IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 30-36
www.iosrjournals.org

Star Classification using Tree based Data Mining Techniques

Dr. R. A. Ingolikar[1], S. R. Gedam[2]
[1](*Department of Computer Science, S.F.S College, Seminary Hills, Nagpur (MS), India)*
[2](*Inter Institutional Computer Centre; RTM Nagpur University, Nagpur (MS). India)*

Abstract: *Classification of Astronomical Data having large database is a troublesome activity. Data mining decision tree based techniques are applied for classification of star. The objective of the work is to evaluate the effectiveness of random forest on astronomical object classification. Random forest is an ensemble based classifier method where each classifier is decision trees. Results shows that ensemble method performs better as compared to single tree based classifier.*
Keywords - *Ensemble, Random forest, Decision trees, Classification*

I. INTRODUCTION

Data mining is finding out facts about the data. Classification is a task which is a part of data mining. Classification process classifies the new data with the help of model already built using initially available data of the similar type. Building model means generating rules from the available data and then implementing it. The model learns from the available data and then classifies the new data. This type of classification comes under predictive modeling. These types of models are helpful in solving problems in different areas, such as medicine, industry, education, security, astronomy and many more [1].

Data mining techniques have already been applied on Astronomical data. Several techniques of Data mining have been used to solve tasks in Astronomy. Some of them are : application of Bayesian analysis to the problem of star formation in young galaxies[3]; a Bayesian Markov Chain Monte Carlo method to determine whether the stars in the galaxies form in one monolithic collapse of a giant gas cloud, or if they form in a hierarchical fashion ; the use of computer vision and artificial neural network [4] in an application that classifies large number of galaxies which show up in the thousands of digitized images from sky surveys. Other works are the use of support vector machines [5] to explain the determination of the photometric redshift estimate for distant galaxies and the use of a decision tree for classifying spatial data streams using a data structure called Peano Count Tree[4].

In this paper classification is performed on stars. Classification is a two step process, consisting of a learning step(where a classification model is constructed) and a classification step(where the model is used to predict class label for given data)[2]. In this work Random forest algorithm for stellar spectral classification of stars is proposed. Data is generated using Slogan Digital Sky Survey (SDSS) database. This paper is organized as follows. Section Method introduces the random forest algorithm. Section Data describes the data used in the experiments. Section Experiments and Results shows the experimental results. Section Conclusion presents the conclusion of this work.

II. METHOD

Random forest is a recently proposed ensemble method [6] which uses many tree classifiers and aggregates their results. The individual decision trees are generated using a random selection of attributes at each node to determine the split. The CART (Classification and Regression trees) methodology is used to grow the trees. The CART applies greedy, top-down binary approach for tree construction. The trees are grown to the maximum size and are not purned. During classification, each tree votes and the most popular class is returned.

OOB Error

In random forest algorithm, sampling method from training data is based on early bagging method [7] (bagging-bootstrap aggregation : parallel combination of learners, independently trained on distinct bootstrap samples) which uses bootstrap sampling method to generate different training sets. Nearly 37 percent of the sample will not be chosen to construct classifiers. These nearly 37 percent data are called out of bag data (OOB). These OOB data can be used to estimate the generalization error of the trees. For each tree, we get an OOB error estimate. The generalization error of random forests can be obtained by averaging all the OOB error estimates of trees.

Random Forest Algorithm

1. For b= 1 to B

a) Draw a bootstrap sample(new training sets by random sampling) Z of size N from the training data

b) Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree until the minimum node size n_{min} is reached.

i. Select m variables at random from the p variables.

ii. Pick the best variable split- point among the m.

iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$.

III. DATA

The Slogan Digital Sky Survey(SDSS) is the largest optical survey of the astronomical bodies(objects) including stars, galaxies, asteroids etc., and contains data of ~10^9 objects(data release 9) covering 1/3 of sky[8]. The images are taken in five photometric bands u, g, r, i and z in the optical wavelength range 0.3-1.0μm. Figure 1 shows the image of a star and Figure 2 shows its spectra.

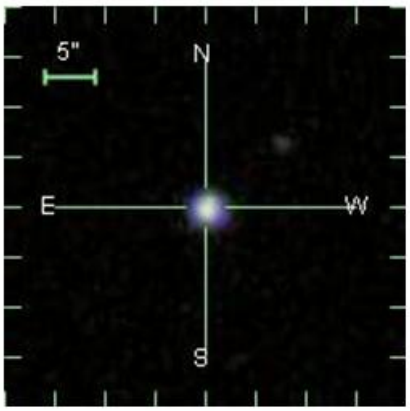

Figure1. Image of star

These bands provide enough information to broadly classify these objects as stars. From the available spectrum of the individual object redshift, velocity, intensity of light, temperature are calculated. As the wavelength in the available spectra ranges from 3800 to 9200 Å[8], only this range is considered.

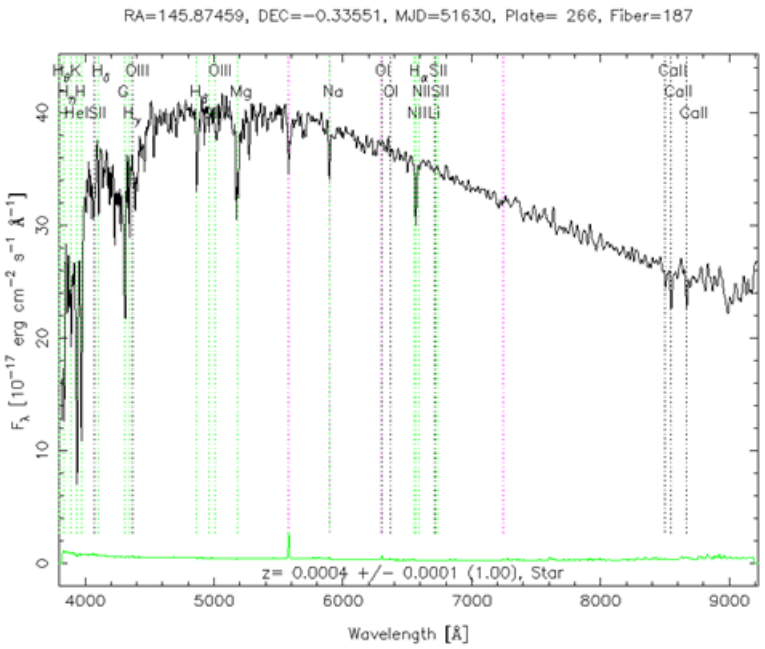

Figure 2. Spectrum of star in Figure 1

IV. EXPERIMENTS AND RESULTS

Random forest classifier is designed using Weka Software[9]. Morgan-Keenan(MK) system is widely used in astronomy classification. According to MK classification, stellar spectra were divided into totally 10 star class types: O, B, A, F, G, K, M, R, S, N. Star class type is taken as the target output of the random forest. As the wavelength in the available spectra ranges from 3800 to 9200 Å , designed model only classifies A, F, G, K, M star class types. This section shows the prediction results of the developed model.

Random forest is generated using 25 trees, each tree is constructed while considering 3 attributes randomly and depth of trees is taken as 3.

In Weka Software the parameters that are set for constructing the random forest are

numTrees (number of trees)= 25,

maxDepth (depth of the tree) =3,

numFreatures (number of attributes to be used in random selection) =3.

Table. 1 shows the classification results using Random forest. From table 1 it is observed that this algorithm gets a good performance(Accuracy) of about 96.60% .True positive rate are 100%, 100%, 0%, 100%, 100% and false positive rate are 0%, 3%, 0%, 2%, 0% of Class A, F, G, K, M respectively. For five class types, higher ROC Area 1, 1, 0.750, 1, 1 is obtained.

Table 1 Performance of SDSS Data

TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	A
1.000	0.034	0.968	1.000	0.984	0.967	1.000	1.000	F
0.000	0.000	0.000	0.000	0.000	0.000	0.982	0.750	G
1.000	0.022	0.933	1.000	0.966	0.955	1.000	1.000	K
1.000	0.000	1.000	1.000	1.000	1.000	1.000	1.000	M
0.966	0.023	0.934	0.966	0.950	0.939	0.999	0.992	Weighted Avg

The terms used in the Table 1 are[2]

TP Rate- True Positives Rate refers to percentage of positive tuples (tuples of the main class of interest) that were correctly labeled by the classifier.
FP Rate –False Positive Rate refers to percentage of negative tuples (all other tuples) that were incorrectly labeled as positive.
Precision- Measure of exactness (what percentage of tuples labeled as positive are actually such)
Recall- Measure of completeness (what percentage of positive tuples are labeled as such)
F- Measure- It is the combine measure that assesses both Precision and Recall. F-Measure is harmonic mean of Precision and Recall.
MCC- Mathews correlation coefficient (MCC) measures the correlation of the actual and predicted class.
ROC Area- A Receiver Operating Characteristics Curve (ROC) for a model shows a relationship between true positive rate and false positive rate. The area under the ROC curve is a measure of accuracy of the model. The higher ROC Area value denotes better model.
PRC Area – A Precision Recall Curve (PRC) shows a relationship between Precision and Recall. The area under PRC is another way to measure accuracy of the model which does not consider true negative. The higher PRC Area value denotes better model.

The formulae used for calculation[2] are

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \dots\dots\dots\dots\dots\dots (1)$$

$$Precision = \frac{TP}{TP + FP} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (2)$$

$$Recall = \frac{TP}{TP + FN} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (3)$$

$$F = \frac{2 * precision * recall}{precision + recall} \dots\dots\dots\dots\dots\dots\dots\dots\dots (4)$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)(TP + FP)(TN + FN)(TN + FP)}} \dots\dots\dots\dots (5)$$

where TP, TN, FP, FN refer to number of true positive, true negative, false positive and false negative samples respectively.
Confusion matrix is used to analyze the classification problem. It tells how well the classifier has recognized tuples of different classes. True Positive(TP) and True Negative(TN) tells when the classifier is getting things right, while False Positive(FP) and False Negative(FN) tells when the classifier is getting things wrong.
Table. 2 gives confusion matrix of SDSS Data using random forest algorithm. We correctly predict 6 as class A, 30 as class F,14 as class K and 7 as class M sample data. But we also wrongly predicted 1 as class F and 1 as class K sample data.

Table 2 Confusion Matrix of SDSS Data

a	b	c	d	e	classified as
6	0	0	0	0	a = A
0	30	0	0	0	b = F
0	1	0	1	0	c = G
0	0	0	14	0	d = K
0	0	0	0	7	e = M

Parameter Tuning

To achieve optimal performance of random forest model, number of trees in the forest and number of attributes to be used in random selection are varied keeping depth of each tree constant. The parameters whose values were varied are numTrees(number of trees) , numFeatures(number of attributes to be used in random selection) and depth of each tree is kept constant i.e., maxDepth(depth of tree) = 3. To avoid negative values root mean square error(RMSE)is used for checking performance of the model. RMSE is measure of differences between values predicted by a model and the values actually observed.

numTrees was varied in {10,15,20,25,30,35,40,45,50} and numFeatures was set to 3. Fig 3 shows the relationship between Root Mean Square Error (RMSE) and different numTrees values. For each numTrees we run the program four times and got an average RMSE. From Fig. 3 it is observed that RMSE is lower when numtrees is 20.

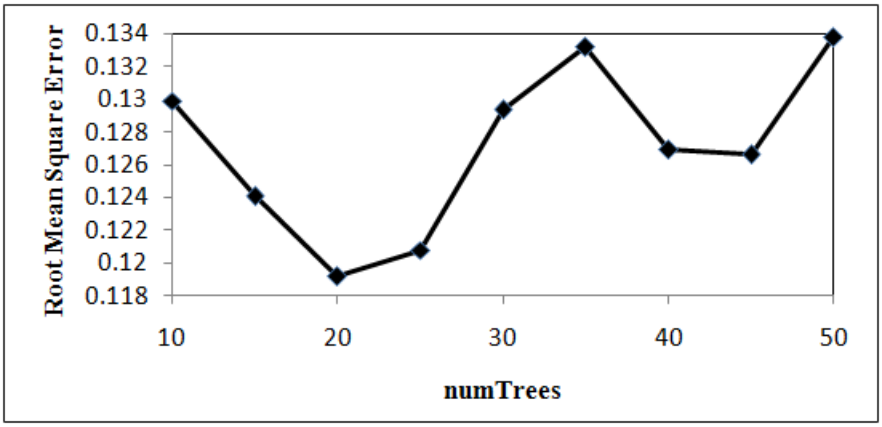

Figure 3. Relationship between numTrees and RMSE

The attribute numFeatures is varied in the same way. The range of numFeatures is {2,3,4,5,6,7,8,9,10} and numTrees is set as 20 (best value observed in Figure 3). Fig. 4 shows the relationship between numFeatures and RMSE. From Fig. 4, it can be concluded that when the number of features are more the RMSE tends to lower. Table 3 shows the details of parameter tuning.

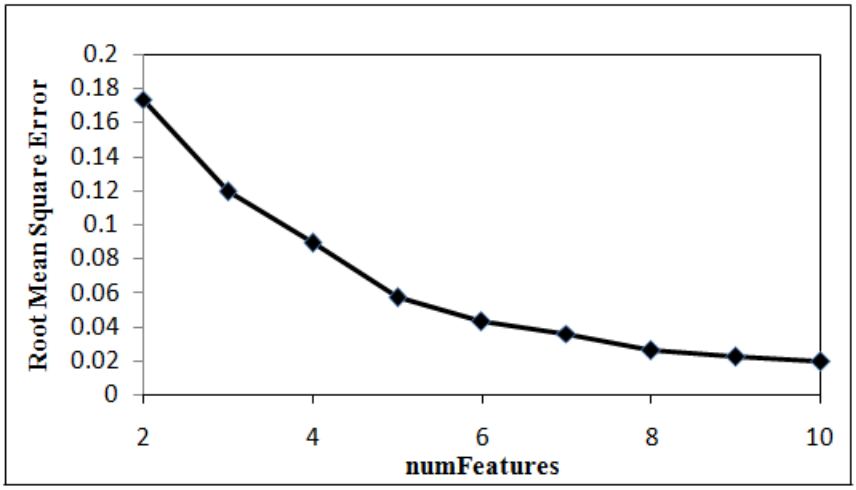

Figure 4. Relationship between numFeatures and RMSE

Table 3 Performance of Random forest with varying parameters

numtrees	numFeatures	RMSE	performance(%)
10	3	0.1299	96.6102
15	3	0.1241	96.6102
20	3	0.1192	96.6102
25	3	0.1208	96.6102
30	3	0.1294	96.6102
35	3	0.1332	96.6102
40	3	0.1269	96.6102
45	3	0.1266	96.6102
50	3	0.1338	96.6102
20	2	0.1734	91.5254
20	3	0.1192	96.6102
20	4	0.0893	98.3051
20	5	0.0572	100
20	6	0.0431	100
20	7	0.036	100
20	8	0.0261	100
20	9	0.0222	100
20	**10**	**0.0197**	**100**

Table 3 shows that Random forest model performs best when number of trees are 20, number of attributes used in random selection are 10 and depth of each tree in the random forest is 3.(i.e. numTrees=20 , numFeatures=10 and maxDepth=3) . The corresponding RMSE is about 0.0197 and performance of the model is about 100%.

Comparing performance

The performance of Random Forest model is compared with different decision tree classifiers. Weka Decision Tree(DT) Tool is for comparison, which include Hoeffding Tree, J48, LMT, Random Tree, REPTree and Decision Stump.

Following is brief description of each algorithm :

Hoeffding Tree induce model in the form of decision trees where each node contains a test on the attribute, each branch from a node corresponds to a possible outcome of the test and each leaf contains a class prediction. A decision tree is learned by recursively replacing leaves by test nodes, starting at the root.

J48 is Weka implementation of C4.5 algorithm(Quinlan 1993). Given a data set it generates a DT by recursive partitioning of data. The tree is grown using a depth-first strategy, i.e.., the algorithm calculates the information gain for all possible tests that can split the data set and selects a test that gives the greatest value. This process is repeated for each new node until a leaf node is reached.

LMTs(Logistic Model Trees; Landwehr et al.2005) basically consists of a standard decision tree structure with logistic regression functions at the leaves. LMT produces a single tree containing binary splits on numeric attributes, multiway splits on nominal ones, and logistic regression models at the leaves, and the algorithm ensures that only relevant attributes are included. It is a classification model with an associated supervised training algorithm that combines logistic regression (LR) and decision tree learning.

Random Tree model builds a tree considering k randomly chosen attributes at each node.

REPTree is a fast DT learner that builds a decision/regression tree using information gain/variance as the criterion to select the attribute to be tested at the node.

Decision Stump is a binary decision tree classifier consisting of a single node (based on one attribute) and two leaves.

Table. 4 shows the performance of seven DT methods. From this table it can be concluded that Random Forest gets the best performance about 100 % and Decision Stump obtains lowest performance of about 74.5763%.

Table 4 Comparative performance of Hoeffing Tree, J48, LMT, Random Tree, REPTree, Random Forest and Decision Stump

Method	Accuracy(%)
Hoeffding Tree	96.6102
J48	99.2170
LMT	98.3051
Random Tree	98.3051
REPTree	99.1715
Random Forest	100
Decision Stump	74.5763

V. CONCLUSION

In this paper, Data mining tree based techniques are applied to spectral classification and their performance are compared. Results show that ensemble learning performs better method than individual classifier. That is Random Forest Method is an effective method which is used for classification. Through tuning the parameter (number of trees, number of attributes used in random selection at each node), the best performance of random forest is achieved. As future work, more attributes that describe the astronomical objects, more kinds of these objects and large volume of data will be considered.

REFERENCES

[1]. Nong Ye. The Handbook of Data Mining. Lawrence Erlbaum Publishers, 2003.
[2]. J. Han and M. Kamber. Data Mining, Concepts and Techniques. Morgan Kaufmann Publishers, 2001.
[3]. Marios Kampakolou, Roberto Trotta, and Joseph Silk. Monolithic or hierarchical star formation? A new statistical analysis. Monthly Notices of the Royal Astronomical Society, 384(4):1414-1426, 2008.
[4]. Shaukat N. Goderya and Shawn M. Lolling. Morphological classification of galaxies using computer vision and artificial neural network: A computational scheme. Astrophysics and Space Science, 279(4):pp. 377-387, 2005.
[5]. Yogesh Wadadekar. Estimating photometric redshifts using support vector machines. Publications of the Astronomical Society of the Pacific, 117(827):pp. 79-85, 2005.
[6]. Leo Breiman, Random Forests, Machine Learning,45,5-32,2001.
[7]. Breiman, L. Bagging predictors. Machine Learning 26,2 (1996),123-140.
[8]. D.G. York, et al., and SDSS Collaboration. The Sloan Digital Sky Survey: Technical Summary. AJ,120:1579-1587, September 2000.
[9]. M.Hall, E. Frank, G. Homes, B. Pfahringer, P. Reutemann and I.H. Witten. The weka data mining software: An update. SIGKDD Explorations, 11(1):10-18, 2009.