Hyperspectral Image Classification using Supervised Classifiers

Shriya T.P. Gupta
Computer Science & Information Systems
BITS Pilani, Goa Campus
Goa, India
shriyatp99@gmail.com

***Abstract*—Hyperspectral image processing is becoming an active topic in remote sensing and other applications in current times. Hyperspectral images with a very narrow spectrum band, can easily distinguish materials which are spectrally similar. This paper explores supervised image classifiers like K- Nearest Neighbors, Parallelepiped and the Multi-Layer Perceptron classifier. These classifiers are trained based on intensities of the pixels as feature. In the testing phase, the sections of the image will be classified into corresponding geographical regions. The accuracy of these classifiers for images from different datasets are discussed to acquire a comparative view. Finally, a graphical user interface has been created for the ease of utility.**

***Index Terms*—Hyperspectral, Remote sensing, Classification, Supervised**

I. Introduction

Hyperspectral image of a geographical area captures the spectrum of solar radiation reflected by the earth's surface. These images have many applications in meteorology, agriculture, geology, forestry, landscape, biodiversity conservation, regional planning, education, intelligence and warfare. Hyperspectral image consists of 100 or more narrow bands and spatial variability of the spectral signature [4].

The proposed system deals with the classification of hyperspectral image efficiently. Several methods of image classification exist and a number of fields apart from remote sensing like image analysis and pattern recognition make use of a significant concept, classification. In some cases, the classification itself may form the entity of the analysis and serve as the ultimate product. In other cases, the classification can serve only as an intermediate step in more intricate analyses. Moreover, the selection of the appropriate classification technique to be employed can have a considerable upshot on the results of whether the classification is used as an ultimate product or as one of numerous analytical procedures applied for deriving information from an image for additional analyses. The information contained in hyperspectral image allows the characterization, identification, and classification of the land covers with improved accuracy and robustness [2].

The three classifiers used here are, K-Nearest Neighbor (KNN), Parallelepiped and Multilayer Perceptron (MLP) and their accuracy has been assessed for different parameters and varied images.

II. Dataset

INDIAN PINES: This scene was gathered by AVIRIS sensor over the Indian Pines test site in North-western Indiana and consists of 145x145 pixels and 224 spectral reflectance bands. This scene is a subset of a larger one. The Indian Pines scene contains two-thirds agriculture, and one-third forest or other natural perennial vegetation. The ground truth available is labeled into sixteen classes which are not mutually exclusive. We have also reduced the number of bands to 200 by removing bands covering the region of water absorption.

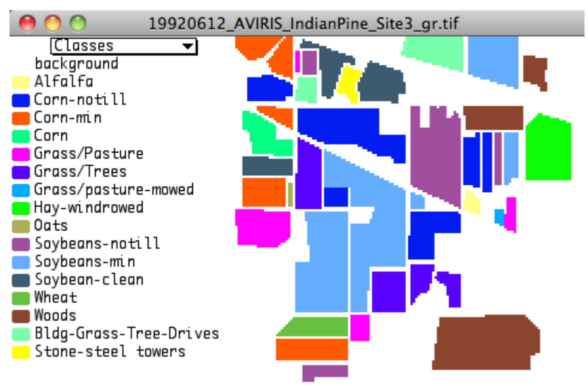


Fig. 1: Indian Pines

SALINAS A: This scene was collected by the 224 band AVIRIS sensor over Salinas Valley, California, and is characterized by high spatial resolution of 3.7-meter pixels. A small sub-scene of Salinas image, denoted Salinas A, is used as data. It comprises 86x83 pixels and includes six classes.

III. K-Nearest Neighbor (KNN)

The KNN algorithm belongs to the family of instance-based, competitive and lazy learning algorithms. Instance-based algorithms are those algorithms that model the problem using data instances (or rows) in order to make predictive decisions. The KNN algorithm is an extreme form of instance-based method because all training observations are retained as part of the model. It is a competitive learning algorithm, because it internally uses competition between model elements (data instances) in order to make a predictive decision.

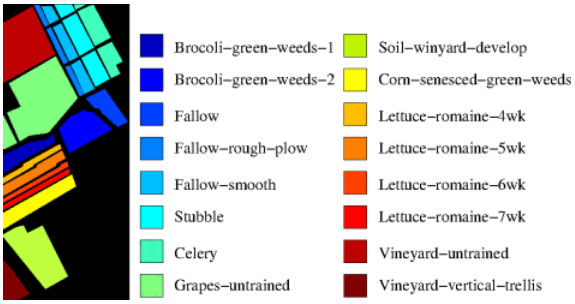
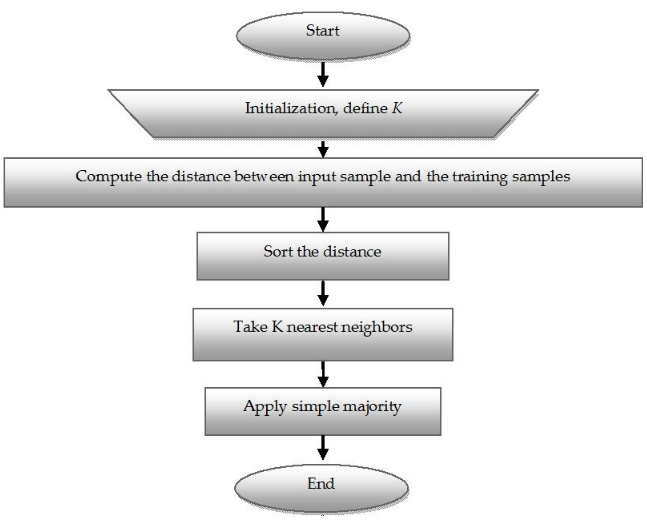


Fig. 2: Salinas

The objective similarity measure between data instances causes each data instance to compete to be most similar to a given unseen data instance and contribute to a prediction. Lazy learning refers to the fact that the algorithm does not build a model until the time that a prediction is required. This has the benefit of only including data relevant to the unseen data, called a localized model. A disadvantage is that it can be computationally expensive to repeat the same or similar searches over larger training datasets.

Finally, KNN is powerful because it does not assume anything about the data, other than a euclidean distance measure which can be calculated consistently between any two instances. As such, it is called non-parametric or non-linear as it does not assume a functional form. [5]

A. Methodology



B. Performance Evaluation

First six entries of the table correspond to the classifier created using Numpy and GDAL libraries of python (first three for indian pines and next three for Salinas A).The last three entries correspond to the KNN classifier of the scikit learn library for Indian pines data[3]. The classifier outputs are shown in plots 1 to 9.

Image	train:test	Accuracy(%)	Time(s)	Kvalue	Plot
Pines1	70:30	85.11	2211.5	7	1
Pines1	70:30	85.05	48100	5	2
Pines1	80:20	85.85	549.67	5	3
SalinasA	70:30	91.20	89.689	7	4
SalinasA	70:30	90.61	87.923	5	5
SalinasA	80:20	93.88	71.902	5	6
Pines2	70:30	84.63	9.888	7	7
Pines2	70:30	84.55	5.101	5	8
Pines2	80:20	84.56	6.295	5	9

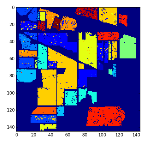


Fig. 3: Plot 1

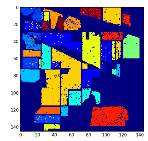


Fig. 4: Plot 2

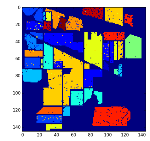


Fig. 5: Plot 3

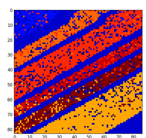


Fig. 6: Plot 4

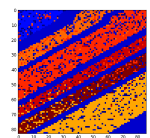


Fig. 7: Plot 5

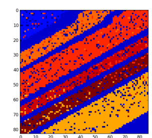


Fig. 8: Plot 6

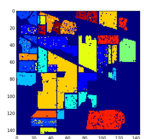


Fig. 9: Plot 7

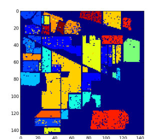


Fig. 10: Plot 8

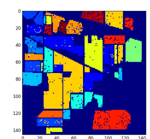


Fig. 11: Plot 9

IV. Parallelepiped Classifier

Parallelepiped classification uses a simple decision rule to classify data. The decision boundaries form an n-dimensional parallelepiped in the image data space. The dimensions of the parallelepiped are defined based upon a standard deviation threshold from the mean of each selected class. If a pixel value lies above the lower threshold and below the upper threshold for all n bands being classified, it is assigned to that class. If the pixel value falls in multiple classes, the pixel is assigned to the first class matched. Areas that do not fall within any of the parallelepiped classes are labeled as unclassified. This method is quick to run, but not very accurate as the parallelepipeds are formed based on their maximum and minimum pixel values that may not be representative of a class.

A. Performance Evaluation

The parameter varied here is the upper and lower limit of the parallelepiped which is defined as mean plus standard deviation and mean minus standard deviation respectively. All calculations are for aviris image.The classifier outputs are shown in plots 1 to 3.

Limits	Accuracy(%)	Time(s)	Output
1 std dev	56.147	34.042	plot1
2 std dev	52.199	36.064	plot2
3 std dev	51.414	36.909	plot3

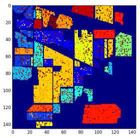


Fig. 12: Plot 1

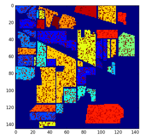


Fig. 13: Plot 2

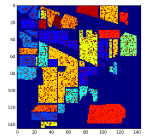


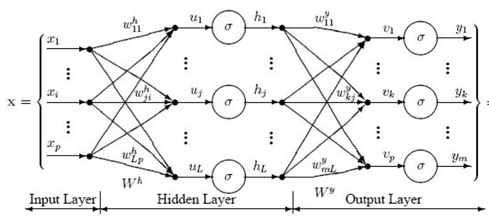
Fig. 14: Plot 3

V. Multi Layer Perceptron (MLP)

Multi Layer Perceptron utilizes a supervised learning algorithm for training the network. Its multiple layers and non-linear activation function distinguishes MLP from a linear perceptron and hence it can distinguish data that is not linearly separable. Initially the input signals are forward propagated through the various layers of the network. Then learning occurs in the perceptron by changing connection weights after each piece of data is processed, using a back propagation algorithm. Back propagation is a method which calculates the gradient of the loss function with respect to the weights in an artificial neural network. It is commonly used as part of algorithms that optimize the performance of the network by adjusting the weights based on the amount of error in the output compared to the expected result, for example in the gradient descent algorithm. This project uses a network with a single hidden layer with ten neurons and finds the accuracy of the classification for a fixed number of epochs. [1]

A. Structure

This is the structure of a multi layer perceptron with a single hidden layer which is usually known as a vanilla neural network.



B. Performance Evaluation

First six entries of the table correspond to the classifier created using Numpy and GDAL libraries of python (first three for indian pines and next three for Salinas A).The last three observations correspond to the MLP classifier of the scikit learn library for aviris indian pines data. The classifier outputs are shown in plots 1 to 9.

VI. Conclusion

This project attempts to study and provide a brief knowledge about the different image classification approaches to classify hyperspectral images using supervised classifiers. It compares the performance of these classifiers for different images with varying parameters. Future work includes improving the accuracy of the already implemented classifiers and exploring the neural network classifiers further.

Image	Split	Accuracy(%)	Time(s)	Epoch	Plot
Pines	70:30	63.04	2285.9	100	1
Pines	70:30	62.98	15,492	200	2
Pines	80:20	50.98	4920.5	100	3
SalinasA	70:30	54.48	966.99	100	4
SalinasA	70:30	56.02	1779.5	200	5
SalinasA	80:20	41.94	1139.8	100	6
sklearn	70:30	86.16	14.45	100	7
sklearn	70:30	89.09	15.155	200	8
sklearn	80:20	88.41	14.233	100	9

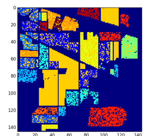


Fig. 15: Plot 1

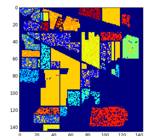


Fig. 16: Plot 2

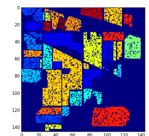


Fig. 17: Plot 3

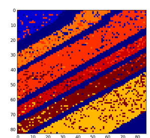


Fig. 18: Plot 4

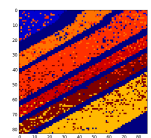


Fig. 19: Plot 5

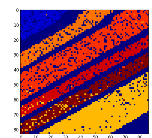


Fig. 20: Plot 6

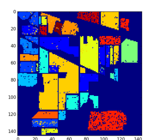


Fig. 21: Plot 7

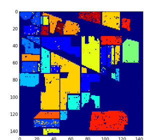


Fig. 22: Plot 8

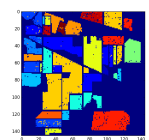


Fig. 23: Plot 9

VII. Acknowledgements

The author would like to thank Prof. B.K.Mohan, H.O.D, CSRE Dept, IIT-Bombay for the valuable guidance and Ms. Arti Tiwari, Research Scholar, CSRE Dept, IIT-Bombay for the fruitful discussions.

References

[1] Simon Haykin. Introduction. In *Nonlinear Methods of Spectral Analysis*, pages 1–8. Springer, 1979.

[2] Biao Hou, Xiangrong Zhang, Qiang Ye, and Yaoguo Zheng. A novel method for hyperspectral image classification based on laplacian eigenmap pixels distribution-flow. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6(3):1602–1618, 2013.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[4] Yuliya Tarabalka, Jón Atli Benediktsson, and Jocelyn Chanussot. Spectral–spatial classification of hyperspectral imagery based on partitional clustering techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 47(8):2973–2987, 2009.

[5] Sergios Theodoridis, Aggelos Pikrakis, Konstantinos Koutroumbas, and Dionisis Cavouras. *Introduction to pattern recognition: a matlab approach*. Academic Press, 2010.