Classification of Stars, Galaxies and Quasars

Vansh Teotia[a,b], Praveena Melepatte[a,c], Poojan Upadhyay[a,d], Saksham Chandna[a,e], Dr Aditi Panda[f]
[a]Research Intern, Spartificial Innovations Private Limited, Indira Colony, Kotkasim, 301702, Rajasthan, India
[b]Department of Physics , M6M5+639, University of Delhi, New Delhi, 110007, New Delhi, India
[c]Department of Physics, ARSD ,Ring Rd, Dhaula Kuan Enclave I, Dhaula Kuan, New Delhi, 110021, New Delhi, India
[d]Department of Physics, KSV university Sector 15, Gujarat, 382016, Gujarat, India [e]Department of Physics, Indian Institute of Science Education and Research, Bhauri Bypass Rd., Bhopal, 462066, Madhya Pradesh, India
[f]Department of CSE, J-15, Khandagiri Marg, Dharam Vihar, Jagamara, Bhubaneshwar, 751030, Odisha, India

Abstract:- The objective of this study was to use and compare multi- ple classifying models that can be used for classifying as- tronomical data and was tested upon data obtained from Sloan Digital Sky Survey: Data Release-16. Various Classi- fying models have been trained and tested by dividing the data into two parts- 80 per cent of the data was then used for training purposes and 20 per cent for testing. In order to achieve the task of classifying the tabular data consist- ing of spectroscopic and photometric parameters effectively, the study was not just limited to usage of indivial models. Stacking : the combination of multiple Classifying mod- els has also been implemented. Multiple stacking models were created for the same .Stacking models have on mul- tiple occasions proven to have higher evaluation metrics , thus having significantly better performance than any in- dividual classifier, proving that stacking is a better choice to classify data. certain Individual models such as Bag- ging , Hard Voting etc have been found to have comparable performance to that of Stacked Models. Box plots for in- diviual classes were also plotted to compare and determine the models that were capable in identifying a single class of stellar objects. The models from this study could be used as a reliable classification tool for a wide variety of astro- nomical purposes to accelerate the expansion of the sample sizes of stars, galaxies, and quasars.

***Keyword:-** Classification, Classifiers, Stars, Galaxies, Quasars.*

I. INTRODUCTION

The universe is composed of various objects of different shape, size and color. In order to understand the universe we first need to classify the objects that make it up. For centuries, astronomers have been observing and studying the sky to understand what kinds of objects were in the universe. From the ancient to the present day, humanity has created thousands of different astronomical catalogs. The goal of all of them is to collect observations of astronomical objects made with one or more instruments and to combine them into a unique homogeneous description. This enables anybody interested in the study of a given class of sources to compare their properties on an equal basis. Now, with more extensive and higher quality catalogs, we can perform this study in a better way. Stars, Quasars and Galaxies are the most commonly found objects in the universe[1]. A star is an astronomical object comprising a luminous spheroid of plasma held together by its gravity [wiki]. Galaxies are made of billions of such stars that revolve around a gravitation center of the black hole[1]. Quasars are quasi-stellar objects which emit electromagnetic radiation more potent than the luminosities of the galaxies, combined [1]. There have been numerous large scale survey catalogs that have been done to map the universe and the celestial objects present in it. The most important surveys are the Sloan Digital Sky Survey (SDSS), which commenced observations of the universe in 1998 [2]. There have been four major phases of this survey with multiple data releases(5'th phase going on in 2022). The information captured by the SDSS survey includes optical, spectroscopic, and photometric information, along with an array of other observations. Here we use the data of SDSS from Data release 14 which was made available in 2017 [sdss.org]. Our objective in this project is to compare multiple classifying machine learning models and determine the best classifier among them. Here we only use the spectroscopic and photometric information of the SDSS DR-14 dataset. The objectives are as follows:-

- To perform Exploratory Data Analysis on the SDSS dataset and to tidy the data. Also, to count plot & scatter plot the data for data visualization.
- To select multi-class classifier models like DecisionTree, LogisticRegression, Stacked, Boosted etc., which can be used to classify the data
- Split the data into 80/20 ratio for training and testing the data
- Use the training data to train the selected models and then test them.
- Evaluation Metrics such as Accuracy score, Precision score, Recall, F1 & Classification report are used to compare the tested models with the original test data. Thus identifying the best performing model
- Box Plotting the evaluation metrics with classification models to get a better visualization of the performance of the Models

A. Background

Many Large Surveys of the universe have been done over for a while. Amongst the most popular surveys which capture information about the celestial objects in the universe is the Sloan Digital Sky Survey (SDSS) [2]. Machine learning and Deep learning architectures are being continually designed and utilized in many large-scale astronomical surveys. Both supervised, and unsupervised Machine Learning is used for classification but supervised Machine Learning has proven to be superior for the task [7]. CNN & ANN, like Skynet & AstroNN, are designed and used to survey astronomical data collected by observatories like APO (Apache Point Observatory) [1]. The Javalambre Photometric Local

Universe Survey (J-PLUS) is also one of the surveys designed to observe several thousand square degrees in optical bands [5]. Some other databases used for such classification instead of SDSS are Gaia, WISE and UKIDSS [6]. There is a significant increase in research works related to stellar spectra detection and classification. Many researchers focused on star-quasar, galaxy-quasar or star-galaxy binary classification. Others focused on multi-class classification of stars, galaxies and quasars. In these works, various methods have been applied to automatically classify the heavenly bodies accurately [6]. Many authors used classical machine learning algorithms such as support vector machines (SVM), k-nearest neighbors (KNN), DecisionTree (DT), XGBoost, RandomForest (RF) etc. [1][2][3]. Others adopted deep learning techniques or developed their own novel solution. Other authors use data released from surveys like VEXAS and use an Ensemble of classifiers like KNN, ANN & CatBoost to get a better result for classifying stellar objects [4].

B. Data

The data used in this study are from the Sloan Digital Sky Survey (SDSS), which is a leading astronomical survey that has been working for more than 20 years to produce extremely precise and detailed imaging and maps of the universe. This public dataset, Data Release 14, is the second release of the fourth phase of the survey and had observations through July 2016[7]. It contains 18 variables with 10,000 total entries and no missing values and has been extracted from the SDSS public server using SQL query. 11 Variables (location of an object on the celestial sphere, the field/area of pixels in the image taken, info and specifications on the spectroscopy, optical fiber, etc.) have been removed since they are not contributing towards classification in any way[8]. The descriptions of the remaining 6 feature variables and 1 class variable (Camera; Measures Of Flux And Magnitude; Redshifts, The Photometric Camera and the CCDs; Understanding SDSS Imaging Data; Understanding the Imaging Data), their first 10 entries, and their statistics are shown in Tables below:

U	*The intensity of light (flux) with a wavelength of* $3551\mathring{A}$ *emitted by the object*
G	*The intensity of light (flux) with a wavelength of* $4686\mathring{A}$ *emitted by the object*
R	*The intensity of light (flux) with a wavelength of* $6166\mathring{A}$ *emitted by the object*
I	*The intensity of light (flux) with a wavelength of* $7480\mathring{A}$ *emitted by the object*
Z	*The intensity of light (flux) with a wavelength of* $8932\mathring{A}$ *emitted by the object*
Red Shift	*Measurement of how fast the object is moving away relative to Earth. A result of Doppler's Effect: light emitted from an object moving away increases in wavelength and shifts to the red end of the light spectrum*
Class	*Classification of the object as star, galaxy and Quasar*

Table 1: Description Of Variables

C. Exploratory Data Analysis

Exploratory data analysis is the initial process to analyse the data set through data statistics and graph plots , wherever applicable. For EDA the libraries of pandas, seaborn, were used. The number of variables , their types, entries, and missing or null values were examined. Then, the mean, standard deviation, minimum, lower quartile, median, upper quantiles, and maximum of the feature variables were calculated and organised. Then , a count plot and scatter plot of the dataset were plotted.

	U	**G**	**R**	**I**	**Z**	**Redshift**
Mean	18.619355	17.371931	16.840963	16.583579	16.422833	0.143726
Std	0.828656	0.945457	1.067764	1.141805	1.203188	0.388774
Min	12.988970	12.799550	12.431600	11.947210	11.610410	-0.004136
25%	18.178035	16.815100	16.173333	15.853705	15.618285	0.000081
50%	18.853095	17.495135	16.858700	16.554985	16.389945	0.042591
75%	19.259232	18.010145	17.512675	17.258550	17.141447	0.092579
Max	19.599900	19.918970	24.802040	28.179630	22.833060	5.353854

Table 2: Dataset Statistics

	U	G	R	I	Z	Redshift	Class
1	19.4706	17.0424	15.9469	15.5034	15.2253	-8.96E-06	STAR
2	18.6628	17.2149	16.6737	16.4892	16.3915	-5.49E-05	STAR
3	19.3829	18.1916	17.4742	17.0873	16.8012	0.1231112	GALAXY
4	17.7653	16.6027	16.1611	15.9823	15.9043	-0.0001106	STAR
5	17.5502	16.2634	16.4386	16.5549	16.6132	0.0005903	STAR
6	19.4313	18.4679	18.1645	18.0147	18.0415	0.0003146	STAR
7	19.3832	17.8899	17.1053	16.6639	16.3695	0.1002423	GALAXY
8	18.9799	17.8449	17.3802	17.2067	17.0707	0.0003148	STAR
9	17.9061	16.9717	16.6754	16.5377	16.4759	8.91E-05	STAR
10	18.6724	17.7137	17.4936	17.2828	17.2264	0.0405081	GALAXY

Table 3: First 10 entries of the dataset

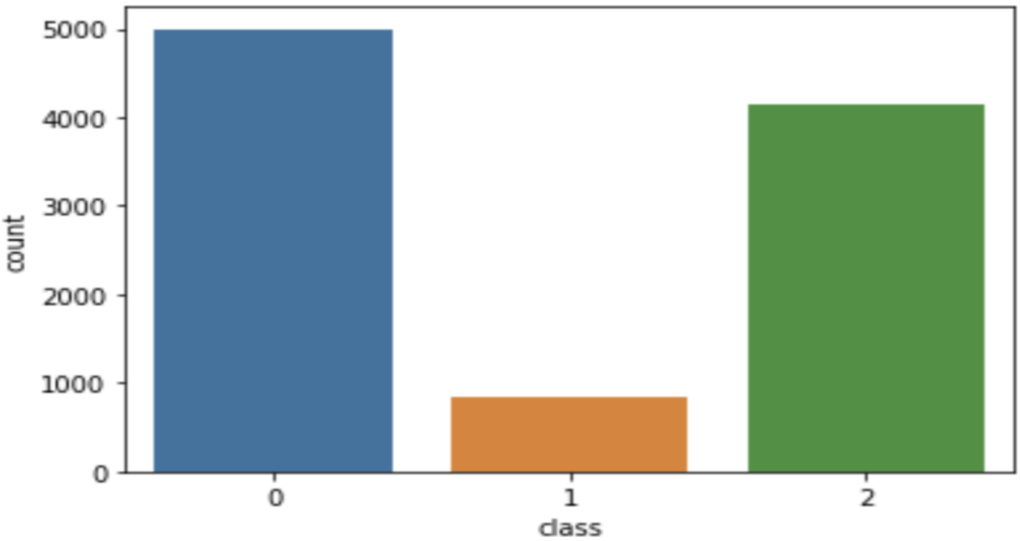


Fig. 1: Quantitative distribution of class-labels in data

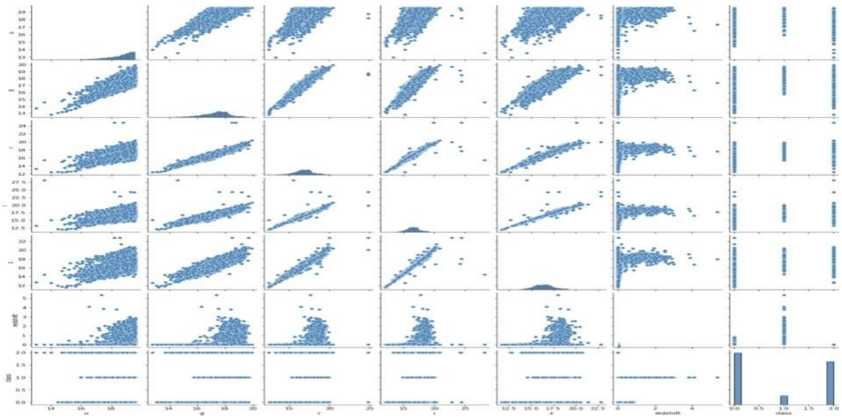


Fig. 2: Pair plots between features of dataset.

II. REVIEW AND BACKGROUND INFORMATION

Various algorithms and machine learning models are possible to achieve the goal of classifying the data from SDSS(Sloan Digital Sky Survey). In this Study we have focused on Comparing results from Individual models and stacked models to determine the best possible method.

A. Background

The classification of celestial objects is of great significance in the fields of astronomy. Its most direct benefit is providing the means to gather data samples of stars, galaxies, and quasars. Particularly for quasars, despite how important they are to a wide range of astronomy studies and research, their sample sizes are still in the relative minority class[1][2]. In order to achieve the increase in sample size , fast and reliable classifying models are crucial. Modern day telescopes have the capability to gather large amounts of data , leading to the need of fast and reliable classification models being emphasised even further[1]. Past Studies have prove that both supervised models have been shown to have higher accuracies than unsupervised models, which have been shown to be more efficient in identifying unknown objects[3][4]. Supervised models such as Support Vector, Random forest have been shown to classify the data with accuracies of up to 90 percent in studies to solely identify quasars or to classify stars , galaxies and quasars[5]. Decision tree classifier has been shown to be considerably effective than most other supervised classifiers such as Logistic Regression[2].

III. PROPOSED SOLUTIONS

The solution towards the class task devised by us has been described below

A. Classifiers

We've compared forty three classifiers in this project, including boosting, bagging and ensemble classifiers. Naive Bayes (NB) is a probabilistic classifier that is based on the Bayes' theorem with conditional dependence. Three variants of the classifier, which varied based on their assumptions on the distribution of likelihood of features, were used in this project: Gaussian Naive Bayes (GNB), Multinomial Naive Bayes (MNB) and Bernoulli Naive Bayes (BNB) [9]. Linear Discriminant Analy- sis (LDA) and Quadratic Discriminant Analysis (QDA) are classifiers that use linear and non-linear surfaces to separate the classes [9]. K-nearest neighbor (KNN) is an algorithm that classifies test data points by calculating the distance between the training data points and test data points, and then finding the probability for the point belonging to K nearest neighbors [10]. The value of K can be changed (n neighbors = 5, default), and the algorithm used to compute the nearest neighbor can also be varied. We have used three KNN classifiers with three different algorithms for finding the K nearest neighbors: Brute force (KNNB), Ball tree (KNN) and KD tree (KNNKD) [10]. Nearest Centroid (NC) is a classifier who assigns the test data to the class whose centroid is nearest to the test data point [9]. The Ridge Classifier (RC) is based on the ridge regression method that uses linear least squares with l2 regularization. Ridge CV (RCV), is a ridge classifier with built in cross-validation. Stochastic Gradient Descent classifier (SGD) is a linear classifier optimized by SGD, which computes the gradient of the loss per sample [11]. We used 3 different variations of the classifier by changing the penalty parameter, as follows: SGD l1 (penalty = 'l1'), SGD l2 (penalty = 'l2', which is the default) and SGD elastic-net (penalty = 'elasticnet'). Support Vector Machines (SVM) are classifiers that use thresholds with soft margins to classify the training data into clusters and subsequently classify the test data based on their position relative to the threshold. We used two SVM models with different kernels, which help in non-linear classification: PSVM (kernel = 'polynomial) and RBFSVM (kernel = 'rbf', radial basis function). SVM uses the one-versus-one approach for multi-class classifications as a default, since they are inherently a binary classifier [9]. Nu-Support Vector Machine (Nu-SVM), is similar to SVM, but uses a different regularization parameter Nu and Linear Support Vector Machine (LSVM) is similar to a SVM with a linear kernel and works on the liblinear library instead of libsvm [9]. We used LSVM with 2 penalty parameters: LSVM1 (penalty = 'l1') and LSVM2 (penalty = 'l2', which is the default) [9]. Perceptron (PTN), is a linear classifier algorithm that acts as an artificial neuron while Multilayer Perceptron (MLP) is an artificial neural network that works non-linearly [12]. Passive-Aggressive (PA) is also a linear classifier that doesn't use a learning rate but uses a regularization parameter (c = 1.0, default). Calibrated CV (CCV) uses cross validation to determine the parameters and also to calibrate a classifier (estimator = LinearSVC, default) [9]. Logistic regression (LR) is a linear clas- sifier, in which the probabilities of an outcome are modeled using a logistic function. In the case of models that are binary classifiers, the multi-class dataset is split into binary class subsets. The default setting for binary classifiers in scikit-learn library for handling multi-class dataset is, in general, the One-vs-Rest approach. Decision Trees (DT) is a classifier that has a tree-like hierarchical structure and uses different functions (criterion = 'gini', default) to find the optimum split. Extra Trees (ET) and Random Forest (RF) are collections of decision trees with ET using a random split and RF using an optimum split and subsampling the data [13]. We used 2 models of RF with different functions to calculate the optimum split: RFE (criterion = 'entropy') and RF (criterion = 'gini') [9]. Both are extensions of Bagging classifier, which is an ensemble algorithm which uses multiple versions of the base estimators (estimator = DecisionTreeClassifier, default) on data subsets, run them in parallel and then combine their individual predictions to achieve the final outcome. Boosting is an ensemble learning technique that builds a number of weak classifiers sequentially to produce a strong classifier [9]. We have used numerous boosting classifiers such as Adaptive Booster (AdaB), which identifies misclassified data points and adjusts their weights so as to minimize the error, and feeds it to the next sequential classifier; Gradient Booster (GB), a boosting algorithm that works on reducing the residuals of the predictors of the previous classifier; Extreme Gradient Booster (XGB), which is a computation- ally efficient implementation of GB; Light Gradient Boosting Machine (LightGBM), which is similar to XGBoost but different in that it chooses a leaf that will lead to a higher reduction in loss, and continues the tree (leaf-wise growth), CatBoost (CB), which is an open source library with gradient boosting framework and HistGradBoost (HGB), which is a Histogram-based Gradient boosting classifier. Inorder to increase the accuracy of classifiers, we stacked a few of them and made five stacked models:

- Stacked model-1 with DT, LR, KNN and SVM (RBF)
- Stacked model-2 with KNN, SVM (RBF) and AdaBoost, which were some of the classifiers with lower performance
- Stacked model-3 with LR, KNN and SVM (RBF)
- Stacked model-4 with MLP, RF and SVM (RBF)
- Stacked model-5 with SVM (RBF), KNN and PA

The stacked models showed higher performance than any individual models. Voting classifier is also an ensemble learning method, and is of two types: Hard voting (HV), where the voting is calculated based on the predicted output and Soft Voting (SV), where it is based on the predicted probability of the output class. We built one soft voting and two hard voting classifiers as such:

➢ *Soft Voting:*

(i) QDA, (ii) Nu - SVM, (iii) RBFSVM, (iv) PSVM, (v) DTC, (vi) RF, (vii) XGBC, (viii) BC, (ix) MLP, (x) ETC, (xi) GNB.

- **Hard Voting 1:** (i)RFE, (ii) XGBoost.
- **Hard Voting 2:** (i) QDA, (ii) Nu - SVM, (iii) RBFSVM, (iv) PSVM, (v) DTC, (vi) RF, (vii) XGBC, (viii) BC, (ix) MLP, (x) ETC, (xi) GNB.

B. Evaluation Metrics

We have used four Evaluation metrics for calculating and comparing the performance of all the classifier models: Precision, Recall, Accuracy and F1 Score. We also calculated Precision, Recall and F1 score of each class (Galaxy, Stars and quasars) for each of the classifier models using the classification report function in the sci-kit learn library. To calculate these values, a confusion matrix is required, such as this:

Actual	**Predicted**
	Negative Positive
Negative	True Negative False Positive
Positive	False Negative True Positive

Table 3: TP, TN, FP, & FN explanation table

- **True Positive (TP) -** The actual value is positive and is predicted to be positive.
- **True Negative (TN) -** The actual value is negative and is predicted to be negative.
- **False Positive (FP) -** The actual value is negative but is predicted to be positive.
- **False Negative (FN) -** The actual value is positive but is predicted to be negative

IV. EXPERIMENTAL RESULTS

To compute the performance of the classifier models we used evaluation metrics such as accuracy, recall, precision and F1 score. We also used classification report from sklearn to obtain class-wise evaluations for each model.

A. Evaluation Metrics of the Classifiers used

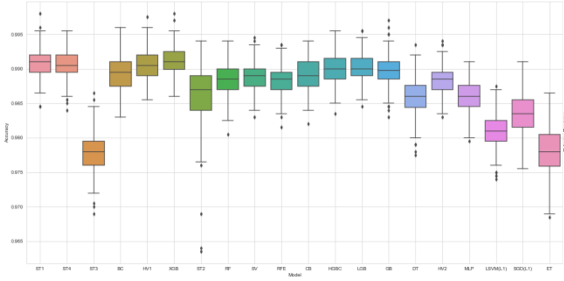
Out of the 43 models used , the evaluation metrics of the best 20 have been listed below:

S.no	Classifier	Accuracy	Precision	Recall	F1 score
1	Stacking Model-5	0.9785	0.9787	0.9785	0.9784
2	Extra Trees	0.9790	0.9790	0.9790	0.9789
3	SGD l1	0.9790	0.9790	0.9790	0.9789
4	LSVM(L1)	0.9800	0.9799	0.9800	0.9789
5	Multi Layer Perception	0.9825	0.9825	0.9825	0.9824
6	Hard Voting-2	0.9875	0.9875	0.9875	0.9874
7	Decision Tree	0.9880	0.9879	0.9880	0.9879
8	Grad Boost	0.9885	0.9879	0.9885	0.9884
9	LightGBM	0.9885	0.9884	0.9885	0.9884
10	HistGradBoost	0.9890	0.9890	0.9890	0.9889
11	CatBoost	0.9890	0.9889	0.9890	0.9889
12	RF entropy	0.9890	0.9890	0.9890	0.9889
13	Soft Voting	0.9890	0.9890	0.9890	0.9889
14	RF gini	0.9895	0.9894	0.9895	0.9894
15	Stacked Model-2	0.9895	0.9895	0.9895	0.9894
16	XGBoost	0.9895	0.9895	0.9895	0.9894
17	Hard Voting-1	0.9900	0.9900	0.9900	0.9899
18	Stacked Model-3	0.9900	0.9900	0.9900	0.9900
19	Stacking Model-4	0.9903	0.9903	0.9903	0.9903
20	Stacking Model-1	0.9910	0.9910	0.9910	0.9909

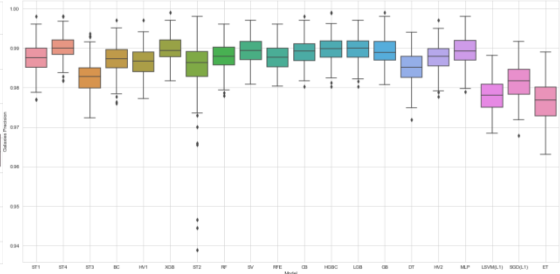
Table 4: Best 20 classifiers

B. Corresponding Box-Plots

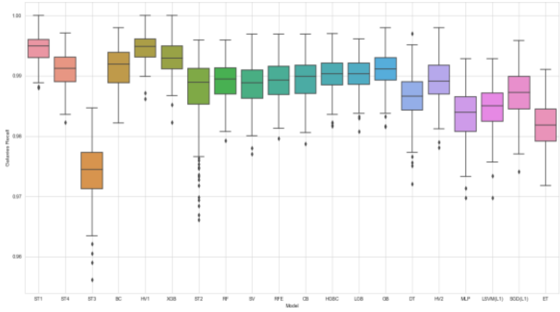
Box-plots are made for 20 models which have the highest accuracy of the total 43:



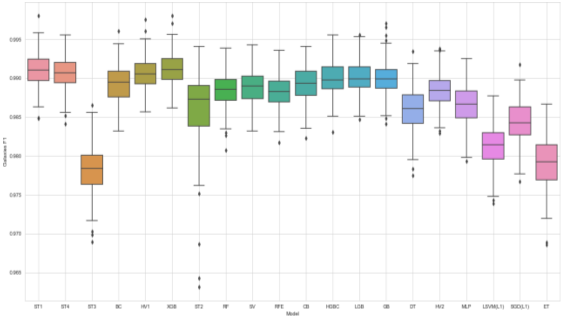
(a) Accuracy Box-Plot



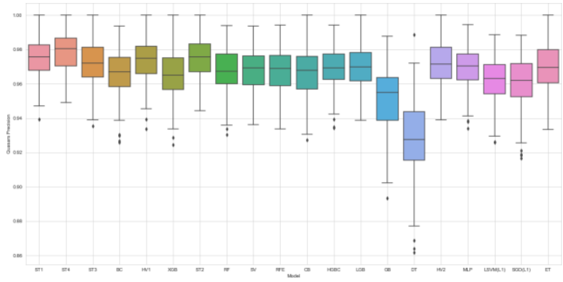
(b) Galaxies Precision Box-Plot



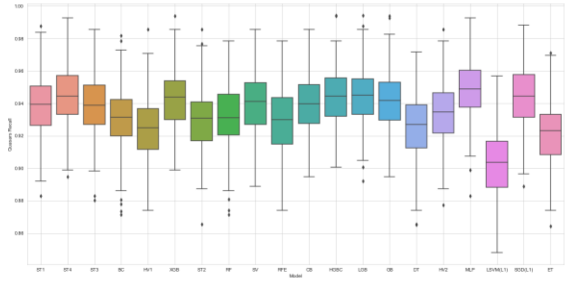
(c) Galaxies Recall Box-Plot



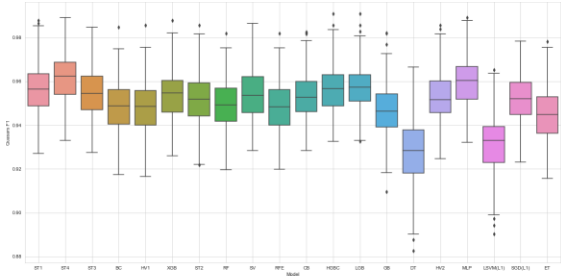
(d) Galaxies F1-score Box-Plot



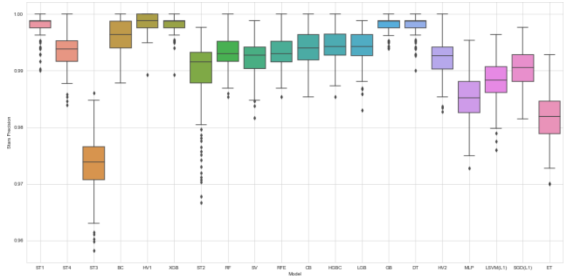
(e) Quasars Precision Box-Plot



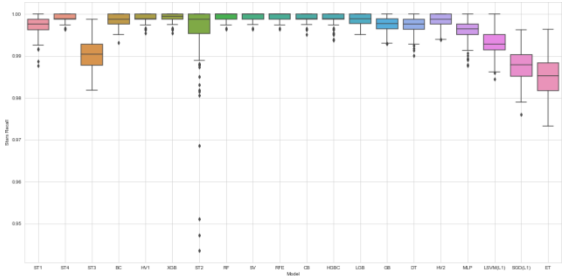
(f) Quasars Recall Box-Plot



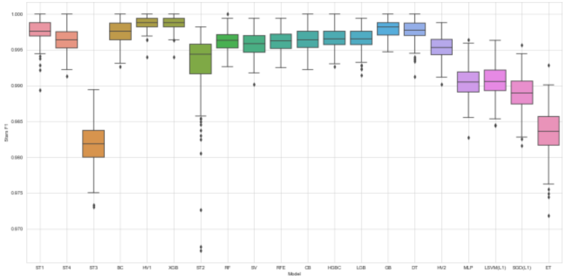
(g) Quasars F1-score Box-Plot



(h) Stars Precision Box-Plot



(i) Stars Recall Box-Plot



(j) Stars F1-score Box-Plot

Fig. 3: Box-Plots

V. CONCLUSION AND FUTURE SCOPE

After utilizing several machine learning models for subsequent classification of astronomical dataset from the Sloan Digital Sky Survey, Data Release - 14 (SDSS - DR14) into class-labels: "GALAXY", "QSO" & "STAR", a thorough analysis of the evaluation metrics: Accuracy, Precision (per class label), Recall (per class label), & F1-score (per class label) is performed and results are analyzed. The data consists of information on 17 feature variables and 1 class variable of 10,000 astronomical objects in total, and of the 17 only 15 feature variables are retained as an input for classification models. An initial EDA of the dataset is performed in order to prepare the data for the subsequent classification task, wherein a 80/20 split of given data into training and testing data respectively is done. The training data is used to train independent Machine Learning models, directly accessible from the sklearn Python library, while keeping the hyper-parameters to a default. Some ensemble based algorithms, such as voting (hard and soft) are also analyzed with their ensemble members being derived based on an optimally performing voting algorithm evident in previous research. Stacking algorithms are constructed from poorly performing models and high performing algorithms to observe an increase in subsequent performance. In total, 43 machine learning models are fitted to the training data, and their corresponding evaluation metrics are found using the testing data. The results are tabulated above in an increasing order of accuracy of models. As was expected, stacked models tended to perform better based on almost all evaluation metrics. Soft and hard voting algorithms also managed to score better than most individual models tested. For further detailed analysis of metric variation with repeated fitting, box-plots of all evaluation metrics are developed for the top 20 machine learning models in the table, and are presented above. Given that the data is unbalanced in the frequency of class labels, the plots of precision, recall and F1-score are separately analyzed for each class label. In case of "QSO" class label, the evaluation metric box-plots are observed to be more spread out than for other class labels, a result that can be attributed to the fact the number of objects labeled "QSO" are significantly less than those labeled "GALAXY" and "STAR", both are which are almost equally distributed. The combined accuracy plot identifies the Stacked Model 1, Stacked Model 4 and XGBoost as the best performing models in terms of accuracy due to their high mean accuracy values and small spread about the mean. For "GALAXIES", Stacked Model 1, Hard voting model 1 and XGBoost model seem to dominate with respect to recall and F1-score, but individual models seem to give better precision results than ensemble-based models. For "QSO", we note due to a lower frequency of data, almost all models perform equivalently across all evaluation metrics, and therefore, a decision regarding the best models for this class-labels can't be reached due to unavailability of sufficient data. For "STARS", Stacked Model 1, Hard voting model 1, XGBoost model, Gradient Boosting model and Decision tree models seem to outperform in precision and F1-score, whereas almost all models perform equivalently well when it comes to recall. In case of "STARS", a generic high performance is observed when compared to other class-labels, implying that the objects belonging to this class label are more easily classified with several machine learning models. The hierarchy of model performance based on each class-label is sufficiently presented, and with the ever-increasing data incoming from various ongoing astronomical surveys, machine learning algorithms specialized to classifying each class-label can be identified using the models already narrowed down above. Since the astronomical survey data is publicly released, a basic set of specialized high-performing models can be made readily available for public-access in order to avoid long hours associated with finding the right fit for the data set, considering the numerous machine learning models present today. Further optimization based on computational time and complexity can also be performed based on the requirements of individual projects and the available resources.

REFERENCES

[1.] Sabeesh Ethiraj, Bharath Kumar Bolla, "CLASSIFICATION OF QUASARS, GALAXIES, AND STARS USING MULTI-MODAL DEEP LEARNING", Cornell university arXiv, May 2022.

[2.] Zhuliang Qi, "Stellar Classification by Machine Learning", SHS web of conferences, August 2022.

[3.] Jordi Sabat´es de la Huerta, "Classifying astronomical sources with machine learning", Dipòsit Digital de la Universitat de Barcelona, January 2022.

[4.] V. Khramtsov, C. Spiniello, A. Agnello, A. Sergeyev, "VEXAS: VISTA EXtension to Auxiliary Surveys", Astronomy & Astrophysics, March 2021.

[5.] Cunshi Wang, Yu Bai , C. López-Sanjuan, et al.,"J-PLUS: Support vector machine applied to STAR-GALAXY-QSO classification", Astronomy & Astrophysics, December 2021.

[6.] Micha-l Wierzbiński, Pawe-l P-lawiak, Mohamed Hammad, U. Rajendra Acharya, "Development of accurate classification of heavenly bodies using novel machine learning techniques", Methodologies and Application, March 2021.

[7.] Yulun Wu, "MACHINE LEARNING CLASSIFICATION OF STARS, GALAXIES, AND QUASARS", MATTER: International Journal of Science and Technology, 2021.

[8.] Blanton, M. R., Bershady, M. A., Abolfathi, B., Albareti, F. D., Allende Prieto, C., Almeida, A., ... Zou, H. (2017, July). Sloan Digital Sky Survey IV: Mapping the Milky Way, Nearby Galaxies, and the Distant Universe. NASA/ADS. https://ui.adsabs.harvard.edu/abs/ 2017AJ....154...28B/ abstract.

[9.] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. The Journal of machine Learning research, 12, 2825-2830.

[10.] Christopher, A. (2021, February 3). K-Nearest Neighbor. Medium. Retrieved December 14, 2022, from https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

[11.] Introduction to SGD classifier - Michael Fuchs Python. MFuchs. (2019, November 11). Retrieved December 14, 2022, from https://michael-fuchs-python.netlify.app/2019/11/11/introduction-to-sgd-classifier/ [12] Hange, A. (2021, April 18). Flux prediction using single-layer perceptron and Multilayer Perceptron. Medium. Retrieved December 14, 2022, from https://medium.com/nerd-for-tech/flux-prediction-using-single-layer-perceptron-and- multilayer-perceptron-cf82c1341c33

[12.] Aznar, P. (2020, June 17). What is the difference between extra trees and random forest?

[13.] Quantdare. Quantdare. Retrieved December 14, 2022, from https://quantdare.com/what-is-the-difference-between-extra-trees-and-random-forest/

[14.] Wang, C. ; Bai, Y. ; López-Sanjuan, C. ; Yuan, H. et al. (2022, March). J-PLUS: Support vector machine applied to STAR-GALAXY-QSO classification. Astronomy & Astrophysics, Volume 659, id.A144, 23 pp. https://arxiv.org/pdf/2106.12787.pdf

[15.] Mahalakshmi, G. S., Swadesh, B., Aswin, R. R. V., Sendhilkumar, S., Swaminathan, A., & Surendran, S. (2022, August). Classification and Feature Prediction of Star, Galaxies, Quasars, and Galaxy Morphologies Using Machine Learning. Research Square. https://assets.researchsquare.com/files/rs-1885343/v1 covered.pdf?c=1661794633

[16.] O. Clarke, A. M. M. Scaife, R. Greenhalgh and V. Griguta: Identifying galaxies, quasars, and stars with machine learning: A new catalogue of classifications for 111 million SDSS sources without spectra, Astronomy & Astrophysics, Volume 639, id. A84, 29 pp., July 2020. https://doi.org/10.1051/0004- 6361/20193670.

[17.] Nicholas M. Ball, Robert J. Brunner, Adam D. Myers, David Tcheng: Robust Machine Learning Applied to Astronomical Data Sets. I. Star-Galaxy Classification of the Sloan Digital Sky Survey DR3 Using Decision Trees, The Astrophysics Journal 650 497, October 2006. DOI:10.1086/507440.

[18.] Wierzbiński, M., P-lawiak, P., Hammad, M. et al. : Development of accurate classifica- tion of heavenly bodies using novel machine learning techniques, Soft Comput 25, 7213–7228 (2021). https://doi.org/10.1007/s00500-021-05687-4

[19.] S. Chaini, A. Bagul, A. Deshpande, R. Gondkar, K. Sharma, M Vivek, A. Kembhavi: Photo- metric identification of compact galaxies, stars, and quasars using multiple neural networks, Monthly Notices of the Royal Astronomical Society, Volume 518, Issue 2, January 2023, Pages 3123–3136. https://doi.org/10.1093/mnras/stac3336