ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 8(24), DOI: 10.17485/ijst/2015/v8i24/80843, September 2015* *ISSN (Online) : 0974-5645*

GURLS vs LIBSVM: Performance Comparison of Kernel Methods for Hyperspectral Image Classification

Nikhila Haridas*, V. Sowmya and K. P. Soman

Center for Excellence in Computational Engineering and Networking, Amrita Vishwa Vidyapeetham, Coimbatore - 641001, Tamil Nadu, India; nikhila.haridas92@gmail.com, vsowmya@cb.amrita.edu

Abstract

Kernel based methods have emerged as one of the most promising techniques for Hyper Spectral Image classification and has attracted extensive research efforts in recent years. This paper introduces a new kernel based framework for Hyper Spectral Image (HSI) classification using Grand Unified Regularized Least Squares (GURLS) library. The proposed work compares the performance of different kernel methods available in GURLS package with the library for Support Vector Machines namely, LIBSVM. The assessment is based on HSI classification accuracy measures and computation time. The experiment is performed on two standard Hyper Spectral datasets namely, Salinas A and Indian Pines subset captured by AVIRIS (Airborne Visible Infrared Imaging Spectrometer) sensor. From the analysis, it is observed that GURLS library is competitive to LIBSVM in terms of its prediction accuracy whereas computation time seems to favor LIBSVM. The major advantage of GURLS toolbox over LIBSVM is its simplicity, ease of use, automatic parameter selection and fast training and tuning of multi-class classifier. Moreover, GURLS package is provided with an implementation of Random Kitchen Sink algorithm, which can easily handle high dimensional Hyper Spectral Images at much lower computational cost than LIBSVM.

Keywords: Classification, GURLS, Hyper Spectral Image, Kernel Methods, LIBSVM

1. Introduction

Hyper Spectral Imaging or imaging spectroscopy, deals with the collection of information across the earth's surface in hundreds or thousands of contiguous narrow spectral bands spanning the visible and infrared region of the electromagnetic spectrum[1]. The detailed information present in each pixel vector of Hyper Spectral Images (HSI) allows classification of hyper spectral images with improved accuracy and robustness. However, for hyper spectral images supervised land cover classification is a difficult task due to the unbalance between the limited availability of training samples and the high dimensionality of the data (Hughes Phenomenon)[2]. Over the last decade, many machine learning methods have been developed for classification of multispectral and hyper spectral images. Unfortunately, these methods fail to yield adequate results with hyper spectral data as they are sensitive to Hughes Phenomenon[3]. In order to alleviate these problem, several preprocessing or feature extraction techniques have been integrate in processing chain of HSI prior to classification. Even though the inclusion of such preprocessing methods improves the classification accuracy, they are time consuming and scenario dependant[4]. Therefore, the desirable property of optimum hyper spectral data classifiers is that it should have fast prediction time and high classification accuracy[5]. In recent years, kernel based methods[6,7] has gained considerable attention for pattern recognition with hyper spectral data[3] . Kernel based approaches such as SVM[8] have proved to be valid and effective tool for classification tasks in multispectral and hyper spectral images[9]. In particular, authors[10] classified AVIRIS data using Support Vector Machines. Gustavo Camps-Valls and Lorenzo Bruzzone[3] analyzed different kernel based approaches and their properties for HSI classification. The use of composite kernels

*Author for correspondence

for enhanced classification of hyper spectral images is discussed in[4]. Begum Demir and Sarp Erturk[11] utilized Relevance Vector Machines (RVM) based classification approach for low complex applications. Chih-Chung Chang and Chih-Jen Lin developed a software library for Support Vector Machines namely, LIBSVM[12] in order to help users to apply SVM in their applications. Ali Rahimi and Benjamin Recht[13] introduced Random Kitchen Sink algorithm to significantly reduce the computation cost and time needed for training kernel machines. Tacchetti et al.[14] introduced GURLS, a least squares library for supervised learning which can deal with multi class problems for high dimensional data.

This paper utilize a new modular software package namely, Grand Unified Regularized Least Squares (GURLS) for efficient supervised classification of hyper spectral images. The experiment analyses the performance of different kernels in GURLS and compared the performance to LIBSVM. The comparative assessment is based on classification accuracy and computation time. Based on the comparative assessment measures, our experimental study shows that GURLS library is well suited for classification tasks in hyper spectral images as LIBSVM.

Rest of the paper is outlined as follows. Section 2 gives a brief description of kernel based libraries. Methodology of the proposed work is presented in section 3, Section 4 reports the experimental results and analysis and finally, section 5 concludes the paper.

2. Kernel Based Libraries

In this section, we discuss about the two recent open source kernel based software libraries namely, LIBSVM and GURLS. The section gives a brief theoretical description of the two packages, followed by the mathematical concepts involved in it.

2.1 LIBSVM

LIBSVM[12] is the most popular software library for Support Vector Machines (SVM). It is an integrated software package that can be used for machine learning tasks such as SVM classification and regression. The package has been developed since the year 2000. The goal of LIBSVM package is to allow users to explore and experiment with SVM for their purpose. LIBSVM efficiently supports multi-output/multi-class classification problems. The package includes different SVM formulations, various kernels, cross validation approach for model selection, probability estimates and a simple graphical user interface demonstrating SVM classification and regression. LIBSVM supports different SVM types such as One-Class SVM, Regressing SVM and nu SVM. The following subsection describes about the mathematical concepts involved in Support Vector Machines (SVM). For better understanding and clarity, the standard binary formulation of SVM is discussed.

2.1.1 Support Vector Machines (SVM)

Support Vector Machines are the SVM[7] are the most commonly used class of algorithms in machine learning that uses the idea of kernel substitution. It belongs to a set supervised learning methods suitable for data mining tasks such as classification and regression. SVM is based on the concept of maximum margin that uses a linear separator (hyperplane) to separate the data belonging to different classes. While doing so, the algorithm finds out maximum separation between data of different classes.

Let us consider a case of two-class linearly separable data, (x_i, y_i) for i = 1; 2; ....., m with class labels $y_i \in (-1,1)$ where the training set contains m vectors from n dimensional space. For each vector xi there is an associated class label y_i. The hyperplane for classification is represented as,

$$w^T x - \gamma = 0 \qquad w \in R^n \tag{1}$$

That is,

$$w = (w_1, w_2, w_3, \ldots\ldots, w_n)^T \tag{2}$$

The corresponding decision function is given as,

$$f(x) = sign(w^T x - \gamma) \tag{3}$$

Here $x = (x_1, x_2, x_3, \ldots\ldots, x_n)^T$ is n dimensional feature vector in real space R^n. The problem formulation for a two class SVM in its primal form is given by,

$$\begin{aligned} &\min_{w,\gamma} \frac{1}{2} w^T w \\ &D(Aw - \gamma e) \geq e \end{aligned} \tag{4}$$

where A represents m x n data matrix belonging to R_n and D represents m x m diagonal matrix with class labels as the diagonal elements and e represents m x 1 column vector of all ones. Here, w and are the minimization parameters which helps in constructing the hyperplane that divides the data between the two classes. The dual formulation of Equation (1) is

$$\min \frac{1}{2} u^T DA^T Au - e^T u$$
$$e^T Du = 0, u \geq 0 \tag{5}$$

where the elements in u are Lagrangian multipliers and AA^T represents the kernel matrix. The elements of kernel matrix represents the correlation among the data points. The SVM formulation described above works only if data are linearly separable. Consider the case when data is nonlinear. In such cases, the data is projected to a higher dimensional space using a mapping function, Φ.

$$\Phi : \Re^n \rightarrow \Re^k \tag{6}$$

where $\Phi(x) \in \Re^k$ and k >> n. The computation of explicit mapping function, Φ is expensive and unfeasible. Therefore, instead of finding the explicit map, an alternative was proposed which replaces each element of the kernel matrix with a valid kernel function $k(x_i, x_j)$ that obeys the Mercer's theorem[15].

That is,

$$k(x_i, x_j) = \left\langle \Phi(x_i).\Phi(x_j) \right\rangle \tag{7}$$

The formulation of popular kernels[16] available in LIBSVM is given below,

- *Linear Kernel*

$$k(x_i, x_j) = x_i^T x_j \tag{8}$$

- *Polynomial Kernel*

$$k(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0 \tag{9}$$

- *Radial Basis Function Kernel*

$$k(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right), \gamma > 0 \tag{10}$$

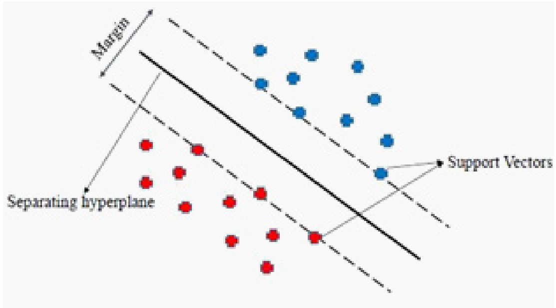
- *Sigmoid Kernel*

$$k(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \tag{11}$$

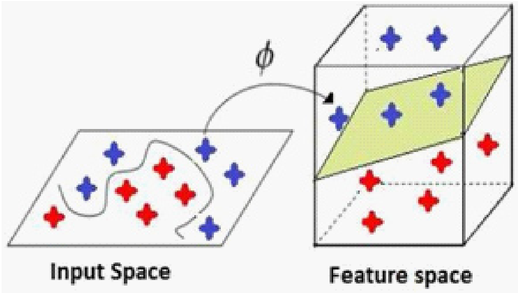
Here γ, r and d are kernel parameters.

2.2 GURLS

Grand Unified Regularized Least Squares[14] commonly known as GURLS, is a recently developed software library for supervised learning. GURLS utilizes Regularized Least Squares (RLS)[17] approach for classification and regression. The library can easily handle multi-class/multi-label problems that are difficult to manage for other linear learning methods. It gives a number of training strategies for solving small, medium and high dimensional learning tasks. The package is currently implemented using MATLAB and C++. GURLS library exploits the use of latest tools in linear algebra for generating solution. The main advantage of GURLS library is that it provides memory mapped storage and distributed task execution which increases the system performance and reduces the computation cost, especially when dealing with large datasets. The library also offers several routines for automatic parameter selection for efficient model selection. The GURLS library was developed to meet the following objectives: 1) Speed: For fast training and output prediction when dealing with high dimensional data 2) Memory: Efficient data management by memory mapped storage 3) Performance: For providing state of art results when compared to conventional methods for multi class classification and 4) Usability and modularity: Simple and easy to use, takes the advantage of RLS for machine learning tasks.



(a)



(b)

Figure 1. SVM classification. **(a)** Linear data. **(b)** Non linear data.

2.2.1 Regularized Least Squares (RLS)

The RLS takes the input data and calculates the weight matrix W required for the classification. Consider a p-class classification of N data points $x = [x_1, x_2, x_3, ..., x_N]$. Each x_i for $i = 1, 2, 3, ..., N$ belongs to $\Re^n$ where n represents the number of bands and let $Y = [y_1, y_2, y_3, ..., y_N]$ be the label vector where $y_i \in \Re^p$. The main objective here is to find a matrix W which can linearly map each data point x_i into a vector y_i, which is unique for each class. Any data x_i will have a corresponding label vector y_i which belongs to one of the p classes. One possible way to fix weight matrix, W is to choose W which minimizes the sum of squares of L2 norm of errors for each of the n equations. This problem can be expressed as,

$$\min_W \sum_i \|WX_i - y_i\|_2^2 \tag{12}$$

This minimization can be equivalently expressed in terms of the Frobenius norm of a matrix as,

$$\min_W \|WX - Y\|_F^2 \tag{13}$$

where X is the $n \times N$ matrix with data along the columns and Y is a $p \times N$ matrix whose i^{th} column gives the class label for the i^{th}column of X. Now, to avoid overfitting a regularization term is added to the minimization expression with λ as the control parameter.

$$\min_W \|WX - Y\|_F^2 + \lambda \|W\|_F^2 \tag{14}$$

This expression is equivalent to

$$\min_W tr((WX - Y)(WX - Y)^T) + tr(\lambda W^T W) \tag{15}$$

which can be solved as,

$$\frac{\partial}{\partial W}(tr(YY^T - YX^T W^T - WXY^T + WXX^T W^T + \lambda W^T W)) = 0 \tag{16}$$

That is, weight matrix, W can be calculated as,

$$W_{p \times n} = Y_{p \times N} X^T{}_{N \times n} (X_{n \times N} X^T{}_{N \times n} + \lambda I_{n \times n})^{-1} \tag{17}$$

2.2.2 Classification based on RLS

Let the training data, training labels, testing data and predicted labels be represented by X_{tr}, Y_{tr}, X_{te}, and Y_{pr}, respectively. The weight matrix, W is calculated from training data, X_{tr} and training labels, Ytr according to the Equation (8).With the calculated weight matrix, W the output class labels are predicted using the equation $Y_{pr} = W X_{te}$

Some of the kernels available in GURLS library are: Linear, RBF and Randfeat kernel. The mathematical formulation of Linear and RBF kernel is discussed in section 2.A.1. The concept and formulation of randfeat kernel is given below,

2.2.2.1 Randfeat Kernel

The randfeat kernel in GURLS library uses the concept of Random Kitchen Sink algorithm[13]. Random Kitchen Sink algorithm is a fast and efficient feature mapping mechanism used to generate random features whose inner products approximate popular kernels. Randomized features makes the kernel machine efficient and less expensive for high dimensional classification. The detailed mathematical description of Random Kitchen Sink algorithm is given by Rahimi in[13].

$$k(x-y) \approx \left\langle \frac{1}{\sqrt{m}} \begin{bmatrix} \cos(x^T\Omega_1) \\ \cos(x^T\Omega_2) \\ \vdots \\ \cos(x^T\Omega_p) \\ \sin(x^T\Omega_1) \\ \sin(x^T\Omega_2) \\ \vdots \\ \sin(x^T\Omega_m) \end{bmatrix}, \frac{1}{\sqrt{m}} \begin{bmatrix} \cos(y^T\Omega_1) \\ \cos(y^T\Omega_2) \\ \vdots \\ \cos(y^T\Omega_p) \\ \sin(y^T\Omega_1) \\ \sin(y^T\Omega_2) \\ \vdots \\ \sin(y^T\Omega_m) \end{bmatrix} \right\rangle = \langle \Phi(x), \Phi(y) \rangle \tag{18}$$

3. Methodology

The proposed work compares and analyses the performance of LIBSVM and GURLS library in context of hyper spectral image classification. The assessment is based on the performance indicators such as classification accuracy and computation time. The experimental analysis considers Linear, Polynomial (Poly), Radial Basis Function (RBF) and sigmoid kernel function available in LIBSVM package and Linear, RBF and Randfeat Kernel in GURLS. The parameters required for each kernel method are tuned using the appropriate cross validation technique. Figure 3 shows a schematic view of the most important processes in the proposed work.

Consider a 3 dimensional hyper spectral data cube represented as $\chi \in \Re^{n_1 \times n_2 \times n_b}$ where each image is of size $n_1 \times n_2$ and n_b represents the total number of bands. The highly noisy bands present in the data are removed manually in advance as they lack detailed information for HSI classification. For the ease of processing, the 3 dimensional data is converted to 2 dimensional form as $\hat{\chi} \in \Re^{n_b \times N}$ where $N = n_1 \times n_2$ represents the total number of pixels in the image. Here, each row vector represents image in a particular band. This vectorised 2D hyper spectral data is given as input to the kernel library. The experiments were carried out on two standard hyper spectral datasets, Salinas

A and Indian pines subscene. Classification includes separation of data into training and testing samples. In the proposed work, the training and testing samples are pixels present in hyper spectral image without applying any pre-processing or feature extraction technique. The training set is generated from 10% of pixels chosen randomly from each class. All the pixels excluding the background pixels are used in the testing phase for validation.

3.1 LIBSVM

The experiment is first carried out using LIBSVM library. The input hyper spectral data is given to different kernel functions available in LIBSVM such as Linear, Poly-Nominal, Radial Basis Function and Sigmoid for classification tasks. The model selection in LIBSVM is performed by cross validation or grid search approach. The kernel parameters C, γ and d is chosen manually by using k - fold cross validation technique. In k-fold cross validation method, the data is divided into k different folds where one fold is used for validation and the rest is given for training. The best parameter value obtained after cross validation is used for training and testing.

3.2 GURLS

The input hyper spectral data is also given to GURLS library for classification. The GURLS library performs the learning experiment in a sequence of tasks called as the learning pipeline. The processes or phases within the same experiment share a common ordered sequence of tasks. All the configuration parameters required for customizing the tasks behavior is stored in an "options structure" referred as OPT. The result of each task is stored for subsequent processing which may help the users to skip execution of certain tasks in the pipeline. The major task categories in GURLS are: splitting the data (cross validation), computation of kernel, selection of model, training, testing and performance assessment. The proposed experiment compares three different kernels, Linear, RBF and Randfeat available in the GURLS library. The performance for each kernel with Hold Out (HO) cross validation and Leave One Out (LOO) cross validation is analyzed in terms of accuracy and time.

4. Experimental Results

In this section, we present the experiments to compare the performance of the GURLS library with that of LIBSVM. The section gives an overview of the datasets used and accuracy assessment measures which is followed by the experimental result analysis.

4.1 Dataset Description

Experiments were carried out using subset portion of two standard hyper spectral datasets, Salinas and Indian Pines. The images are captured using NASA's AVIRIS sensor operating in the wavelength range of 400 to 2450 nm which covers the visible and infrared region in the electromagnetic spectrum.

4.1.1 Salinas A

Salinas A scene is a subset of Salinas scene captured over Salinas valley, California. The pixels in the Salinas-A are located within the Salinas scene at [samples, lines] = [591–676, 158–240] .The image contains 86 x 83 pixels and 224 spectral channels. Out 224 bands, 20 [108–112], [154–167], 224 bands are discarded due to water absorption. The ground truth image contains six different classes which are mutually exclusive.

4.1.2 Indian Pines Subset

A subset portion of the Indian Pines image, called the subset scene was also used to evaluate the performance of the proposed method. The scene was captured over the Indian Pines test site, North-western Indiana in June 1992. The subset is a part of 145 x 145 scene and contains pixels located within Indian pines scene at [27–94] [31–116]. The size of subset portion is 68 x 86 with 224 spectra channels. The subset scene contains 4 different classes with similar spectral signatures.

4.2 Accuracy Assessment Measures

Accuracy assessment is an essential part of any classification. It helps to evaluate the performance of chosen classi-

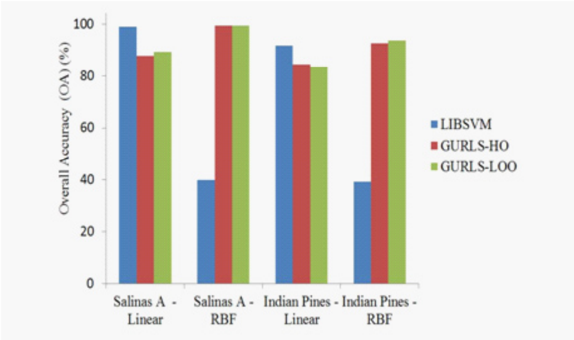


Figure 2. Comparison of Overall Accuracies (OA) obtained using Linear and RBF kernel of LIBSVM and GURLS.

fier. Accuracy assessment can be done qualitatively and quantitatively by comparing the obtained data with some reference data. In hyper spectral domain, the reference data is called as ground truth information where the data is assumed to be 100 % correctly classified. Qualitative assessment involves generation of thematic/classification map and quantitative assessment involves computation of confusion matrix. The classification accuracy measures such as Class wise Accuracy, Overall Accuracy, Average Accuracy and Kappa Coefficient may be derived from the confusion matrix.

The diagonal elements of the confusion matrix represent the number of correctly classified pixels in each class. The Class wise Accuracy, Overall Accuracy, Average Accuracy and Kappa Coefficient is given as:

$$CA = \frac{Correctly\ classified\ \ pixels\ in\ each\ class}{Total\ \ No.of\ \ pixels\ in\ each\ class} \tag{19}$$

$$AA = \frac{Sum\ of\ \ accuracies\ of\ \ each\ class}{Total\ \ No.of\ \ class} \tag{20}$$

$$OA = \frac{Total\ \ No.of\ \ correctly\ classified\ \ pixels}{Total\ \ No.of\ \ pixels} \tag{21}$$

Where N is the total number of pixels, A is the number of correctly classified pixels, B is the sum of product of row and column total in confusion matrix. OA: Overall Accuracy, AA: Average Accuracy, CA: Classwise Accuracy

4.3 Result Analysis

The proposed work compares and assess the performance of different kernel methods available in LIBSVM and GURLS library for hype rspectral image classification. The evaluation is based on HSI accuracy assessment measures such as Overall Accuracy (OA), Average Accuracy (AA), Kappa Coefficient (K) and computation time. The experiments were conducted using the two standard hyper spectral data subsets, Salinas A and Indian Pines subscene. The ability of software packages to deal with low sized training data is analyzed by using a training set containing 10% of total pixels. The experimental analysis considers LIBSVM's: Linear, Polynomial (Poly), Radial Basis Function (RBF) and Sigmoid kernel and GURLS's: Linear, RBF and Randfeat kernel. In addition, a comparative assessment of different cross validation schemes such as Kfold, Hold Out (HO) and Leave One Out (LOO) is also carried out for both the libraries.

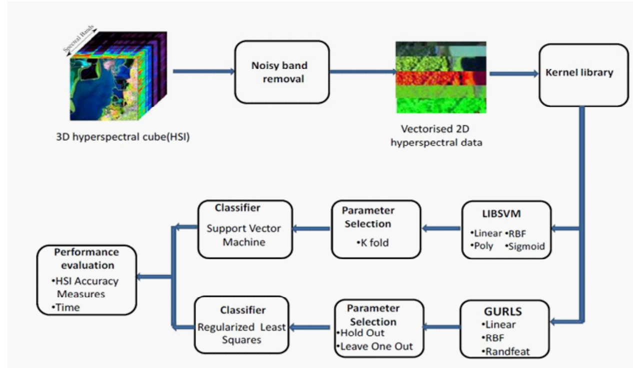


Figure 3. A schematic view of the most important processes in the proposed work.

4.3.1 *Parameter Selection*

It is well known that the performance of kernel methods depends upon the good setting of kernel parameters. The RBF kernel has two control parameters: C and γ. In LIBSVM, these parameters are selected by the users using a k-fold cross validation approach. The experiment uses 5- fold cross validation scheme for parameter selection. That is, the data is divided into five subsets where one subset in used for validation and the remaining four subsets are used for training. Users can provide a set of values for C and the procedure is iterated 5 times for all the possible combinations of C (10,100,500) and (0.1,0.9,10). The best parameters are the ones which produce highest cross validation accuracy. For Salinas A, the control parameters are C = 500 and γ = 0.1 and for Indian Pines subset C = 100 and γ = 0.9. For polynomial kernel, degree of 3 was chosen for analysis. On the other hand, GURLS library has the advantage of automatic parameter selection which reduces the training complexity compared to LIBSVM. The GURLS library selects the optimal parameter for classification without taking any input from the user.

4.3.2 *Analysis of Accuracy Assessment Measures*

Table 4 shows the comparison of Overall Accuracy (OA), Average Accuracy (AA) and Kappa Coefficient (K) obtained with LIBSVM and GURLS package for both the datasets. The results clearly shows that, polynomial kernel in LIBSVM library provides highest OA = 99.13%, AA = 99.15% and K = 0.9892 for Salinas A scene and OA = 91.92%, AA = 92.37 % and K = 0.8847 for Indian Pines subset scene. The performance of RBF kernel in LIBSVM is poor whereas in GURLS, RBF kernel produces the best OA = 99.43% for Salinas A and

OA = 93.68% for Indian Pines subset. It is worth noticing that, RBF and sigmoid kernel in LIBSVM package is not suitable for hyper spectral image classification. Satisfactory results are produced by all the kernels in GURLS when compared to LIBSVM. Table 2 and 3 shows the comparison of class wise accuracies obtained for Salinas A and Indian Pines subset scene respectively. The obtained results shows that the class C4 of Salinas A achieved 100% accuracy with linear and RBF kernel of GURLS and linear kernel of LIBSVM. It is observed that class C1, C2, C4 and C6 of Salinas A and class C2 of Indian Pines subset are almost correctly classified with all the kernels of both the libraries.

4.3.3 Analysis of Computation Time

This section describes about the computation time required by each method for both the data sets. The experiment is performed on Intel(R) Core(TM) i7-4790S CPU, 8.00 GB memory and 64 bit OS using MATLAB 2013. Table 1 shows the time analysis in seconds of kernel methods in LIBSVM and GURLS for hyper spectral datasets Salinas A and Indian Pines subset and Figure 4 shows the time analysis plot for classification of both the datasets using linear and RBF kernel of GURLS and LIBSVM. The analysis clearly shows that LIBSVM has much less computation time compared to GURLS for both the datasets. However, the computation time is only a few seconds which can be compromised to obtain higher classification accuracy.

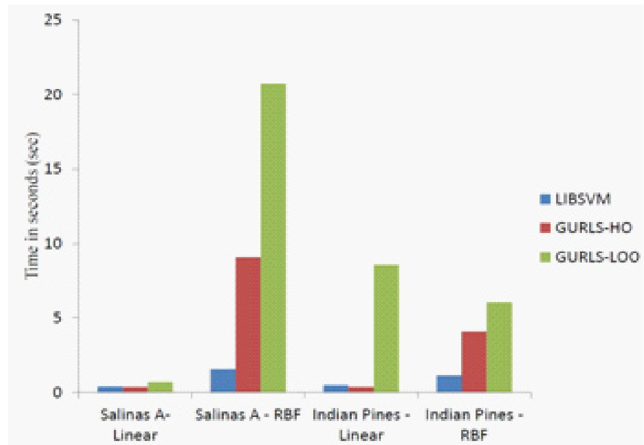


Figure 4. Time analysis plot for classification of hyper spectral datasets using Linear and RBF kernel of LIBSVM and GURLS.

Table 1. Time analysis (sec) of kernel methods in LIBSVM and GURLS for hyper spectral datasets

Dataset	LIBSVM				GURLS					
	Linear	Poly	RBF	Sigmoid	Linear		RBF		Randfeat	
					HO	LOO	HO	LOO	HO	LOO
Salinas A	0.3895	0.3834	1.5438	1.1942	0.354	0.681	9.103	20.748	8.167	21.477
Indian Pines	0.465	0.495	1.133	0.841	0.357	8.586	4.094	6.029	3.320	5.203

Table 2. Comparison of class wise accuracies obtained with LIBSVM and GURLS for Salinas A

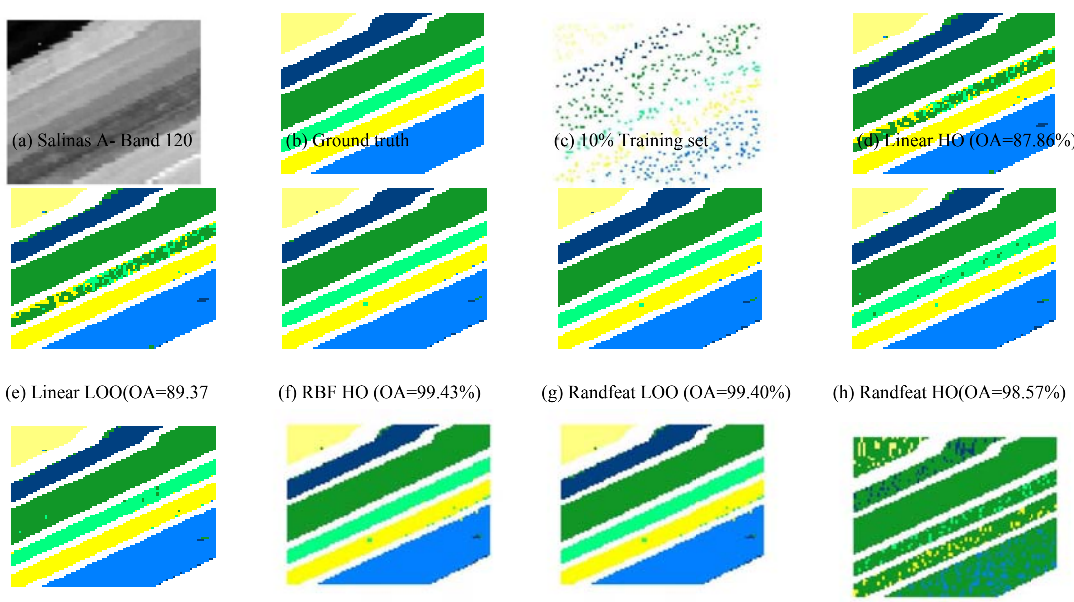
Library	Kernel	Parameters, Cross validation(CV)	C1	C2	C3	C4	C5	C6
LIBSVM	Linear	C = 500, γ = 0.1 CV = 5-fold	99.74	98.58	96.91	100	99.40	97.99
	Polynomial	C = 500, γ = 0.1, d = 3 CV = 5-fold	99.74	98.73	99.67	99.93	99.85	96.99
GURLS	Linear	CV = HO	99.48	99.03	90.90	100	14.83	99.49
		CV = LOO	99.48	98.95	91.07	100	19.58	99.49
	RBF	CV = HO	99.74	98.88	99.67	100	99.40	98.99
		CV = LOO	99.74	98.65	99.67	100	99.55	98.99
	Randfeat	CV = HO	99.48	98.73	96.26	99.93	95.84	99.37
		CV = LOO	99.48	98.88	96.26	99.80	97.77	99.37

Table 3. Comparison of class wise accuracies obtained with LIBSVM and GURLS for Indian pines subset

Library	Kernel	Parameters, Cross validation(CV)	C1	C2	C3	C4
LIBSVM	Linear	C = 100, γ = 0.9, CV= 5fold	87.66	99.86	89.89	91.59
	Polynomial	C = 100, γ = 0.9, d = 3 CV = 5fold	87.66	99.72	90.30	91.80
GURLS	Linear	CV = HO	74.62	99.45	65.30	91.53
		CV = LOO	76.71	99.31	72.67	91.85
	RBF	CV = HO	90.44	99.86	90.98	91.69
		CV = LOO	91.64	99.58	94.12	92.32
	Randfeat	CV = HO	80.59	97.48	86.29	92.59
		CV = LOO	86.56	98.21	84.01	91.06

Table 4. Comparison of Overall Accuracy (OA), Average Accuracy (AA) and Kappa Coefficient (K) obtained with LIBSVM and GURLS for the two hyper spectral datasets

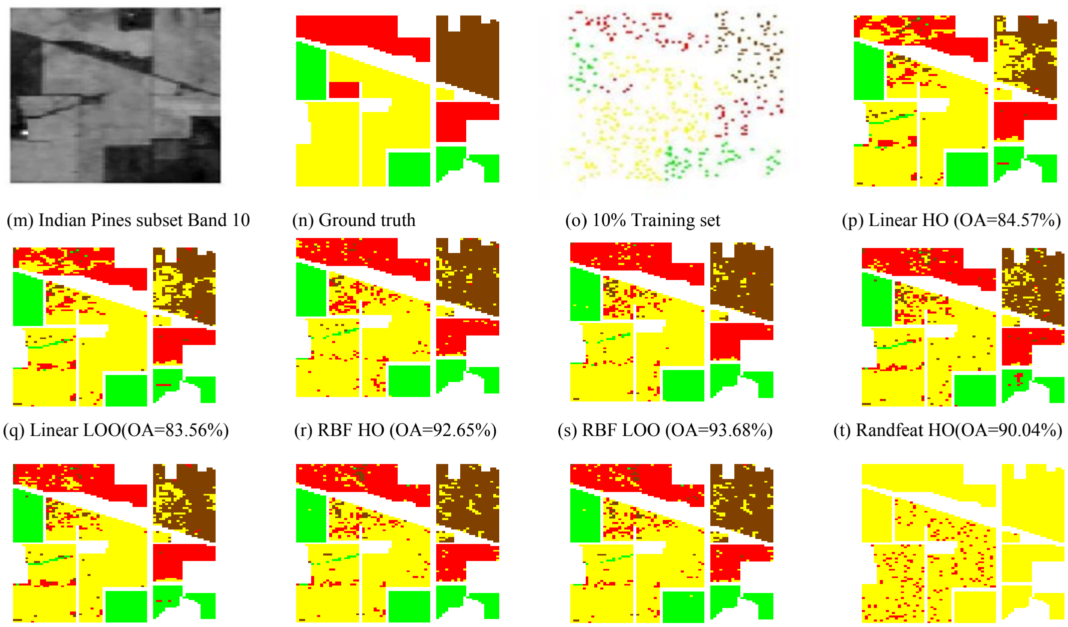
Dataset		LIBSVM				GURLS					
		Linear	Poly	RBF	Sigmoid	Linear		RBF		Randfeat	
						HO	LOO	HO	LOO	HO	LOO
Salinas A	OA	98.89	99.13	39.94	28.51	87.86	89.37	99.43	99.40	98.57	98.82
	AA	98.77	99.15	30.55	16.66	83.96	85.95	99.45	99.43	98.27	98.59
	K	0.9861	0.9892	0.1758	0	0.8452	0.8649	0.9929	0.9925	0.9821	0.9852
Indian Pines	OA	91.78	91.92	39.19	43.54	84.57	83.56	92.65	93.68	90.04	89.59
	AA	92.25	92.37	22.50	25	82.72	80.92	93.24	94.42	89.96	89.24
	K	0.8827	0.8847	-0.6024	0	0.7744	0.7576	0.8952	0.9101	0.8576	0.8502



(a) Salinas A- Band 120 (b) Ground truth (c) 10% Training set (d) Linear HO (OA=87.86%)

(e) Linear LOO(OA=89.37 (f) RBF HO (OA=99.43%) (g) Randfeat LOO (OA=99.40%) (h) Randfeat HO(OA=98.57%)

(i) Randfeat LOO (OA=98.82%) (j) SVM Linear (OA=98.89%) (k) SVM Poly (OA=99.13%) (l) SVM RBF (OA=39.94%)



(m) Indian Pines subset Band 10 (n) Ground truth (o) 10% Training set (p) Linear HO (OA=84.57%)

(q) Linear LOO(OA=83.56%) (r) RBF HO (OA=92.65%) (s) RBF LOO (OA=93.68%) (t) Randfeat HO(OA=90.04%)

(u) Randfeat LOO(OA=89.59%) (v) SVM Linear (OA=91.78%) (w) SVM Poly (OA=91.92%) (x) SVM RBF (OA=39.19%)

Figure 5. Classification maps of hyper spectral datasets, Salinas A and Indian Pines using kernels of LIBSVM and GURLS.

Table 5. Number of training and testing pixels of Salinas A

Class No	Class Name	Train (10%)	Total
1	Brocoli green weeds 1	39	391
2	Corn senesced greenweeds	134	1343
3	Lettuce romaine_4wk	62	616
4	Lettuce romaine_5wk	153	1525
5	Lettuce romaine_6wk	67	674
6	Lettuce romaine_7wk	80	799
Training set size		535	5348

Table 6. Number of training and testing pixels of indian pines subset

Class No	Class Name	Train (10%)	Total
1	Corn notill	39	391
2	Grass trees	134	1343
3	Soybean notill	62	616
4	Soybean mintill	153	1525
Training set size		535	5348

5. Conclusion

GURLS toolbox is quite a recent addition to several methods for supervised learning. This paper examines the performance of two recent promising kernel based software libraries, GURLS and LIBSVM for hyper spectral image classification. The proposed work compares and analyses the performance of different kernels available in both the libraries. The analysis is based on HSI classification accuracy measures and computation time. The experiment is performed on two real hyper spectral datasets namely, Salinas A and Indian Pines subset scene. The study shows that both the libraries are well suited for hyper spectral image classification. It is observed that GURLS library is relatively simple compared to LIBSVM and has the advantage of memory mapped storage with distributed task execution. The computation time is in favor of LIBSVM whereas the computational complexity is significantly less for GURLS package.

6. Acknowledgment

The authors would like to thank all the anonymous reviewers for the valuable comments and suggestions.

7. References

1. Bioucas-Dias JM, Plaza A, Camps-Valls G, Scheunders P, Nasrabadi NM, Chanussot J. Hyperspectral remote sensing data analysis and future challenges. IEEE Geoscience and Remote Sensing Magazine. 2013; 1(2):6–36.
2. Hughes G. On the mean accuracy of statistical pattern recognizers information theory. IEEE Transactions on Information Theory. 1968; 14(1):55–63.
3. Camps-Valls G, Bruzzone L. Kernel-based methods for hyper spectral image classification. IEEE Journal on Selected Topics in Applied Earth Observations and Remote Sensing. 2005; 43(6):1351–62.
4. Camps-Valls G, Gomez-Chova L, Muoz-Mar J, Vila-Francs J, Calpe-Maravilla J. Composite kernels for hyper spectral image classification. IEEE Geoscience and Remote Sensing Letters. 2006; 3(1):93–7.
5. Swain PH. Fundamentals of pattern recognition in remote sensing. Remote sensing: The quantitative approach. 1978; 136–87.
6. Schlkopf B, Smola AJ. Learning with kernels: Support vector machines, regularization, optimization, and beyond.
7. Soman KP, Loganathan R, Ajay V. Machine learning with SVM and other kernel methods. PHI Learning Pvt. Ltd; 2009.
8. Hearst MA, Dumais ST, Osman E, Platt J, Scholkopf B. Support vector machines. IEEE Intelligent Systems and their Applications. 1998; 13(4):18–28.
9. Melgani F, Bruzzoner L. Classification of hyper spectral remote sensing images with support vector machines. IEEE Transactions on Geoscience and Remote Sensing. 2004; 42(8):1778–90.
10. Gualtieri JA, Chettri SR, Cromp RF, Johnson LF. Support vector machine classifiers as applied to AVIRIS data. Proceedings of Eighth JPL Airborne Geoscience Workshop; 1999.
11. Demr B, Erturk S. Hyperspectral image classification using relevance vector machines. IEEE Geoscience and Remote Sensing Letters. 2007; 4(4):586–90.
12. Chang C-C, Lin C-J. LIBSVM: A Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology (TIST). 2011; 2(3):27.
13. Rahimi A, Recht B. Random features for large-scale kernel machines. In Advances in Neural Information Processing Systems. 2007; 1177–84.
14. Tacchetti A, Mallapragada PK, Santoro M. GURLS: A least squares library for supervised learning. The Journal of Machine Learning Research. 2013; 14(1):3201–5.
15. Minh HQ, Niyogi P, Yao Y. Mercers theorem, feature maps and smoothing. Learning theory. Berlin-Heidelberg: Springer; 2006. p. 154–68.
16. Hsu C-W, Chang C-C, Lin C-J. A practical guide to support vector classification; 2003.
17. Zhang P, Peng J. SVM vs regularized least squares classification. IEEE Proceedings of the 17th International Conference on Pattern Recognition (ICPR); 2004. p. 176–9.