

University of Texas at Tyler

Scholar Works at UT Tyler

Computer Science Faculty Publications and
Presentations

Computer Science

2018

Generating Classification Rules from Training Samples

Arun D. Kulkarni

University of Texas at Tyler, akulkarni@uttyler.edu

Follow this and additional works at: https://scholarworks.uttyler.edu/compsci_fac



Part of the [Computer Sciences Commons](#)

Recommended Citation

Kulkarni, Arun D., "Generating Classification Rules from Training Samples" (2018). *Computer Science Faculty Publications and Presentations*. Paper 19.

<http://hdl.handle.net/10950/1175>

This Article is brought to you for free and open access by the Computer Science at Scholar Works at UT Tyler. It has been accepted for inclusion in Computer Science Faculty Publications and Presentations by an authorized administrator of Scholar Works at UT Tyler. For more information, please contact tgullings@uttyler.edu.

Generating Classification Rules from Training Samples

Arun D. Kulkarni

Department of Computer Science
University of Texas at Tyler
Tyler, TX, USA

Abstract—In this paper, we describe an algorithm to extract classification rules from training samples using fuzzy membership functions. The algorithm includes steps for generating classification rules, eliminating duplicate and conflicting rules, and ranking extracted rules. We have developed software to implement the algorithm using MATLAB scripts. As an illustration, we have used the algorithm to classify pixels in two multispectral images representing areas in New Orleans and Alaska. For each scene, we randomly selected 10 per cent of the samples from our training set data for generating an optimized rule set and used the remaining 90 per cent of samples to validate the extracted rules. To validate extracted rules, we built a fuzzy inference system (FIS) using the extracted rules as a rule base and classified samples from the training set data. The results in terms of confusion matrices are presented in the paper.

Keywords—Fuzzy membership functions; classification; rule extraction; multispectral images

I. INTRODUCTION

Many methods have been used to classify pixels in multispectral images using training samples. These include parametric methods such as the maximum likelihood, support vector machines, decision trees, neural networks, fuzzy-neural systems, and fuzzy inference systems. In supervised classification methods during the learning phase, a model is built to map an input feature vector to output classes, and during the classification phase the model is used to classify an unknown sample. The maximum likelihood classification algorithm assumes normal distribution and uses the mean vector and covariance matrix of each class to find the posterior probability. It then assigns a pixel to the class with the higher posterior probability. The Support Vector Machine (SVM) partitions the feature space by using hyper-planes that maximize the distance between the two classes in the feature space [1]. It has been shown that the SVM algorithm yields higher classification accuracy for small datasets compared to conventional classifiers [2]. Neural networks provide a nonparametric method for classification. Neural network models learn from training samples. During the learning process weights are updated using a gradient descent method such that the mean squared error between the desired and actual outputs is minimized [3]. During the decision-making phase the model is used to classify pixels based on their spectral signatures.

Fuzzy-neural systems have been used to classify pixels in Landsat images [4]. Fuzzy logic provides a tool to process

information using linguistic rules. Fuzzy logic in the form of approximate reasoning provides decision support and expert systems with powerful reasoning capabilities. In fuzzy logic class memberships based on a degree of compatibility with the concepts presented are used [5]. A fuzzy inference system (FIS) provides a method to classify pixels in Landsat images. However, the potential of fuzzy inference systems has not been fully explored by the remote sensing community as of yet. The main task in implementing a FIS is to develop a rule base. Classification rules can be generated from training samples or can be obtained from expert's knowledge. These classification rules then can be used to build the FIS. Several methods to generate classification rules from training samples have been reported in the literature. They include extracting classification rules using fuzzy membership functions, decision trees, neural networks, and black-box models. Wang and Mendel [6] suggested a method to extract fuzzy rules from data samples using fuzzy membership functions. They have used the method for a time-series prediction problem, where the output function is a continuous function. Chiu [7] developed a method called subtractive clustering to efficiently extract rules from a high dimensional feature space. The method was able to produce a much simpler fuzzy classifier and could be used to extract rules for function approximation as well as pattern classification. Kulkarni and McCaslin [8] have generated classification rules from neural network models and have built a FIS to classify pixels in Landsat images. Fung et al. [9] developed a cost-efficient method to quickly extract rules from SVMs trained with thousands of samples. Their algorithm forms rule sets that can be easily understood by humans, and only needs simple multivariable optimization problems to be solved. Sicat et al. [10] developed the FIS using farmer's knowledge for agricultural land sustainability classification using fuzzy models. Reshmidevi et al. [11] have developed a fuzzy rule base system for land suitability in agricultural watersheds. They have considered two types of attributes: continuously measured attributes and thematic attributes, and the crop suitability index as the output of the fuzzy rule-based system. They have used heuristic information and farmer's knowledge aggregated through field surveys as the basis for the fuzzy rule-base. Cay and Iscan [12] have developed a fuzzy expert system for land reallocation in land consolidation. They developed a rule base system using farmer's knowledge obtained from survey questions. Meng and Pei [13] have suggested a method to extract linguistic rules from data sets using fuzzy logic and genetic algorithms. They have formalized linguistics based on complex data summaries and used a genetic algorithm to

optimize the number of parameters of membership function of linguistic values. Kulkarni and Khan [14] generated rules to classify Likert-scale survey data by using a multi-layered feed forward neural network. Kulkarni and Shrestha [15] have generated rules using induction trees and built a FIS using the extracted rules.

In this paper we have used the method similar to that suggested by Wang and Mendel [6] for classification of pixels in a Landsat images. In rule extraction the main concerns are the number of extracted rules and the quality of those rules. Technically, each training sample generates a rule, and we get a large number of rules. It is important to note that the generated rules often contain redundant and conflicting rules. Also, a rule set with a large number of rules results in a model that often over-fits the data samples. Generally, rule generation is a two-step process. During the first step all possible rules are generated. In the second step, the rule set is optimized. The suggested algorithm for rule generation is as follows: First, the training data is fuzzified. From the fuzzified data, rules are generated. The generated rules may contain redundant and conflicting rules which are then eliminated. The remaining rules are ranked.

As an illustration, we have considered Landsat scenes from areas in New Orleans and Alaska. We selected training set areas interactively by displaying the scenes. We extracted classification rules from training samples. We built a FIS for each scene using the extracted rule as the rule base and classified all pixels. The outline of the paper is as follows. Section II describes a method for generating classification rules from training samples and optimizing the rule set. Section III provides implementation and results of Landsat data analysis. Section IV provides discussions and results.

II. RULE GENERATION AND OPTIMIZATION

The proposed method for extracting classification rules from data samples and finding the optimized rule set by eliminating conflicting and redundant rules is shown in Fig. 1. The process includes five steps. The first two steps are concerned with rule generation and the last three steps deal with optimization. To illustrate the method, we have chosen a classification problem with two features and three classes, and the training set contains fifty samples from each class. The method can be extended to multiple features and multiple classes. The steps are explained below.

Step-1 Fuzzify Data: We assume a set of desired input-output data pairs as shown in (1).

$$(x_1^1, x_2^1, y^1), (x_1^2, x_2^2, y^2), \dots, (x_1^n, x_2^n, y^n) \quad (1)$$

where (x_1, x_2) represents features, and y represents the corresponding class. For each feature the domain interval is 0 through 10. We divided the domain interval with three fuzzy sets {low, medium, high}. We used trapezoidal membership functions as shown in Fig. 2.

Step-2 Rule Generation: We fuzzified the input values and generated classification rules. Let the input vector $(2.3, 3.5)$ represent class C_1 . From the membership functions

shown in Fig. 2, membership values are given by (2), and the corresponding rule can be stated as If x_1 is low and x_2 is medium then the class is C_1

$$\begin{aligned} \mu_{low}(x_1) &= 0.7, \mu_{med}(x_1) = 0.2, \mu_{high}(x_1) = 0.0 \\ \mu_{low}(x_2) &= 0.0, \mu_{med}(x_2) = 1.0, \mu_{high}(x_2) = 0.0 \end{aligned} \quad (2)$$

We generate a rule using the highest membership values. The firing strength of a rule is given by (3).

$$\begin{aligned} \alpha &= \min(\mu_{low}(x_1), \mu_{med}(x_2)) \\ &= \min(0.7, 1.0) = 0.7 \end{aligned} \quad (3)$$

Each sample pair generates a rule, and the total number of generated rules is equal to the number of samples. The extracted rules contain duplicate and conflicting rules.

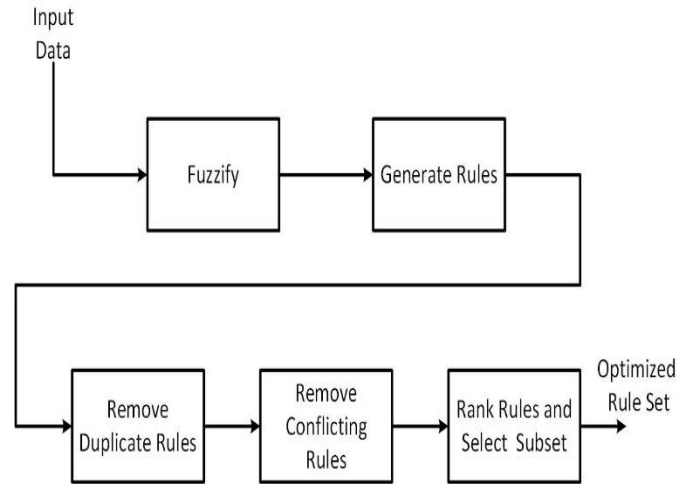


Fig. 1. Rule generation and optimization process.

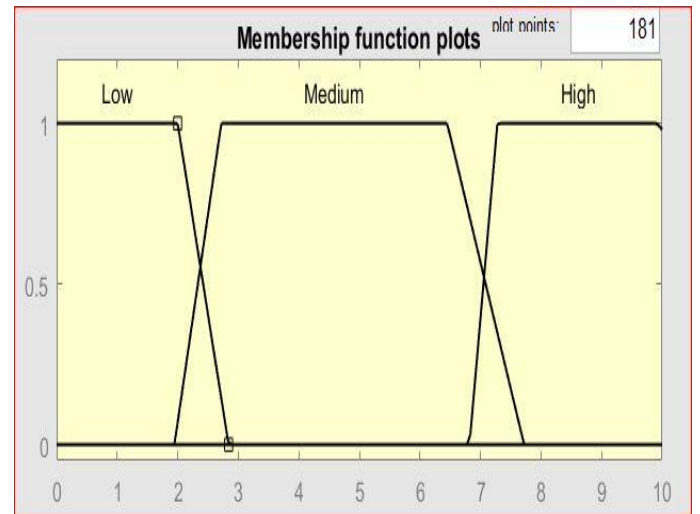


Fig. 2. Fuzzy membership functions.

Step-3 Eliminate duplicate rules: To eliminate repeated rules, extracted rules are mapped onto the Fuzzy Associative Memory (FAM) banks as shown in Fig. 3. In this example

there are three classes and there are 50 samples in each class. There are 150 rules generated as each sample generates a rule. We used three FAM banks, one for each class. Each cell in a FAM bank represents a rule, and the value in the cell represents the count of that rule. It can be seen from Fig. 3 that a rule is as follows: If x_1 is low, and x_2 is low, then class is C_1 . The count for the rule is 32. That means 32 samples satisfied this rule. Looking at the FAM bank in Fig. 3, we can see that by eliminating repeated rules, we get a rule set of only 10 rules. The extracted rules are shown in Table I.

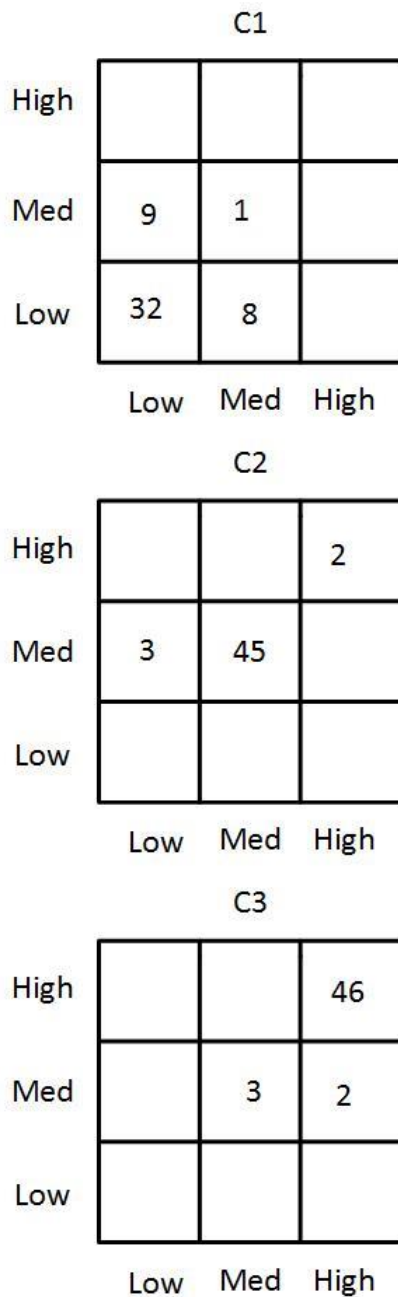


Fig. 3. Fuzzy associative memory (FAM) bank.

TABLE I. RULE SET AFTER ELIMINATING REPEATED RULES

Rule	X_1	X_2	Class	Count
R_1	low	low	C_1	32
R_2	low	medium	C_1	9
R_3	medium	low	C_1	8
R_4	medium	medium	C_1	1
R_5	medium	medium	C_2	45
R_6	low	medium	C_2	3
R_7	high	high	C_2	2
R_8	medium	medium	C_3	3
R_9	high	high	C_3	46
R_{10}	medium	high	C_3	2

TABLE II. RULE SET AFTER ELIMINATING CONFLICTING RULES

Rule	X_1	X_2	Class	Count
R_1	high	high	C_3	46
R_2	medium	medium	C_2	45
R_3	low	low	C_1	32
R_4	low	medium	C_1	9
R_5	medium	low	C_1	8

Step-4 Remove Conflicting Rules: To optimize the generated rules, it is necessary to remove conflicting rules if there are any. Two rules are considered to be conflicting when their antecedent parts are identical while the consequent parts are not the same. The conflicting rule with the highest count is retained, and the other rule is discarded. It can be seen from Table I that Rules 4 and 5 are conflicting rules. For Rule 4 the count is 1, while for Rule 5, the count is 45. Therefore Rule 4 is eliminated. This process is repeated until there are no more conflicting rules.

Step-5 Rank Rules and Select a Subset: After eliminating repeated rules, the remaining rules are organized in descending order from the highest to lowest based on their count. A subset from the ranked rules is then selected using the count as the criterion. Rules with a low count can be excluded. In our example, we removed the rules that represent less than three percent of samples. The final rule set is shown in Table II.

III. IMPLEMENTATION AND RESULTS

In this research work we developed software to generate classification rules from training samples using MATLAB scripts. We also evaluated the extracted rules by classifying pixels in two Landsat scenes. We built FISs with extracted rules as the rule base and classified training set data. The results are provided in this section.

A. Example-1 Landsat Scene from New Orleans

As an example, we considered a Landsat-8 scene from operational Land Imager (OLI) obtained on February 26, 2016; path # 22 and row # 39. We selected an area of the size 512x512 pixels from the full scene. The raw image is shown in Fig. 4. To extract classification rules, we selected six training set areas representing three classes: water, vegetation, and land.

The training set data contained a total of 7000 samples consists of 3400, 1800, and 1800 samples from three classes: water, vegetation, and land, respectively. We used band-2, band-3, band-5, and band-6 as features for classification. We selected these bands because they showed the maximum variance. We used randomly selected ten percent of training samples for generating classification rules. Spectral signatures for the classes are shown in Fig. 5.



Fig. 4. Raw image-new orleans scene.

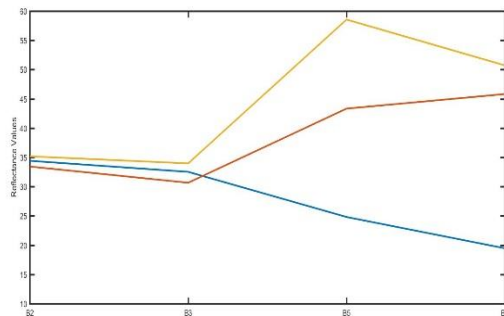


Fig. 5. Spectral signatures - new orleans scene.

TABLE III. OPTIMIZED RULE SET FOR NEW ORLEANS SCENE

	B2	B3	B5	B6	Class	Count
R ₁	low	low	low	low	water	357
R ₂	low	low	medium	medium	vegetation	119
R ₃	low	low	high	medium	land	117
R ₄	low	low	medium	high	vegetation	23
R ₅	low	low	high	high	land	15
R ₆	medium	medium	high	high	land	13
R ₇	medium	low	high	medium	land	9
R ₈	medium	medium	high	medium	land	6
R ₉	low	low	low	high	vegetation	4
R ₁₀	low	low	high	low	vegetation	3

TABLE IV. CONFUSION MATRIX FOR NEW ORLEANS SCENE

	water	vegetation	land
water	3055	0	0
vegetation	51	1564	0
land	0	155	1475



Fig. 6. Classified output new orleans scene.

We used five term sets for each feature: very-low, low, medium, high, and very-high. We used trapezoidal membership functions and generated the optimized rule set using the method outlined in Section II. The extracted optimized rule set contained sixteen rules. The first ten rules of the optimized rule set are shown in Table III. We implemented a FIS with the optimized rule set as a rule base. The process of implementing the FIS is described by Kulkarni & Shrestha [15]. The validation samples were classified using the FIS. The confusion matrix is shown in Table IV. We obtained classification accuracy of 96.73 percent with the FIS system that was built using extracted rules. The classified output is shown in Fig. 6.

B. Example-2 Landsat Scene from Alaska

In this example, we considered Landsat-8 OLI scene from Alaska obtained on June 6, 2016, path # 58 and row # 19. We considered a sub-scene of the size 512 x 512 pixels. The unclassified data for the Alaska scene is shown in Fig. 7. Spectral signatures for four classes are shown in Fig. 8.



Fig. 7. Raw data-Alaska scene.

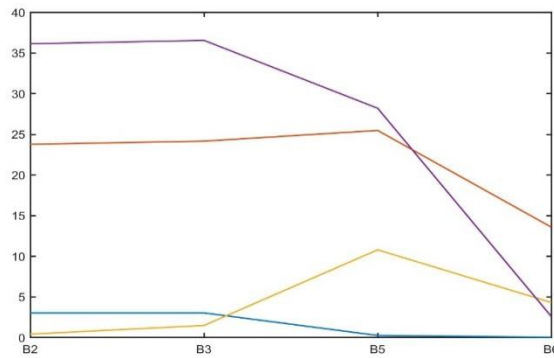


Fig. 8. Spectral signatures - Alaska scene.

To extract classification rules, we selected five training set areas representing four classes: water, vegetation, ice-land, and glaciers. Each selected training area was of the size 100x100 pixels. Our training set data consisted of 50,000 training samples. We used band-2, band-3, band-5, and band-6 as features for classification as these bands showed the maximum variance. We used randomly selected ten percent training samples for generating classification rules.

To define fuzzy membership functions, we used five term sets for each feature: very-low, low, medium, high, and very-high. We extracted fuzzy classification rules using the method described in Section II. The optimized rule set contained twenty rules. The first ten rules are shown in Table V. We implemented a FIS with the optimized rule set as a rule base, and validation samples were classified using the FIS. The confusion matrix is shown in Table VI. The obtained classification accuracy was 91.58 percent. The classified output is shown in Fig. 9.

TABLE V. OPTIMIZED RULE SET FOR ALASKA SCENE

	B2	B3	B5	B6	Class	Count
R ₁	v_low	v_low	v_low	v_low	water	2068
R ₂	v_low	v_low	low	medium	ice_land	851
R ₃	high	high	medium	medium	glaciers	786
R ₄	medium	medium	medium	low	vegetation	282
R ₅	high	high	high	medium	vegetation	281
R ₆	high	high	high	low	vegetation	111
R ₇	medium	medium	high	low	vegetation	83
R ₈	high	high	high	low	glaciers	61
R ₉	medium	medium	high	medium	vegetation	56
R ₁₀	medium	high	high	medium	vegetation	42

TABLE VI. CONFUSION MATRIX FOR ALASKA SCENE

	Water	Vegetation	Ice_land	Glacier
Water	16757	1223	17	0
Vegetation	10	8865	130	0
Ice_land	164	1761	7066	0
Glacier	0	136	347	8524

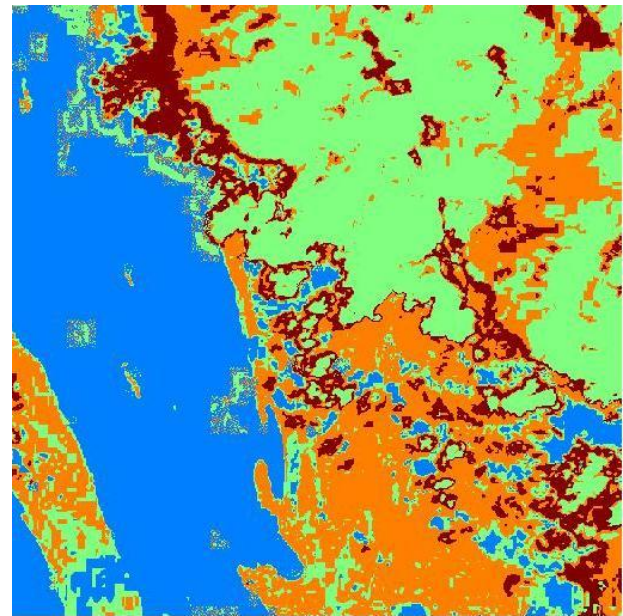


Fig. 9. Classified output - Alaska scene.

IV. DISCUSSIONS AND CONCLUSIONS

In this paper we have suggested an algorithm for generating and optimizing classification rules from training samples using fuzzy membership functions. Furthermore, we developed software using MATLAB scripts to implement the algorithm. As an illustration, we classified pixels from Landsat scenes for two areas in New Orleans and Alaska. We extracted classification rules from training samples for these two scenes. To validate extracted rules, we developed a FIS for each scene using extracted rules a rule base and classified samples from the training sets. The classification accuracy for New Orleans scene was 96.73 percent, and for Alaska, the accuracy was 91.58 percent. This clearly shows that extracting rules using fuzzy membership functions is a valid approach to generate a rule set that can be used develop a FIS for classifying pixels in Landsat images. In our examples we have used five term sets to define fuzzy membership functions. It is possible to use more terms sets to increase granularity, which may lead to an increase in the number of rules in the optimized rule set. It may be noted that as the number of rules in the optimized rule set increases the classification accuracy increases; however, there is a danger of overfitting training data.

The future work includes generating rules using fuzzy membership functions with seven or nine term sets for each membership function. This may increase the number of rules in the optimized rule set and may yield better classification accuracy. Furthermore there is no well-known criterion for evaluating quality of generated rules. That needs to be developed. We also plan a bench mark study to compare accuracy of the suggested algorithm with other existing rule extraction algorithms.

The author is thankful to anonymous reviewers for their valuable comments.

REFERENCES

- [1] N. Vapnik and S. Kotz, Estimation of Dependences Based on Empirical Data, New York: Springer-Verlag, 1982.
- [2] P. Mantero, G. Moser and S. b. Serpico, "Partially supervised classification of remote sensing images through SVM-based probability density estimation," IEEE Transactions on Geoscience and Remote Sensing, vol. 43, no. 3, pp. 559-570, 2005.
- [3] W. Y. Huang and R. P. Lippmann, "Neural Net and Traditional Classifiers," Neural Information Processing Systems, pp. 387-396, 1988.
- [4] A. D. Kulkarni, "Neural-Fuzzy Models for Multispectral Image Analysis," Applied Intelligence, vol. 8, no. 2, pp. 179-187, 1998.
- [5] G. Alba, "Remote Sensing Classification. Algorithms Analysis Applied to Land Cover Change," in Master in Emergency Early Warning and Response Space Application. Mario Gulich Institute, 2014.
- [6] L.X. Wang and J. M. Mendel, "Generating Fuzzy Rules by Learning from Examples," IEEE Transactions on Systems, Man, and Cybernetics, vol. 22, no. 6, pp. 1414-1427, 1992.
- [7] S. L. Chiu, "An Efficient Method for Extracting Fuzzy Classification Rules from High Dimensional Data," Advanced Computational Intelligence, vol. 1, no. 1, pp. 1-7, 1997.
- [8] A. Kulkarni and S. McCaslin, "Knowledge Discovery from Multispectral Satellite Images," IEEE Geoscience and Remote Sensing Letters, vol. 1, no. 4, pp. 246-250, 2004.
- [9] G. Fung, S. Sandilya and R. B. Rao, "Rule Extraction from Linear Support Vector Machines," in Procs. ACM/SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, Chicago, 2005.
- [10] R. S. Sicut, E. J. M. Carranza and U. B. Nidumolu, "Fuzzy Modeling of Farmers' knowledge for land suitability classification," Agricultural Systems, vol. 83, pp. 49-75, 2005.
- [11] T. V. Reshmidevi, T. I. Eldho and R. Jana, "A GIS-Integrated Fuzzy Rule-Based Inference System for Land Suitability Evaluation in Agricultural Watersheds," Agricultural Systems, vol. 101, no. 1-2, pp. 101-109, 2009.
- [12] T. Cay and F. Iscan, "Fuzzy Expert System for Land Reallocation in Land Consolidation," Expert Systems with Applications, vol. 38, no. 9, pp. 11055-11071, 2011.
- [13] D. Meng and Z. Pei, "Extracting Linguistic Rules from Data Sets Using Fuzzy Logic and Genetic Algorithms," Neurocomputing, vol. 78, pp. 48-54, 2012.
- [14] I. Khan and A. Kulkarni, "Knowledge Extraction from Survey Data Using Neural Networks," Procedia Computer Science, vol. 20, pp. 433-438, 2013.
- [15] A. Kulkarni and A. Shrestha, "Multispectral Image Analysis using Decision Tress," International Journal of Advanced Computer Science and Applications, vol. 8, no. 6, pp. 11-18, 2017.