Extreme Learning Machines for Soybean classification in remote sensing Hyperspectral Images

Ramón Moreno[1], Francesco Corona[2], Amaury Lendasse[2], Manuel Graña[1] and Lênio S. Galvão[3]

[1]Computational Intelligence Group, UPV-EHU. Spain
[2]Aalto University, School of Science and Technology, Aalto, Finland.
[3]Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, Brazil.

November 27, 2012

Outline

Introduction

- This paper addresses the application of Extreme Learning Machines (ELM) to the classification of remote sensing hyperspectral data.
- The proposed process begins with an image segmentation using gradient information computed on a hyperspherical representation of the data.

Introduction

- Feature selection is performed by a greedy wraper approach using classification by conventional ELM and incremental OP-ELM on a Functional Data Analysis (FDA) characterization of the spectral data.
- Conventional ELM obtains results improving other state of the art algorithms with reduced number of features.
- OP-ELM is able to find competitive results using FDA features of a single band .

Hyperspectral Images

- Current hyperspectral cameras have a sensitivity ranging from $100nm$ to $2400nm$, covering also part of the infrared spectrum.
- Fig. 1 illustrates the structure of a hyperspectral image.

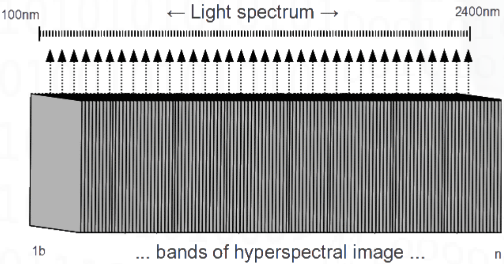
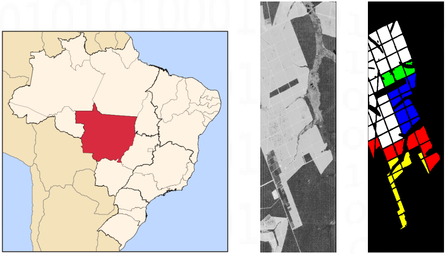


Figure: Schema of hyperspectral image

Soybean Remote Sensing Data

In this paper we focus on a Brazilian soybean crop area. The study area is the Tanguro farm located in Querência municipality, Mato Grosso state, central Brazil (Fig. 2a).



(a) (b) (c)

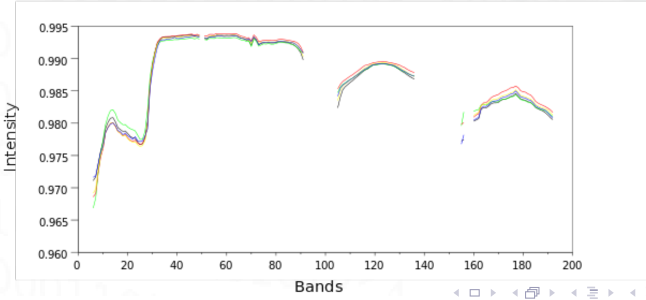
Figure: Location of the Tanguro farm in Brazil. (a) the Mato Grosso State localization, (b) intensity image corresponding to one band, and (c) the ground truth, where soybean class are identified by colors.

Soybean Remote Sensing Data

- Five soybean varieties were planted at this farm in the 2004 to 2005 growing season, covering approximately 8500 ha:
- Perdiz, Monsoy 8411, Monsoy 9010, Kaiabi, and Tabarana.
- The hyperspectral data was obtained by the Hyperion sensor. After radiometrical calibration 196 bands were useful for further analysis.

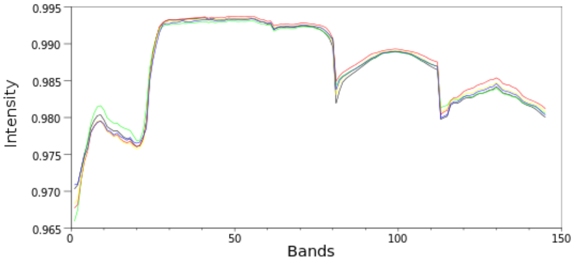
Soybean Remote Sensing Data

- Fig. 4(a) shows the spectral signatures of a sample pixel from each crop class in the normalized intensity range [0-1] after calibrarion, the abcissa corresponds to the actual spectral band number.
- Differences between signature representatives of soybean classes are very small along the 196 bands. Bands with low signal to noise ratio have been removed at the calibration step.



Soybean Remote Sensing Data

- Fig. 4(b) shows the compacted spectral plots, obtained removing the bands without signal.
- This second representation has 145 bands, with five continuous sections: 1 - 44, 45 - 61, 62 - 80, 81 - 112, 113 - 145.



(b)

Figure: Spectral signatures of soybean varieties

Data structures

There are different number of pixel clases on the image:

1. Perdiz : 27614 samples (white)
2. Monsoy 8411 : 3495 samples (green)
3. Monsoy 9010 : 8451 samples (blue)
4. Kaiabi : 9651 samples (red)
5. Tabarana : 5634 samples (yellow)

- For training, we select 300 pixels for each class.

Data structures

Summarizing, we have the following data structures:

$TS_{S \times B}$	Matrix containing all soybean spectra samples (Test set)
$lts_{S \times 1}$	Vector containing the true class of each sample in TS
$cts_{S \times 2}$	Vector containing the domain coordinates (x, y) of each sample
$TR_{S' \times B}$	Matrix containing S' samples (Train set)
$ltr_{S' \times 1}$	Vector with the true class of each sample in TR

where B(number of bands), S(number of soybean pixels) ans S' = 1500 (true class of training set)

Functional Data Analysis

- Functional data analysis (FDA) is an extension of traditional data analysis to functional data; that is, observations that can be thought as real-valued curves over some domain rather than vectors in a high-dimensional space.
- The use of the derivatives is a easy way to apply FDA. In this work we have used the second derivative of the spectral signature in spite of original signal.

Extreme Learning Machines

- Extreme Learning Machine (ELM) is a simple learning algorithm for Sigle-hidden Layer Feedforward Neural network (SLFN).
- This method is based on the Moore-Penrose generalized inverse providing the minimum Least-Squares solution of general linear systems.

Basic ELM

- For N arbitrary distinct samples (x_i, t_i), where input variables are $x_i = [x_{i1}, x_{i2}, \cdots, x_{in}]^T \in \mathbb{R}^n$ and target values are $t_i = [t_{i1}, t_{i2}, \cdots, t_{im}]^T \in \mathbb{R}^m$.

Basic ELM

- A standard ELM with N hidden neurons and activation function $g(x)$ is mathematically modeled as:

$$\sum_{i=1}^{hn} \beta_i \cdot g(w_i \cdot x_j + b_i) = t_j,\ j = 1, \cdots, N. \tag{1}$$

where $w_i = [w_{i1}, w_{i2}, \cdots, w_{in}]^T$ is the weight vector connecting the i-th hidden neuron and the input neurons, $\beta_i = [\beta_{i1}, \beta_{i2}, \cdots, \beta_{im}]^T$ is the weight vector connecting the i-th hidden neuron and the output neurons, and b_i is the threshold of the ith hidden neuron.

Basic ELM

- $w_i \cdot x_j$ denotes the inner product of w_i and x_j and hn is the number of hidden neurons. The activation function can be the identity for the so-called linear kernel approaches, sigmoid for the Multilayer Perceptron approaches, or Gaussian for Radial Basis Function approaches .

Basic ELM

- The equation (1) can be written in matrix form as:

$$\mathbf{H}\beta = \mathbf{T}, \tag{2}$$

  where $\mathbf{H}$, of size $N \times hn$, is the ouput matrix resulting of the SLFN hidden layer activated by the input samples, β is the output weight matrix of size $hn \times N$, and $\mathbf{T}$ is the target matrix with size $N \times m$.
- Training of SLFN is accomplished by computing the least-squares solution $\hat{\beta}$ of the linear system $\mathbf{H}\beta = \mathbf{T}$, given by $\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T}$, where $\mathbf{H}^{\dagger}$ is the Moore-Penrose inverse of $\mathbf{H}$.

Summarized in Algorithm

Given a training set $\aleph = (x_i, t_i) | x_i \in \mathbb{R}^n, t_i \in \mathbb{R}^m, i = 1, \cdots, N,$ activation function $g(x)$, and hn hidden neuron number N ,

1. Assign arbitrary input weight w_i and bias $b_i, i = 1, \cdots, N$.
2. Generate the random hidden layer weight matrix $\mathbf{H}$.
3. Calculate the output weights $\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T}$

Optimally pruned ELM

- Optimally pruned ELM (OP-ELM) is a variation of the ELM introducing some optimal selection of the number of hidden units and the variables modeling the problem, aternative to other incremental approaches to ELM model selection.
- The OP-ELM works in three steps:
    1. Building an overparameterized ELM.
    2. Ranking of the hidden layer neurons by their contribution to the linear explanation of the ELM output by the Multiresponse Sparse Regression (MRSR).
    3. Leave one out (LOO) validation.

Methodology

Summarizing, the process followed to select the best bands for ELM and OP-ELM is:

- First step: Build separate ELM classifiers on the features corresponding to each band, performing separate validations on the test set. The test accuracy is assumed as the saliency of the band.
- Second step: Sort bands by their saliency, denoted as BB. Afterwards, proceed by an incremental training of ELMs orderly adding band features to the training and test sets according to their saliency. As many ELM as bands are trained and tested.
- Third step: Select the ELM with the best testing accuracy.

Methodology

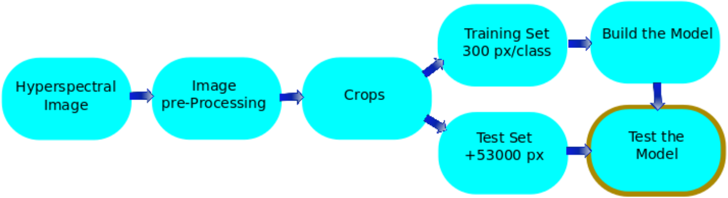


Figure: Diagram flow

Experimental Results

- First, we report the experimental results carried out with with ELM.
- Here we analyze the accuracy versus the dimensionality reducction an their consequences on each class predicion.
- Afterwards, we show the experimental results by using OP-ELM.
- In this experiment, we look for the band with best accuracy comparing the OP-ELM against other well-know algorithms.

ELM Experiment

First step: Independent ELM accuracy on each band
Fig. 6 shows experimental results, red line is the accuracy on training data whereas blue line is the accuracy on testing data, *per* band.

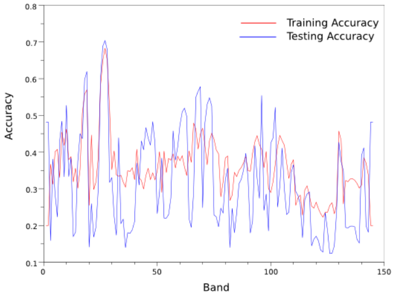


Figure: Accuracy obtained training by independent ELM on each band features.

ELM Experiment

Second step: Incremental ELM training adding bands
Here we sort the bands according to their saliency, building images and ELM in a greedy growing process.

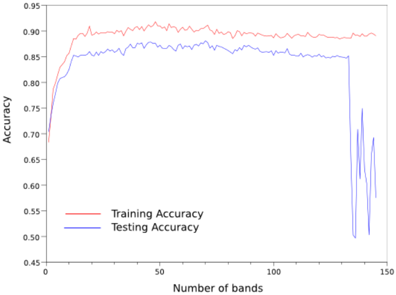


Figure: Accuracy of incremental images of the second derivative

ELM Experiment

Third step: selecting the best ELM

Table 1 summarizes the results of Fig. 7. First column shows the number of bands included in the training/test data, second column shows the accuracy on testing, third column shows the mean absolute error (MAE) and last one shows the Kappa coefficient.

#bands	Accuracy on testing	MAE	Kappa
12	0.855557	0.144443	0.791923
37	0.865093	0.134907	0.806339
46	0.878549	0.121451	0.824533
70	0.882432	0.117568	0.830018

Table: Summary of ELM statistical results

ELM Experiment

Fig. 8 shows the thematic maps produced by the trained ELMs. From left to right (a) to (d) shows the maps obtained using the 12, 37, 46 and 70 most salient bands, respectively.

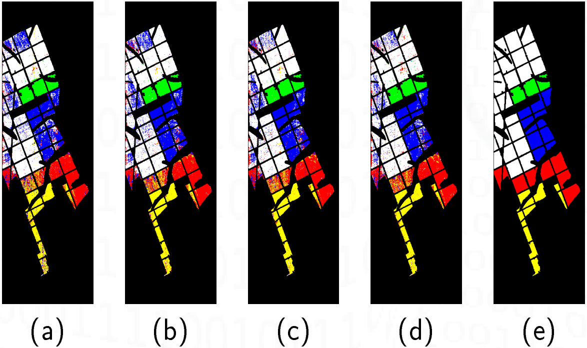


Figure: Predictions for different sizes

OP-ELM Experiments

- In order to find the best band, we train a OP-ELM on each band feature data looking for the best testing accuracy. Fig.9 shows testing accuracy results. The best testing accuracy is found on the band 25.

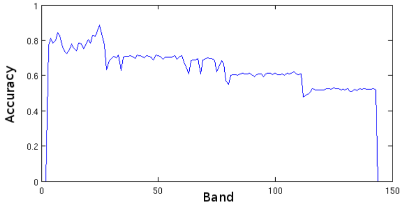


Figure: Testing accuracy obtained by independent OP-ELM trained on each band features.

OP-ELM Experiments

- In order to compare OP-ELM results with well-known classifiers, we have used the WEKA platform, where we have used: 1) a Bayesian network (BayesNet), 2) Support Vector Machine (SVM) , 3) 1-NN, and the decision tree C 4.5 (J48).
- These classifiers are used with their default weka values.

OP-ELM Experiments

Table 2(a) shows the results on training data, while Table 2(b) shows the results on testing data.

Classifier	A	K	MAE
BayesNet	0.676	0.595	0.163
SVM	0.701	0.626	0.259
K-NN	1	1	0.001
C 4.5	0.234	0.043	0.308
OP-ELM	0.876	0.845	0.124

(a)

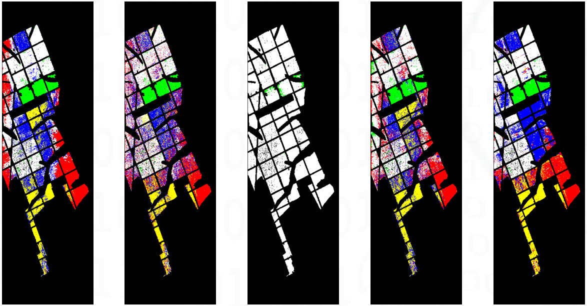
Classifier	A	K	MAE
BayesNet	0.636	0.494	0.192
SVM	0.675	0.547	0.259
K-NN	0.679	0.555	0.128
C 4.5	0.506	0.026	0.314
OP-ELM	0.832	0.762	0.167

(b)

Table: Statistics on training(a), Statistics on testing(b)

OP-ELM Experiments

Fig.10 shows the thematic map produced by each classifier. From left to right, (a) SVM, (b) 1-NN, (c) C4.5, (d) BayesNet and (e) ELM.



(a) (b) (c) (d) (e)

Figure: Pixel-wise class estimation by using only the data in band 25 and its first and second derivatives: (a) SVM, (b) 1-NN, (c) C4.5, (d) BayesNet, (e) OP-ELM.

Conclusions

- In this work we have applied ELM and OP-ELM to Soybean classification in hyperspectral images following a FDA approach for feature extraction, the functional approximation consists of the two first spectral derivatives.
- Soybean classification has proven tough to achieve, and the reported results improve over previous results on the same image data.
- The ELM has been tested in an incremental approach adding salient bands to the data, and training a new ELM at each incremental step.

Conclusions

- Best results with 70 bands improve significatively previous results.
- OP-ELM improves state-of-the art results with only one band information.
- As further works we will study other FDA strategies like splines or polynomial approaches of the hyperpixel spectra.

Thanks for your attention!