1.5em

EM525 Bayesian Techniques for Pixel Classifiers in Higher Dimensional Imaging Techniques

Submitted by : Gihan Jayatilaka E/14/158
gihanjayatilaka@eng.pdn.ac.lk

***Abstract*—Pixel classification in aerial photographs is a fundamental task of aerial sensing techniques. The classification algorithms span over supervised, unsupervised and semi supervised techniques. This paper discusses the use of naive Bayes technique for the task as a supervised learning algorithm. The classification performance is analyzed for different variants of the algorithm. Finally, a novel semi supervised technique is proposed and evaluated for the same data set.**

I. Introduction

Aerial sensing is a broad field of capturing information about the land through aircraft, baloon, UAV or sattelite photography. Aerial photography is a sub field where spatio-spectral data in obtained (Photography is basically recording the sampled bands of the electromagnetic spectrum). The photographs vary in both spectral and spatial resolution. Higher spatial resolution is always desirable for better interpretation of the photographs. This is obtained by high resolution cameras placed closer to the earth surface.
Spectral resolution depends on the sensor elements on the camera. Higher spectral resolution can be both advantageous and disadvantageous. While a higher number of spectral bands in an photograph contains more information, it could be hard to visualize and interpret.
This paper proposes a class of algorithms to interpret aerial photographs of very high spectral resolution and average spatial resolution. It discusses the challenges and evaluates the solutions.

A. Hyper spectral photography

Monochromatic images have one spectral sampling and RGB photos have three spectral bands – red (around 700 nm), greed (around 550nm) and blue (around 450nm). Hyperspectral imagery refers to the techniques where a very high number of spectral bands are sampled. Usually the number os bands is more than 100 and the frequencies of the bands range from 450nm to 2400nm.
Hyperspectral imagery is used in survaillance, chemical imaging, bio-medical applications and aerial sensing. The aerial sensing applications of hyperspectral imaging span across agriculture, mineralogy and mapping.

B. Problem

Labelling the pixels in an aerial photograph to generate a labelled land cover map is the objective of this project.

II. Available Solutions

A. Supervised learning algorithms

Supervised learning algorithms such as support vector machines, decision trees and neural networks have been used for pixel classification in land cover maps extensively[1]. They show better performance if the training data has the same conditions as the testing data. Common problems seen are related to over fitting which results in inferior performance on data obtained under different conditions.

B. Unsupervised learning algorithms

Unsupervised learning algorithms try to see the patterns in the data itself. Even though classification itself is not an unsupervised learning task, it could be achieved by unsupervised clustering paired with a technique to attach the label.

Unsupervised learning is better at handling overfitting related problems. But they require more huam effort and domain expertise in the filed of hyperspectral imaging and land cover maps to get acceptable results.

C. Semi-supervised learning algorithms

Semi supervised learning algorithms[2] tries to exploit advantages in both approaches to get better results. These algorithms make it possible to incorporate the domain expertise with learning from data. This paper proposes a semi supervised approach to the problem.

III. Proposed Solution

This paper proposes a solution based on Naive Bayes classifier (NB) for labelling the pixels in a land cover map. First the standars NB algorithm is applied on the dataset as supervised learning. Then a new variant of the algorithm is proposed as a semi-supervised learning approach to the problem.

A. Abstraction

The datasets consists of hyper spectral images and the labels corresponding to the pixels. Consider a case where a pixel has N spectral bands. Let $X_{(N\times 1)}$ be a vector with variables $X = (X_1, X_2, ..., X_N)$ corresponding to the intensity of each spectral wave band. Since continuous variables are difficult to handle in NB approach, the continuous probability distribution is discretized such that X_i becomes a discrete

random variable as,

$$X_i = \begin{cases} \mathrm{x}_{i,1}, & \mathrm{X}_i < s_1 \\ \mathrm{x}_{i,2}, & \mathrm{s}_1 <= X_i < s_2 \\ \mathrm{x}_{i,M}, & \mathrm{s}_{M-1} <= X_i < s_M \end{cases} \quad (1)$$

The labels of the pixels are words corresponding to what the pixel represents such as land, roof, water. This is abstracted to a discrete variable Y as

$$Y = \begin{cases} \mathrm{y}_1, & Label01 \\ \mathrm{y}_2, & Label02 \\ \mathrm{y}_k, & LabelK \end{cases} \quad (2)$$

The problem is identifying the label of a pixel – $f(X)$ given its spectral information. This can be expressed in a formulae as,

$$f(X) = \text{argmax } p(Y = y_i|X) \quad (3)$$

This is a conditional probability problem. The most probable label given the spectral information should be calculated. The proposed algorithms are based on the assumption that the individual spectral information S_i is independent of other spectral information $X_j; j$ given the label Y,

$$p(X_i, X_j|Y) = p(X_i|Y) \times p(X_j|Y) \quad (4)$$

B. Splitting the dataset

The dataset (both pixel information and their corresponding ground truth labels) s split into two parts as the training dataset and the test dataset. Only the training dataset if used for the building up the bayesian model for pixel classification. The test dataset is used for performance analysis of the algorithms.
All testing was done with a 50:50 split for this project.

C. Naive Bayes algorithm

Naive Bayes algorithm[3] require the calculation of likelihood $p(X_i|Y)$ from the dataset using a statistical analysis,

$$p(X_i = x_{i,j}|Y = y_k) = \frac{\text{No of } y_k, x_{i,j} \text{ pixels}}{\text{Total number of } y_k \text{ pixels}} \quad (5)$$

These $p(X_i|Y)$ values are used in determining the label of a test set pixel by equations (6), (3)

$$p(Y|X = (x_1, ..., x_N)) = p(Y)p(x_1|Y) \times ... \times p(x_N|Y) \quad (6)$$

The label can be interpreted easily by the inverse of (2). The naive Bayes algorithm have many variants depending on how the discretization (1) happens. The standard possibilities are Bernoulli naive Bayes, multinomial naive Bayes and Gaussian naive Bayes. This paper also purposes a two new techniques as improvements for Gaussian Naive Bayes.

D. Bernoulli Naive Bayes

This algorithm interprets $X = (X_1, X_2, .., X_N)$ as a multi variate Bernulli distribution. Therefor every X_i is discritized into two levels as per (7) in place of (1).

$$semi - supervised X_i = \begin{cases} \mathrm{T}_i, \\ \mathrm{F}_i, \end{cases} \quad (7)$$

E. Multinomial Naive Bayes

This algorithm interprets the dataset as a multinomial distribution. This replaces (5) with (8)

$$p(X|Y = y_k) = \frac{(\sum_i x_i)!}{\prod_i x_i!} \prod_i p_{ki}^{x_i} \quad (8)$$

where p_{ki} is the parameter defining the i^{th} multinomial distribution for label k.

F. Gaussian Naive Bayes

This algorithm is usually good for random processes since it assumes a normal distribution of features (vector elements) for a given label. The algorithm first calculates the mean and variance of each element X_i for a given label Y_k.

$$\mu_{X_i}|Y = \frac{\sum x_i}{\text{count}|Y} \qquad \sigma_{X_i}|Y = \frac{\sum (x_i - \mu_{X_i})^2}{\text{count}|Y} \quad (9)$$

Then these parameters are used for X_i discritization in place of (1) as (10) substituting from (11)

$$X_i = x_{i,j} \text{ if } j = \text{argmax}_k \; p(X_i = x_{i,k}|Y = y_i) \quad (10)$$

where,

$$p(X_i = x_{i,k}|Y = y_i) = \frac{1}{\sqrt{2\pi\sigma_{y_i}}} \exp \frac{-(x_{i,k} - \mu_{y_i})^2}{2\sigma_{y_i}} \quad (11)$$

G. Modified Gaussian Naive Bayes (Assuming feature independence)

The Gaussian naive Bayes algorithm[4] assumes that the probability distribution of a feature is a single Gaussian distribution. This assumption does not hold in every case because a certain label can be a blanket term for a set of spectral different pixels. Consider the label ”vegetation” or ”trees”. This may have trees of green colour as well as trees of some other colours. Trying to put them all under a single Gaussian distribution is problematic.
Therefor, we introduce an unsupervised step for clustering the distribution of $X_i|Y$ first and then using (11) on these clusters separately. This approach exploits the advantage of seeing patterns in data which the traditional Gaussian Naive Bayes would not see.
Please note that all clustering happens with standard kmeans algorithm[5].

H. Modified Gaussian Naive Bayes (Assuming feature correlation)

All the previous approaches assume the independence of individual features once the label is given. This is why (6) or its variants can be applied to every X_i separately. But in a real world application, the feature elements are correlated. The correlation of X_i can be calculated by (12)

$$\rho_{X_i,X_j} = \frac{\text{covariance}(X_i, X_j)}{\sigma_{X_i}\sigma_{X_j}} \tag{12}$$

Algorithm 1 Modified NB

1: **procedure** CORRELATION FULL DATASET
2: Calculate correlation of each X_i from (12)
3: Cluster correlated features together $Z_i = (X_a, X_b, X_c)$
4: Perform Modified Gaussian Naive Bayes assuming the independence of $Z_i|Y_k$
5: **procedure** CORRELATION OF INDIVIDUAL LABELS
6: Seperate X according to Y Calculate correlation of each $X_i|Y$ from (12)
7: Cluster correlated features together $Z_i|Y = (X_a|Y, X_b|Y, X_c|Y)$
8: Perform Modified Gaussian Naive Bayes assuming the independence of $Z_i|Y_k$

Please note that all clustering happens with kmeans algorithm given in appendix.

I. Performance measures

The performance of each algorithm is tested by the confusion matrix and the accuracy. The accuracy itself is not a perfect measure in this problem since the label counts are not balanced in most of the datasets.

IV. Implementation and Testing

A. Implementation

The algorithms were implemented and evaluated using the following technologies[6]

Technology	Purpose
Python 3	Scripting
Numpy	Numerical computation
Sklearn	Standard algorithms
Matplotlib	Visualization

TABLE I
TECHNOLOGIES USED

B. Testing

The algorithms were tested on an open dataset – **Pavia University**. This dataset is captured by an airbone ROSIS (reflective optics system imaging spectrometer)). The dataset is a $610px \times 610px$ photograph of $1.3m^2$ spatial resolution and 103 bands of spectral resolution. The unusable pixels were removed in the preprocessig step. The dataset contains 9 labels and the datapoints per pixel is not balanced.

#	Label	No of pixels
1	Water	824
2	Trees	820
3	Asphalt	816
4	Self-Blocking Bricks	808
5	Bitumen	808
6	Tiles	1260
7	Shadows	476
8	Meadows	824
9	Bare Soil	820

TABLE II
PAVIA UNIVERSITY DATASET

The dataset's groundtruth (colour coded labels) are illustrated in figure 1 and the distribution of a single feature (a single dimension of vector X is given in figure 2.

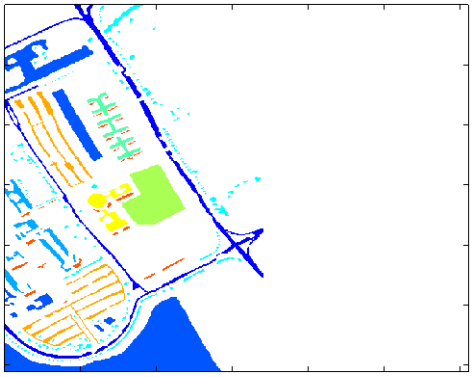

Fig. 1. Dataset Groundtruth

Fig. 2. Distribution of a single feature

C. Results

Running the algorithms on Pavia University Dataset gave the accuracies given in table III and figure 3.

#	Algorithm	Accuracy
1	Bernoulli NB	48.06
2	Multinomial NB	50.04
3	Gaussian NB	61.45
4	Modified Gaussian NB (feature independence)	65.58
5	Modified Gaussian NB (feature correlation full dataset)	53.66
6	Modified Gaussian NB (feature correlation individual label)	70.25

TABLE III
THE VARIATION OF ACCURACY WITH ALGORITHM

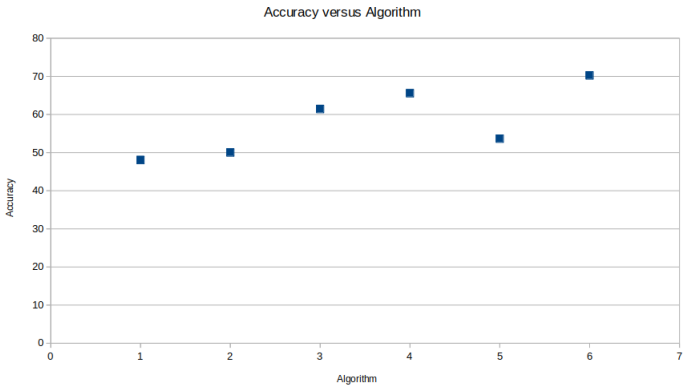

Fig. 3. The variation of accuracy with algorithm

Testing the same dataset against other general algorithms gave the accuracies given in IV. SVM is a widely used linear classifier. Neural netowrks are a widely used non linear classifier. The Neural Network architecture was obtained from previous work

#	Algorithm	Accuracy
1	Support Vector Machines	60
2	Neural Networks	80

TABLE IV
THE VARIATION OF ACCURACY FOR OTHER ALGORITHM

V. Conclusion

A. Standard Naive Bayes Algorithms

The Bernoulli and Miltinomial NB algorithms show very weak results. This is interpretable as the disparity between the assumptions made and the actual dataset. The spectral intensity distributions are neither Berunoulli nor Multinomial.

The higher performance of Gaussian NB is expected given that all the natural processes have a normal distribution like nature (since the emisssion of electro magnetic waves is a random process in its core). However it is evident that further improvement is necessary for Gaussian NB.

B. Proposed Naive Bayes Based Algorithms

The modified Gaussian NB outperforms Gaussian NB due to the higher number of discretizations levels of the feature distributions. This is evidence for the presence of multiple clusters in the distributions. This is expected in the presence of labels such as bare soil and shadows which can take multiple colours (therefor multiple spectral signature clusters).

The assumption of feature correlation is a dimensionality reduction step. This could be understood as reducing $X_{(1\times N)} \leftarrow Z_{1\times M}$ where $N > M$. Usually the NB algorithms only try to reduce the possible values of the random variable's individual feature by the process of discritization. But the dimensionality reduction step reduces the number of features (by turning a large number of features into a smaller number of meaningful features which are easier for interpretation).

The inferior performance of Modified Gaussian NB assuming feature correlations in the whole dataset could be interpreted as the full dataset being very diverse and therefor the correlation patterns may not be general to the whole dataset.

The Modified Gaussian NB assuming feature correlation within a label performs better than all other algorithms. This could be interpreted as the assumption holding for the dataset. Every label has some dominant features that is visible through multiple correlated individual features. Finding these correlation and reducing them to individual meaningful features has increased the accuracy of the algorithm.

C. Other algorithms

The proposed algorithms outperforms SVM linear classifier but falls behind the non linear neural network based classifiers. Since Neural Network's require higher amount of computation power the proposed algorithm is suitable for low computation environments. However the proposed algorithm is not in par with modern non linear classifiers considering the accuracy measures.

VI. Future Work

Hyper parameter tuning is bound to give better accuracy results. The hyperparameters of these algorithms are

1) Number of clusters per a feature's distribution
2) The threshold of correlation for clustering

Analyzing the correlation patterns between the individual features can give better insights to the actual behaviour of hyper spectral emission of material.

VII. Acknowledgement

This work is closely based on a project supervised by Dr. Roshan Godaliyadda, Dr. Vijitha Herath and Dr. Parakrama Ekanayeka from Department of Electrical and Electronics Engineering, Faculty of Engineering, University of Peradeniya. The datasets used were released for public use by University of Pavia.

References

[1] G. P. Petropoulos, K. Arvanitis, and N. Sigrimis, "Hyperion hyperspectral imagery analysis combined with machine learning classifiers for land use/cover mapping," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3800 – 3809, 2012.

[2] J. Prakash and D. L. M. Nithya, "A survey on semi-supervised learning techniques," *International Journal of Computer Trends and Technology*, vol. 8, 02 2014.

[3] I. Rish, "An empirical study of the naïve bayes classifier," *IJCAI 2001 Work Empir Methods Artif Intell*, vol. 3, 01 2001.

[4] N. Dobigeon, J. Tourneret, and C. Chang, "Semi-supervised linear spectral unmixing using a hierarchical bayesian model for hyperspectral imagery," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2684–2695, July 2008.

[5] Y. Li and H. Wu, "A clustering method based on k-means algorithm," *Physics Procedia*, vol. 25, pp. 1104–1109, 12 2012.

[6] 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*, vol. 12, pp. 2825–2830, 2011.