0 \\ \vdots \\ x_i^d \end{bmatrix} \tag{6.7}$$

Let $\tilde{\mathbf{y}}_i$ be the output of the autoencoder when $\tilde{\mathbf{x}}_i$ is the input sample. That is, $\tilde{\mathbf{y}}_i = \Theta(\tilde{\mathbf{x}}_i)$, where $\tilde{\mathbf{x}}_i = t(\mathbf{x}_i)$. Likewise, $\mathbf{y}_i = \Theta(\mathbf{x}_i)$, without masking the input sample.

Initially, $\mathbf{r}^{(0)} = 0$, and at iteration q, the calculation of the rankings $r_h^{(q)} \in \mathbf{r}^{(q)}$ is

$$r_h^{(q)} = \frac{1 + v_1}{1 + v_2} + r_h^{(q-1)}, \tag{6.8}$$

when $\tilde{x}_i^h$ is masked. Where

$$v_1 = \sum_{k=1}^{d} (\tilde{y}_i^k - x_i^k)^2 \tag{6.9}$$

and

$$v_2 = \sum_{k=1}^{d} (y_i^k - x_i^k)^2. \tag{6.10}$$

From Equation 6.9, we see that v_1 measures the reconstruction error between the corrupted output $\tilde{\mathbf{y}}_i$ and the input $\mathbf{x}_i$. If v_1 takes small values, it means that the autoencoder Θ can reconstruct $\tilde{\mathbf{x}}_i$ without any difficulties. Therefore, the masked features of $\tilde{\mathbf{x}}_i$ cannot be considered as unimportant information. On the other hand, if v_1 takes big values, one may infer that the masked bands are important to accomplish the input vector reconstruction. In fact, the extent to which $v_1 > v_2$ indicates the importance of the masked features—or bands.

The parameters update of the autoencoder is done by the back-propagation algorithm, based on the quadratic error e between the output with masked input and the input without masking noise. That is,

$$e = \frac{1}{2}(\tilde{\mathbf{y}}_i - \mathbf{x}_i)^2. \tag{6.11}$$

Algorithm 3 shows the steps of the proposed BS method. The indices of the biggest values of $\mathbf{r}$ are those of the best bands to be selected. As stated in the step 10 of Algorithm 3, the division $\mathbf{r}/\mathbf{c}$ means r_k/c_k, for $k = 1, 2, \ldots, d$. This measure assures that all spectral bands will be equally compared in the final ranking.

Algorithm 3 Proposed method.

1: input : $\mathbf{X}$
2: initialize: $\mathbf{r}^{(0)} = 0$
3: **for** $q = 1$: MaxIterations **do**
4: $\quad \mathbf{y}_i = \Theta(\mathbf{x}_i)$
5: $\quad \tilde{\mathbf{y}}_i = \Theta(\tilde{\mathbf{x}}_i)$
6: $\quad$ Update $\mathbf{r}^{(q)}$ using Equation (6.8)
7: $\quad$ Update $\mathbf{c}$ using Equation (6.6)
8: $\quad$ Update the weights and biases of Θ using the back-propagation algorithm, according to the error calculated in Equation (6.11)
9: **end for**
10: return: $\mathbf{r}/\mathbf{c}$

6.2 Results

In this section, the results of the proposed method are shown. Furthermore, they will be compared with other BS methods by analyzing the accuracy of two supervised classifiers —k-Nearest Neighbors and Classification and Regression Trees—, which have as input the selected bands.

The dataset is the Indian Pines image. Regarding the ground truth, there are 16 classes, which are used only for classification comparison purposes.

6.2.1 Competitors

The band selection performance of the proposed method, which will be called `AE`, is compared with four other methods from the literature. They are:

- One method is clustering-based (Martinez-Uso et al., 2007), which will be referred to as `WaLuDi`;
- The other approach uses both clustering and ranking techniques for band selection (Datta et al., 2015), which will be called `CR`;
- Another competitor uses band elimination with partitioned image correlation (Datta et al., 2014), and this method will be called `EM`;
- This competitor is based on information divergence, and it will be referred to as `ID` (Chang and Wang, 2006).

6.2.2 Masking noise percentage

In (Chandra and Sharma, 2015), each feature $x_i^k \in \mathbf{x}_i$ has a probability of $p = 0.25$ to be masked. However, in this work we run our algorithm with ten different probability values, with $p_v = \frac{2.5v}{100}$, where $v = 1, 2, \ldots, 10$. For each p_v, we summed up the reconstruction error for every input data sample, according to Equation 6.11. Figure 6.2 shows the reconstruction error for each masking noise probability. The masking noise probability that yielded the smallest reconstruction error was 7.5%.

The smaller the error, the better the autoencoder can reconstruct the input vector. If an autoencoder can properly reconstruct an input vector, it means that this neural network could sufficiently learn about the structure of the dataset. Consequently, this noise probability of 7.5% is kept throughout this work.

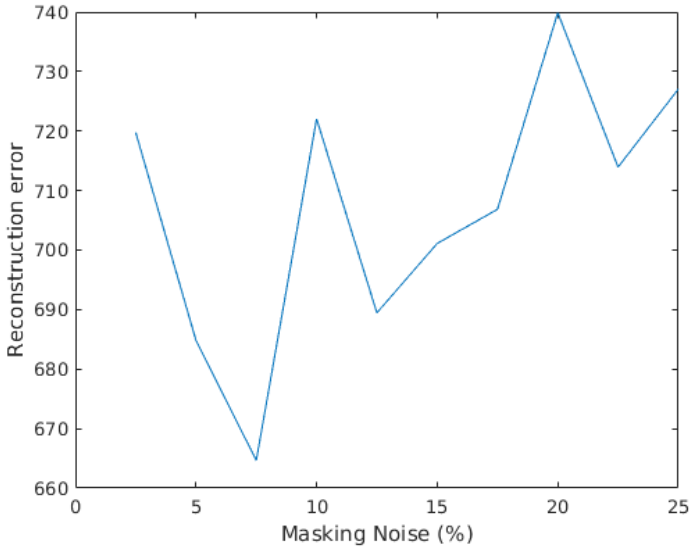


Figure 6.2: Reconstruction error with different masking noise probabilities. The lower the reconstruction error, the better the autoencoder.

6.2.3 Selected bands

Firstly, the bands selected by the proposed method can be found in Table 6.1. We let them available to other researchers who may be interested in comparing results. Because we have access to only the first 18 best-ranked bands of our competitors, we restrict the analysis of results to this number of bands.

It is worth-mentioning that the bands in Table 6.1 are placed in descending order of importance. That is, $r_{43} > r_{13} > r_{133} > \cdots > r_{99} > r_{215}$. For example, the value in the 43^{rd} position of the vector $\mathbf{r}$ is the biggest. This indicates that the 43^{rd} spectral band is the most important, according to the proposed method.

Table 6.1: Selected bands in order of importance, according to the rankings $\mathbf{R}$.

Selected bands	43, 13, 133, 190, 174, 123, 82, 118, 209, 144, 73, 98, 11, 137, 77, 106, 99, 215.

6.2.4 Results comparison

All the classification results shown in this chapter are the mean values over ten runs. The standard-deviation values are also calculated.

In Table 6.2, the results of the KNN classifier are shown. The proposed method `AE` achieves the best results in almost all cases. It is illustrated in Figure 6.3.

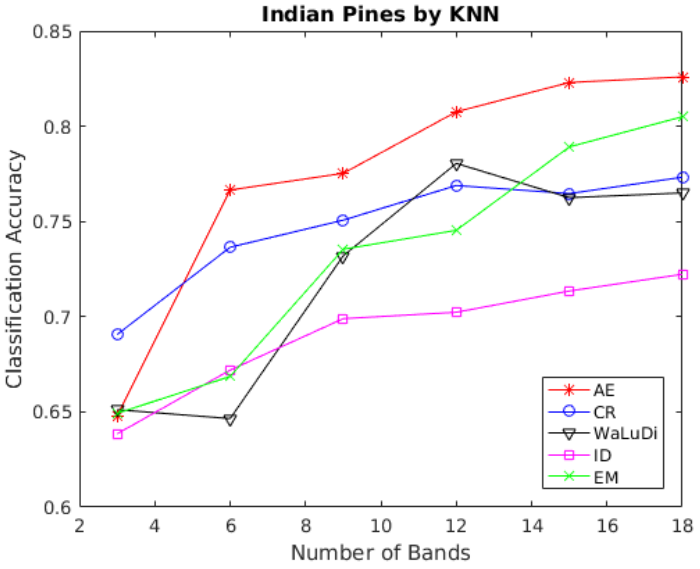


Figure 6.3: The Indian Pines dataset under the KNN classification. The proposed `AE` framework gets the best results in almost all cases.

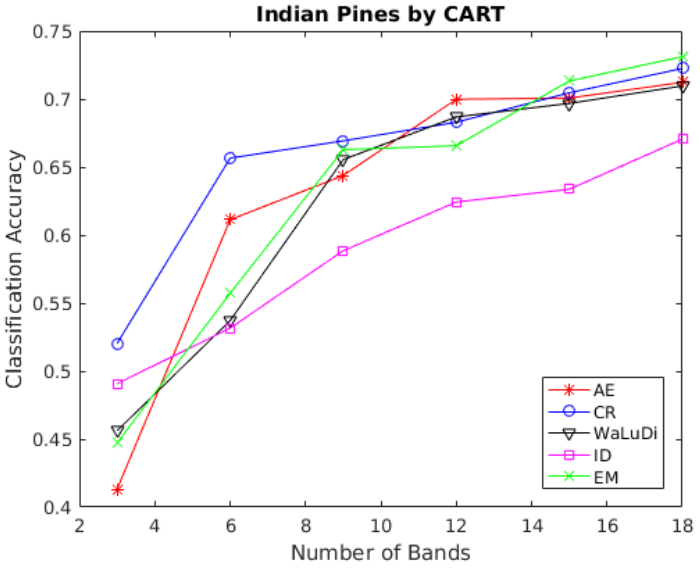


Figure 6.4: The Indian Pines hyperspectral image classified by CART. The proposed BS method achieves the best result in only one case.

Table 6.3 shows the overall results achieved by the CART classifier. The proposed method has the best result in only one case, with 12 spectral bands. In Figure 6.4, it is possible to have a visual idea of the results.

Table 6.2: KNN results.

	3 bands		6 bands		9 bands		12 bands		15 bands		18 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
AE	64.79%	0.24%	**76.65%**	0.37%	**77.53%**	0.34%	**80.76%**	0.53%	**82.30%**	0.50%	**82.59%**	0.37%
WaLuDi	65.12%	1.02%	64.65%	0.25%	73.19%	0.72%	78.05%	0.56%	76.25%	0.19%	76.50%	0.67%
CR	**69.06%**	0.52%	73.65%	1.03%	75.07%	1.43%	76.89%	1.07%	76.47%	1.14%	77.32%	0.22%
EM	64.92%	1.15%	66.86%	1.03%	73.54%	0.28%	74.54%	1.07%	78.92%	0.41%	80.50%	0.52%
ID	63.85%	079%	67.20%	0.22%	69.90%	0.18%	70.23%	1.16%	71.35%	0.47%	72.23%	1.34%

Table 6.3: CART results.

	3 bands		6 bands		9 bands		12 bands		15 bands		18 bands	
Method	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
AE	41.32%	0.55%	61.13%	0.84%	64.38%	1.17%	**69.98%**	1.48%	70.07%	1.06%	71.23%	0.80%
WaLuDi	45.62%	1.00%	53.71%	1.23%	65.55%	0.95%	68.68%	0.28%	69.68%	0.75%	70.96%	1.15%
CR	**52.03%**	1.14%	**65.66%**	0.39%	**66.93%**	0.37%	68.29%	1.48%	70.46%	0.99%	72.25%	1.91%
EM	44.72%	0.93%	55.72%	1.04%	66.28%	0.52%	66.57%	1.24%	**71.33%**	0.76%	**73.12%**	0.51%
ID	49.07%	0.82%	53.16%	1.35%	58.85%	1.42%	62.43%	1.67%	63.37%	1.01%	67.06%	0.87%

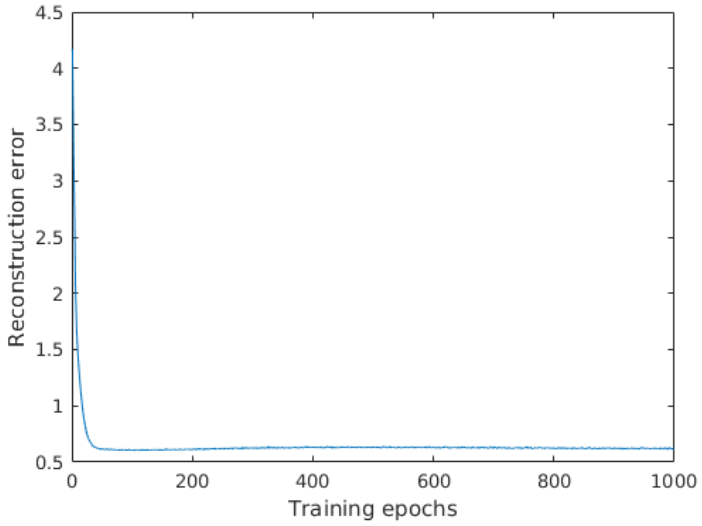


Figure 6.5: The autoencoder reconstruction error over training epochs.

6.2.4.1 Reconstruction-error based ranking

As stated in the step 10 of Algorithm 3, the ranking vector $\mathbf{r}$ is updated by taking into consideration the number of times each feature of the input vector $\mathbf{x}_i$ is masked.

As the autoencoder learns over the training epochs, it is reasonable to assume that the rankings calculated —according to Equation 6.8—in the last training epochs could have a bigger importance than the rankings calculated during the first training epochs. However, as shown in Figure 6.5, the autoencoder converges to a minimum reconstruction error very quickly, and it keeps this situation until the last training epoch.

Since this minimum position is kept throughout most of the training epochs, we consider that all the rankings calculations have the same importance, under the *reconstruction error* criterion. Consequently, the step 10 of Algorithm 3 suffices as a ranking update measure.

6.2.4.2 Visual inspection of the selected bands

It is also useful to have a visual idea of the selected bands.

In Figure 6.6, the mean values of the spectral signature for each of the 16 classes of the Indian Pines image are displayed. The red vertical lines indicate the location of the first six selected bands, to avoid excessive visual information. The red circle near the 100^{th} spectral band indicates a place where all the spectral signatures seem to merge. There are more regions like that in this *spectral band* $\times$ *reflectance* plot.

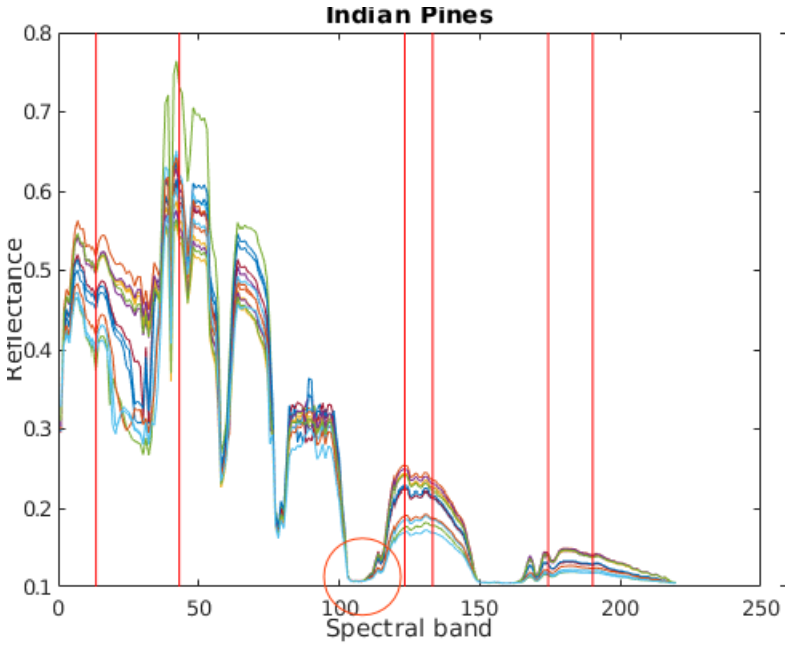


Figure 6.6: Mean spectral signature values of the Indian Pines image classes. The vertical lines indicate the location of the first 6 bands selected by the proposed `AE` framework. The red circle, near the band 100, indicates an example of region to be avoided, where all the spectral signatures seem to merge.

It is important to note that all the first six selected bands fall in regions where the spectral signatures are somehow spread, avoiding regions whose spectral bands are not discriminative—for example, the red circle.

6.2.5 Remarks about the results

6.2.5.1 KNN versus CART

In general, KNN results are superior than CART accuracies, 73.36% and 62.65%, respectively. It happens due to the fact that CART divides the feature space into several regions, one for each class. So, once a x_i falls in a region that belongs to the class α, for instance, it will be given the label α, even if it belongs to class β. Whereas KNN would inquire the k nearest neighbors of x_i before giving it a label. For this reason, KNN is better than CART in highly non-linear separating boundaries.

Another worth-mentioning fact is that the accuracies increase as more bands are used, as shown in Tables 6.2 and 6.3.

Figure 6.7 shows that KNN has better accuracies than CART, considering all the BS methods together. Moreover, it is possible to see that the more bands are used, the better the classifier accuracies.

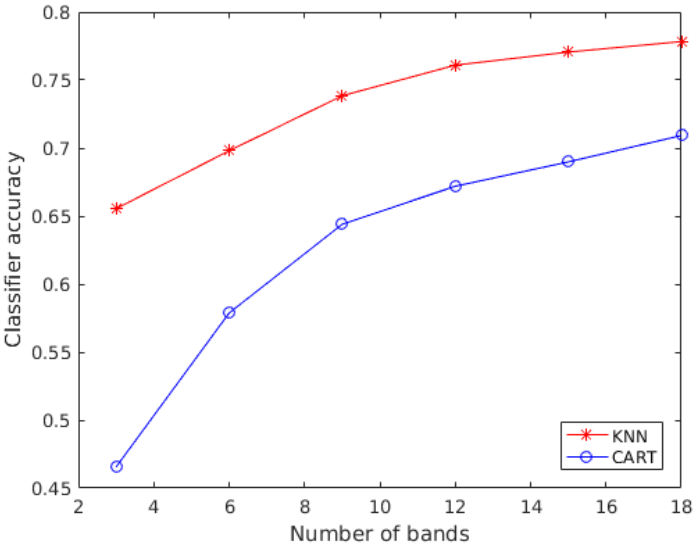


Figure 6.7: Mean results of all methods together.

6.2.5.2 BS methods comparison

When it comes to the band selection approaches, considering the KNN classifier, the proposed method has the best results in 5 out of 6 situations. Furthermore, the standard-deviation values shown in Table 6.2 indicate that the `AE` method have statistically better results.

Considering the CART classifier, the proposed method achieves the best accuracy only with 12 bands, that is, in 1 out of 6 situations. With 9, 15 and 18 bands, our method `AE` has similar results in relation to its competitors.

In general, considering all the 12 situations—6 for KNN and 6 for the CART classifier—, the final score is thus:

1. `AE` (the proposed method): 6 best results;
2. `CR:` 4 best results;
3. `EM:` 2 best results; and
4. `WaLuDi` and `ID:` 0 best result.

Thus, it is possible to see that the proposed method has the best outcome, compared to its competitors.

Chapter 7

Conclusion

7.1 Conclusions

Hyperspectral images provide rich spectral information about the scene under analysis as a result of their both numerous and contiguous bands. Since different materials have distinct spectral signatures, objects with similar characteristics in terms of colors and shape may still be distinguished in the spectral domain that goes beyond the visual range.

In a pattern recognition system, the huge amount of data contained in HSIs may pose problems in terms of data storage and transmission, or cause the overfitting of the classifier in case of insufficient training data. One way to solve or, at least, reduce those troubles is to resort to band selection, because it decreases the size of the dataset while keeping the useful information.

In supervised frameworks, it is possible to proceed to the band selection by taking into account the class information of the data samples. Thus, classes separability measures, for example, can be used to assess the validity of the selected spectral bands. As for the unsupervised approaches, the BS task is done by selecting the bands that keep the original data structure.

In this thesis, we proposed three band selection frameworks. Two unsupervised and one of them is supervised, whose conclusions are given in the sequel.

7.1.1 Supervised Band Selection using Single-Layer Neural Network

In this context, the present thesis proposed a supervised filter-based band selection framework based on single-layer neural networks using only 20% of the available training data.

For each class in the data set, a binary classification into class and non-class was performed, and the bands corresponding to the largest and smallest weights were selected. During this iterative process, the bands most correlated with the bands selected are automatically discarded, according to a procedure also proposed in this thesis. In general, the proposed method may be seen as a class-oriented band selection approach, allowing a BS criterion that meets the needs of each class.

A number of other filter-based BS algorithms perform their choice of bands based, for instance, on statistical properties of the data set. A positive aspect of the filter-based method proposed in this thesis is that it is based on classification, that is, it uses a linear classifier to rank and select the bands. The proposed method outperformed its competitors in 43% of the cases analyzed in this thesis.

As a secondary conclusion, we showed that wrapper-based approaches are not necessarily better than their filter counterparts, when using different classifiers for the band selection process and for the classification. More research on this subject is necessary.

7.1.2 Unsupervised Clustering-based Band Selection using Single-Layer Neural Network

The proposed unsupervised BS method is based on k-Means clustering and single-layer neural networks.

It starts by clustering the whole data set into two groups. Then, a single-layer neural network is used to find a separating hyperplane between the clusters. The bands linked to the biggest and smallest coefficients of the hyperplane equation are selected. Then, this procedure is repeated using the generated clusters to select the desired number of bands.

By analyzing the results, one could see that the proposed method outperformed its competitors in both datasets analyzed. More specifically, using the Indian Pines image, our method had the best results in 41.7% of the experiments, whereas the competitors achieved their best results in 33%, 17%, 8.3% and 0% of the tests. For the Pavia University dataset, our framework had the best result in 66.7% of the experiments, and the competitors had the best performances in 33.3% and 0% of the experiments.

7.1.3 Unsupervised Band Selection using Autoencoder

The last proposed BS method is based on autoencoders.

During the training phase of the autoencoder, each input data sample is subjected to a masking noise transform, which flips some features of the input vector into zero, following a given probability. Then, the output error is assigned to those indices with masking

noise. The errors are summed up to their respective positions during the whole training phase. At the end, there is a ranking of the bands, and the most important are the ones with the biggest rankings.

According to the results, one could conclude that the KNN classifier is better than CART for the Indian Pines image, 73.36% and 62.65% of accuracy, respectively. Also, the bigger the number of bands, the better the classifier accuracy. It is worth noting that we selected from 3 up to 18 spectral bands.

Regarding the proposed method, it achieved the best results in almost all situations using the KNN classifier. With the CART classifier, the proposed method got the best results in one situation and similar to other competitors' results in other situations. In general, our method had the best results in 50% of the experiments, whereas the competitors achieved the best marks in 33.3%, 16.7% and 0% of the tests.

7.2 Perspectives

The present work is still in progress. In this document, we presented the latest version of the three methods we have been devising during the last three years. Indeed, the more we work on the frameworks, the more ideas pop out.

Datasets types: In this thesis, we devised band selection methods and performed tests and analyses using hyperspectral datasets acquired by satellites. In future developments, we intend to use UAV-borne sensors and try to use our methods in such equipment.

Furthermore, it will be important to verify the performance of our methods with multispectral images, in which the spectral bands are not contiguous nor numerous. We could use, for example, the Sequoia sensor[1].

Input data type: All the proposed BS methods are designed to work in a pixel-wise fashion. We did so for two reasons:

- To be consistent with the datasets, whose class information is associated with pixels, and not image patches (for the supervised method); and
- To be consistent with the algorithm that we used (autoencoder).

Concerning future developments of our methods, it would be interesting to test other frameworks, like convolutional neural networks.

[1] https://www.korecgroup.com/product/parrot-sequoia-sensor/

7.2.1 Supervised Band Selection using Single-Layer Neural Network

A next step of this framework is to devise a methodology in order to find the optimum number of bands to be selected for a given application and image. Thus, a specialist would not be necessary.

Also, as already seen, when the number of bands to be selected in less than the number of classes, not all the classes can indicate the bands to be selected. In future works, we could devise a method that takes into account the importance of classes, and prioritize the most important during the band selection.

7.2.2 Unsupervised Clustering-based Band Selection using Single-Layer Neural Network

With regard to the future works, we will investigate other clustering algorithms and binary classifiers and use them in our framework.

Moreover, we will investigate the impact of calculating the covariance matrix of the dataset and use it during the clustering procedure. Thus, we will try other distances measurements rather than Euclidean.

7.2.3 Unsupervised Band Selection using Autoencoder

Concerning the future works, we will investigate some heuristics to choose the features to be masked, instead of using a uniform distribution.

Bibliography

Adlakha, A. and Chhikara, R. R. (2016). Comparative analysis of filter feature selection techniques with different classifiers for image steganalysis. In *2016 International Conference on Computing, Communication and Automation (ICCCA)*, pages 1122–1127.

Antonio, R.-K. and Huynh, C. P. (2012). *Imaging Spectroscopy for Scene Analysis*. Springer Publishing Company, Incorporated.

Bai, E. W., Cheng, C., Zhao, W., and Chen, H. F. (2017). Variable selection of high-dimensional non-parametric nonlinear systems: A way to avoid the curse of dimensionality. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 6469–6474.

Bai, J., Xiang, S., Shi, L., and Pan, C. (2015a). Semisupervised pair-wise band selection for hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2798–2813.

Bai, X., Guo, Z., Wang, Y., Zhang, Z., and Zhou, J. (2015b). Semisupervised hyperspectral band selection via spectral spatial hypergraph model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2774–2783.

Banerjee, S., Choudhary, A., and Pal, S. (2015). Empirical evaluation of k-means, bisecting k-means, fuzzy c-means and genetic k-means clustering algorithms. In *2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pages 168–172.

Bilbao, I. and Bilbao, J. (2017). Overfitting problem and the over-training in the era of data: Particularly for artificial neural networks. In *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pages 173–177.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Bitar, A. W., Ovarlez, J. P., and Cheong, L. F. (2017). Sparsity-based cholesky factorization and its application to hyperspectral anomaly detection. In *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 1–5.

Bollobás, B. (1998). *Modern Graph Theory*. Graduate texts in mathematics. Springer, Heidelberg, corrected edition.

Braga, J. R. G., d. C. Velho, H. F., and Shiguemori, E. H. (2015). Estimation of uav position using lidar images for autonomous navigation over the ocean. In *2015 9th International Conference on Sensing Technology (ICST)*, pages 811–816.

Braga, J. R. G., Velho, H. F. C., Conte, G., Doherty, P., and Shiguemori, E. H. (2016). An image matching system for autonomous uav navigation based on neural network. In *2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1–6.

Bris, A. L., Chehata, N., Briottet, X., and Paparoditis, N. (2014). Use intermediate results of wrapper band selection methods: A first step toward the optimization of spectral configuration for land cover classifications. In *2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4.

Buyukyazi, T., Bayraktar, S., and Lazoglu, I. (2013). Real-time image stabilization and mosaicking by using ground station cpu in uav surveillance. In *2013 6th International Conference on Recent Advances in Space Technologies (RAST)*, pages 121–126.

Camps-Valls, G., Mooij, J., and Scholkopf, B. (2010). Remote sensing feature selection by kernel dependence measures. *IEEE Geoscience and Remote Sensing Letters*, 7(3):587–591.

Cao, X., Li, X., Li, Z., and Jiao, L. (2017a). Hyperspectral band selection with objective image quality assessment. *International Journal of Remote Sensing*, 38(12):3656–3668.

Cao, X., Wei, C., Han, J., and Jiao, L. (2017b). Hyperspectral band selection using improved classification map. *IEEE Geoscience and Remote Sensing Letters*, PP(99):1–5.

Cao, X., Wu, B., Tao, D., and Jiao, L. (2016a). Automatic band selection using spatial-structure information and classifier-based clustering. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(9):4352–4360.

Cao, X., Xiong, T., and Jiao, L. (2016b). Supervised band selection using local spatial information for hyperspectral image. *IEEE Geoscience and Remote Sensing Letters*, 13(3):329–333.

Chandra, B. and Sharma, R. K. (2015). Exploring autoencoders for unsupervised feature selection. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6.

Chang, C. I., Lee, L. C., Xue, B., Song, M., and Chen, J. (2017). Channel capacity approach to hyperspectral band subset selection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(10):4630–4644.

Chang, C.-I. and Wang, S. (2006). Constrained band selection for hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 44(6):1575–1585.

Cui, M., Prasad, S., Mahrooghy, M., Bruce, L. M., and Aanstoos, J. (2011). Genetic algorithms and linear discriminant analysis based dimensionality reduction for remotely sensed image analysis. In *2011 IEEE International Geoscience and Remote Sensing Symposium*, pages 2373–2376.

da Silva, W., Shiguemori, E. H., Vijaykumar, N. L., and d. C. Velho, H. F. (2015). Estimation of uav position with use of thermal infrared images. In *2015 9th International Conference on Sensing Technology (ICST)*, pages 828–833.

Damodaran, B. B., Courty, N., and Lefevre, S. (2016). Unsupervised classifier selection approach for hyperspectral image classification. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5111–5114.

Damodaran, B. B., Courty, N., and Lefèvre, S. (2017). Sparse hilbert schmidt independence criterion and surrogate-kernel-based feature selection for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(4):2385–2398.

Datta, A., Ghosh, S., and Ghosh, A. (2012). Clustering based band selection for hyperspectral images. In *2012 International Conference on Communications, Devices and Intelligent Systems (CODIS)*, pages 101–104.

Datta, A., Ghosh, S., and Ghosh, A. (2014). Band elimination of hyperspectral imagery using partitioned band image correlation and capacitory discrimination. *International Journal of Remote Sensing*, 35(2):554–577.

Datta, A., Ghosh, S., and Ghosh, A. (2015). Combination of clustering and ranking techniques for unsupervised band selection of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2814–2823.

dos Santos, L. C. B., Guimarães, S. J. F., and dos Santos, J. A. (2015). Efficient unsupervised band selection through spectral rhythms. *IEEE Journal of Selected Topics in Signal Processing*, 9(6):1016–1025.

Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern Classification*. Wiley.

Eiben, A. E. and Smith, J. E. (2015). *Introduction to Evolutionary Computing*. Springer Publishing Company, Incorporated, 2nd edition.

Emmanuel Arzuaga-Cruz, Luis O. Jimenez-Rodriguez, M. V.-R. (2003). Unsupervised feature extraction and band subset selection techniques based on relative entropy criteria for hyperspectral data analysis.

Fauvel, M., Dechesne, C., Zullo, A., and Ferraty, F. (2015). Fast forward feature selection of hyperspectral images for classification with gaussian mixture models. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2824–2831.

Feng, S., Itoh, Y., Parente, M., and Duarte, M. F. (2017). Hyperspectral band selection from statistical wavelet models. *IEEE Transactions on Geoscience and Remote Sensing*, 55(4):2111–2123.

Freitas, S., Almeida, C., Silva, H., Almeida, J., and Silva, E. (2018). Supervised classification for hyperspectral imaging in uav maritime target detection. In *2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 84–90.

Gan, L., Xia, J., Du, P., and Xu, Z. (2017). Dissimilarity-weighted sparse representation for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 14(11):1968–1972.

Gong, M., Zhang, M., and Yuan, Y. (2016). Unsupervised band selection based on evolutionary multiobjective optimization for hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(1):544–557.

Habermann, M., Fremont, V., and Shiguemori, E. H. (2017a). Feature selection for hyperspectral images using single-layer neural networks. In *8th International Conference of Pattern Recognition Systems (ICPRS 2017)*, pages 1–6.

Habermann, M., Fremont, V., and Shiguemori, E. H. (2017b). Problem-based band selection for hyperspectral images. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1800–1803.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC.

Haykin, S. S. (2009). *Neural networks and learning machines*. Pearson Education, Upper Saddle River, NJ, third edition.

Hua, W., Guo, T., and Liu, X. (2015). Camouflage target reconnaissance based on hyperspectral imaging technology.

Jahanshahi, S. (2016). Maximum relevance and class separability for hyperspectral feature selection and classification. In *2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*, pages 1–4.

Jiao, L., Feng, J., Liu, F., Sun, T., and Zhang, X. (2015). Semisupervised affinity propagation based on normalized trivariable mutual information for hyperspectral band selection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2760–2773.

Kang, S., Kim, D., and Cho, S. (2016). Efficient feature selection-based on random forward search for virtual metrology modeling. *IEEE Transactions on Semiconductor Manufacturing*, 29(4):391–398.

Keshava, N. (2004). Distance metrics and band selection in hyperspectral processing with applications to material identification and spectral libraries. *IEEE Transactions on Geoscience and Remote Sensing*, 42(7):1552–1565.

Khorram, S., van der Wiele, C.F., K., F.H., N., S.A.C., and Potts, M. (2016). *Principles of Applied Remote Sensing*. Springer.

Klemas, V. V. (2014). Advances in coastal wetland remote sensing. In *2014 IEEE/OES Baltic International Symposium (BALTIC)*, pages 1–16.

Klima, R. L., Izenberg, N. R., Holsclaw, G. M., Helbert, J., D'Amore, M., McClintock, W. E., and Solomon, S. C. (2014). Visible to near-infrared hyperspectral measurements of mercury: Challenges for deciphering surface mineralogy. In *2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273 – 324. Relevance.

Kuroswiski, A. R., de Oliveira, N. M. F., and Shiguemori, E. H. (2018). Autonomous long-range navigation in gnss-denied environment with low-cost uav platform. In *2018 Annual IEEE International Systems Conference (SysCon)*, pages 1–6.

Li, J. and Hao, P. (2007). Hierarchical structuring of data on manifolds. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Li, J. and Liu, H. (2017). Challenges of feature selection for big data analytics. *IEEE Intelligent Systems*, 32(2):9–15.

Liu, Q., Li, P., Zhao, W., Cai, W., Yu, S., and Leung, V. C. M. (2018). A survey on security threats and defensive techniques of machine learning: A data driven view. *IEEE Access*, 6:12103–12117.

Lu, G. and Fei, B. (2014). Medical hyperspectral imaging: a review. *Journal of biomedical optics*, 19 1:10901.

Lu, X., Hu, Z., and Guo, S. (2009). The quantitative estimation of periurban vegetation ecology using hyperspectral remote sensing. In *2009 Joint Urban Remote Sensing Event*, pages 1–6.

Luo, R., Liao, W., Philips, W., and Pi, Y. (2015). An improved semi-supervised local discriminant analysis for feature extraction of hyperspectral image. In *2015 Joint Urban Remote Sensing Event (JURSE)*, pages 1–4.

Luo, X., Xue, R., and Yin, J. (2017). Information-assisted density peak index for hyperspectral band selection. *IEEE Geoscience and Remote Sensing Letters*, 14(10):1870–1874.

Ma, L., Li, M., Gao, Y., Chen, T., Ma, X., and Qu, L. (2017). A novel wrapper approach for feature selection in object-based image classification using polygon-based cross-validation. *IEEE Geoscience and Remote Sensing Letters*, 14(3):409–413.

Martinez-Uso, A., Pla, F., Sotoca, J. M., and Garcia-Sevilla, P. (2007). Clustering-based hyperspectral band selection using information measures. *IEEE Transactions on Geoscience and Remote Sensing*, 45(12):4158–4171.

Molina, L. C., Belanche, L., and Nebot, A. (2002). Feature selection algorithms: a survey and experimental evaluation. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, pages 306–313.

Molisch, A. (2005). *Wireless Communications*. Wiley-IEEE Press.

Monteiro, S. T. and Murphy, R. J. (2011). Embedded feature selection of hyperspectral bands with boosted decision trees. In *2011 IEEE International Geoscience and Remote Sensing Symposium*, pages 2361–2364.

Murphy, R. J., Monteiro, S. T., and Schneider, S. (2012). Evaluating classification techniques for mapping vertical geology using field-based hyperspectral sensors. *IEEE Transactions on Geoscience and Remote Sensing*, 50(8):3066–3080.

Murugan, D., Garg, A., Ahmed, T., and Singh, D. (2016). Fusion of drone and satellite data for precision agriculture monitoring. In *2016 11th International Conference on Industrial and Information Systems (ICIIS)*, pages 910–914.

Narendra, P. M. and Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.*, 26(9):917–922.

Oliveira, H. C., Guizilini, V. C., Nunes, I. P., and Souza, J. R. (2018). Failure detection in row crops from uav images using morphological operators. *IEEE Geoscience and Remote Sensing Letters*, pages 1–5.

Patra, S., Modi, P., and Bruzzone, L. (2015). Hyperspectral band selection based on rough set. *IEEE Transactions on Geoscience and Remote Sensing*, 53(10):5495–5503.

Pawlak, Z. (1992). *Rough Sets: Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publishers, Norwell, MA, USA.

Pessoa, A. S. A., Stephany, S., and Fonseca, L. M. G. (2011). Feature selection and image classification using rough sets theory. In *2011 IEEE International Geoscience and Remote Sensing Symposium*, pages 2904–2907.

Pu, R. (2017). *Hyperspectral Remote Sensing: Fundamentals and Practices*. Remote Sensing Applications Series. CRC Press.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850.

Roberge, V., Tarbouchi, M., and Labonté, G. (2018). Fast genetic algorithm path planner for fixed-wing military uav using gpu. *IEEE Transactions on Aerospace and Electronic Systems*, pages 1–1.

Saqui, D., Saito, J. H., Jorge, L. A. D. C., Ferreira, E. J., Lima, D. C., and Herrera, J. P. (2016). Methodology for band selection of hyperspectral images using genetic algorithms and gaussian maximum likelihood classifier. In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 733–738.

Schowengerdt, R. A. (2006). *Remote Sensing, Third Edition: Models and Methods for Image Processing*. Academic Press, Inc., Orlando, FL, USA.

Scott, D. W. (1992). *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, Inc, second edition edition.

Shahana, A. H. and Preeja, V. (2016). Survey on feature subset selection for high dimensional data. In *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pages 1–4.

Silva, G. R. L., Medeiros, R. R. D., Jaimes, B. R. A., Takahashi, C. C., Vieira, D. A. G., and Braga, A. D. P. (2017). Cuda-based parallelization of power iteration clustering for large datasets. *IEEE Access*, 5:27263–27271.

Silva, W. D., Habermann, M., Shiguemori, E. H., d. L. Andrade, L., and d. Castro, R. M. (2013). Multispectral image classification using multilayer perceptron and principal components analysis. In *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*, pages 557–562.

Solimini, D. (2016). *Understanding Earth Observation The Electromagnetic Foundation of Remote Sensing*. Springer.

Su, H., Cai, Y., and Du, Q. (2017). Firefly-algorithm-inspired framework with band selection and extreme learning machine for hyperspectral image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(1):309–320.

Su, H., Li, Q., and Du, P. (2014). Hyperspectral band selection using firefly algorithm. In *2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4.

Su, H., Yang, H., Du, Q., and Sheng, Y. (2011). Semisupervised band clustering for dimensionality reduction of hyperspectral imagery. *IEEE Geoscience and Remote Sensing Letters*, 8(6):1135–1139.

Su, H., Yong, B., and Du, Q. (2016). Hyperspectral band selection using improved firefly algorithm. *IEEE Geoscience and Remote Sensing Letters*, 13(1):68–72.

Sui, C., Tian, Y., Xu, Y., and Xie, Y. (2015). Unsupervised band selection by integrating the overall accuracy and redundancy. *IEEE Geoscience and Remote Sensing Letters*, 12(1):185–189.

Sun, K., Geng, X., Ji, L., and Lu, Y. (2014). A new band selection method for hyperspectral image based on data quality. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):2697–2703.

Sun, W., Tian, L., Xu, Y., Zhang, D., and Du, Q. (2017). Fast and robust self-representation method for hyperspectral band selection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(11):5087–5098.

Sun, W., Zhang, L., Du, B., Li, W., and Lai, Y. M. (2015). Band selection using improved sparse subspace clustering for hyperspectral imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2784–2797.

Sun, W., Zhang, L., Zhang, L., and Lai, Y. M. (2016). A dissimilarity-weighted sparse self-representation method for band selection in hyperspectral imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(9):4374–4388.

Suto, J., Oniga, S., and Sitar, P. P. (2016). Comparison of wrapper and filter feature selection algorithms on human activity recognition. In *2016 6th International Conference on Computers Communications and Control (ICCCC)*, pages 124–129.

Theodoridis, S. and Koutroumbas, K. (2008). *Pattern Recognition, Fourth Edition.* Academic Press, 4th edition.

Uddin, M. P., Mamun, M. A., and Hossain, M. A. (2017). Feature extraction for hyperspectral image classification. In *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pages 379–382.

Ustinov, E. A. (2015). *Sensitivity Analysis in Remote Sensing.* Springer.

Wang, C., Gong, M., Zhang, M., and Chan, Y. (2015). Unsupervised hyperspectral image band selection via column subset selection. *IEEE Geoscience and Remote Sensing Letters*, 12(7):1411–1415.

Wang, J., Zhang, K., Wang, P., Madani, K., and Sabourin, C. (2017). Unsupervised band selection using block-diagonal sparsity for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 14(11):2062–2066.

Wang, Q., Lin, J., and Yuan, Y. (2016). Salient band selection for hyperspectral image classification via manifold ranking. *IEEE Transactions on Neural Networks and Learning Systems*, 27(6):1279–1289.

Wang, Q., Zhang, F., and Li, X. (2018a). Optimal clustering framework for hyperspectral band selection. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–13.

Wang, T., Zhang, H., Lin, H., and Jia, X. (2018b). A sparse representation method for a priori target signature optimization in hyperspectral target detection. *IEEE Access*, 6:3408–3424.

Webb, A. and Copsey, K. (2011). *Statistical Pattern Recognition*. Wiley.

Wijitdechakul, J., Sasaki, S., Kiyoki, Y., and Koopipat, C. (2016). Uav-based multispectral image analysis system with semantic computing for agricultural health conditions monitoring and real-time management. In *2016 International Electronics Symposium (IES)*, pages 459–464.

Wu, J. Z., Yan, W. D., Ni, W. P., and Bian, H. (2013). Feature extraction for hyperspectral data based on mnf and singular value decomposition. In *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*, pages 1430–1433.

Xia, W., Wang, B., and Zhang, L. (2013). Band selection for hyperspectral imagery: A new approach based on complex networks. *IEEE Geoscience and Remote Sensing Letters*, 10(5):1229–1233.

Xu, X., Shi, Z., and Pan, B. (2017). A new unsupervised hyperspectral band selection method based on multiobjective optimization. *IEEE Geoscience and Remote Sensing Letters*, 14(11):2112–2116.

Yan, X., Li, Q., and Tao, S. (2017). A clustering algorithm for binary protocol data frames based on principal component analysis and density peaks clustering. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, pages 1260–1266.

Yuan, Y., Lin, J., and Wang, Q. (2016). Dual-clustering-based hyperspectral band selection by contextual analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3):1431–1445.

Yuan, Y., Zheng, X., and Lu, X. (2017). Discovering diverse subset for unsupervised hyperspectral band selection. *IEEE Transactions on Image Processing*, 26(1):51–64.

Zhan, Y., Hu, D., Xing, H., and Yu, X. (2017). Hyperspectral band selection based on deep convolutional neural network and distance density. *IEEE Geoscience and Remote Sensing Letters*, 14(12):2365–2369.

Zhang, F., Du, B., Zhang, L., and Zhang, L. (2016). Hierarchical feature learning with dropout k-means for hyperspectral image classification. *Neurocomput.*, 187(C):75–82.

Zhang, J., Weng, J., Luo, W., Liu, J. N., Yang, A., Lin, J., Zhang, Z., and Li, H. (2018). Remt: A real-time end-to-end media data transmission mechanism in uav-aided networks. *IEEE Network*, pages 12–17.

Zhang, M., Ma, J., and Gong, M. (2017a). Unsupervised hyperspectral band selection by fuzzy clustering with particle swarm optimization. *IEEE Geoscience and Remote Sensing Letters*, 14(5):773–777.

Zhang, M., Ma, J., Gong, M., Li, H., and Liu, J. (2017b). Memetic algorithm based feature selection for hyperspectral images classification. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pages 495–502.

Zhong, Y., Wang, X., Xu, Y., Jia, T., Cui, S., Wei, L., Ma, A., and Zhang, L. (2017). Mini-uav borne hyperspectral remote sensing: A review. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 5908–5911.

Zhu, G., Huang, Y., Li, S., Tang, J., and Liang, D. (2017). Hyperspectral band selection via rank minimization. *IEEE Geoscience and Remote Sensing Letters*, 14(12):2320–2324.