

ASTROVISION

A PROJECT REPORT

Submitted by

Pranjal Roy (19BAI10008)
Shanzeh Batool (19BAI10017)
Karan Jain (19BAI10095)
Shalini Das (19BAI10139)

*in partial fulfillment for the award of the degree
of*

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Specialization in

Artificial Intelligence and Machine Learning



SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

**KOTHRI KALAN, SEHORE
MADHYA PRADESH - 466114**

MAY 2021

**VIT BHOPAL UNIVERSITY, KOTHRI KALAN, SEHORE
MADHYA PRADESH – 466114**

BONAFIDE CERTIFICATE

Certified that this project report titled “**ASTROVISION**” is the bonafide work of “**Pranjal Roy (19BAI10008), Shanzeh Batool (19BAI10017), Karan Jain (19BAI10095), Shalini Das (19BAI10139)**” who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here does not form part of any other project / research work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR

Dr. S Sountharajan, Senior Assistant Professor
School of AI & ML division
VIT BHOPAL UNIVERSITY

PROJECT GUIDE

Dr. Pon Harshavardhanan, Associate Professor
School of Computer Science Engineering
VIT BHOPAL UNIVERSITY

The Project Exhibition II Examination is held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

First and foremost we would like to thank the Lord Almighty for his presence and immense blessings throughout the project work.

We wish to express our heartfelt gratitude to Dr. S Sountharajan, Head of the Department, School of Computing Science and Engineering for much of his valuable support and encouragement in carrying out this work.

We would like to thank our internal guide Dr. Pon Harshavardhanan, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

We would like to thank all the technical and teaching staff of the School of Computing Science and Engineering, who extended directly or indirectly all support.

Last, but not the least, we are deeply indebted to our parents who have been the greatest support while we worked day and night for the project to make it a success.

LIST OF ABBREVIATIONS

S.NO	ABBREVIATION	INDICATION
1.	AI	Artificial Intelligence
2.	ML	Machine Learning
3.	PhotoRApToR	Photometric Research Application To Redshift
4.	SDSS	Sloan Digital Sky Survey
5.	U	Ultraviolet
6.	G	Green
7.	R	Red
8.	I	Near Infrared
9.	Z	Infrared

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
3.3	Hubble's Law : Relationship between distance of Galaxy and it's Redshift	10
4.6	Decision Tree upto layer 3	13
4.7	u-g vs r-i graph. Redshift color scale on the right.	13
5.3.1	Confusion matrix without normalisation	16
5.3.2	Confusion matrix with normalisation	17

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
5.2	Performance of various models	15

ABSTRACT

More and more astronomers are using AI as a powerful discovery tool to offer rich and complex data, classify galaxies, sift through data for signals, find pulsar stars, identify unusual exoplanets, among other things. We will go through all the existing algorithms being used in this particular field and their shortcomings. Our project has two proposed models: Redshift Regressor and Galaxy Classifier. Redshift Regressor uses the Decision Trees algorithm while Galaxy Classifier uses the Random Forest algorithm. These algorithms are supervised machine learning that shall help in making the predictions. The paper explains in detail the functioning and analysis of each module in our system as well as the implementation of each one. The performance analysis gives the accuracy rate of our model and how it is an improvement over the other existing ones. Finally, we conclude with a few future enhancements which if implemented, can make work as an aid for astronomical predictions.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	List of Abbreviations	iv
	List of Figures	v
	List of Tables	vi
	Abstract	vii
1	INTRODUCTION	
	1.1 Introduction	1
	1.2 Motivation for the work	2
	1.3 About Introduction to the project including techniques	2
	1.5 Problem Statement	2
	1.6 Objective of the work	3
	1.7 Organization of the thesis	3
	1.8 Summary	3
2	LITERATURE SURVEY	
	2.1 Introduction	4
	2.2 Core area of the project	4
	2.3 Existing Algorithms	5
	2.3.1 Deep Neural Networks	5
	2.3.2 Multilayer Neural Network	5
	2.3.3 K Nearest Neighbour	5
	2.3.4 Galaxy Image Classification and Citizen Science	5
	2.3.5 Feature Extraction Plus a Classifier	6
	2.3.6 Convolutional Neural Networks	6
	2.4 Other method used in the project	7
	2.5 Research issues/observations from literature Survey	7
	2.6 Summary	8

3	SYSTEM ANALYSIS <ul style="list-style-type: none"> 3.1 Introduction 3.2 Disadvantages/Limitations in the existing system 3.3 Proposed System <ul style="list-style-type: none"> 3.3.1 Redshift Regressor 3.3.2 Galaxy Classifier 3.4 Summary 	8 8 9 11
4	SYSTEM DESIGN AND IMPLEMENTATION <ul style="list-style-type: none"> 4.1 Introduction <ul style="list-style-type: none"> Redshift Regressor 4.2 Importing Libraries 4.3 Loading the Dataset and Generate Feature Targets 4.4 Implementation of the function median_diff 4.5 Splitting of data & Validation of decision tree model 4.6 Plot the decision tree 4.7 Defining color indexes and plotting the redshift graph <ul style="list-style-type: none"> Galaxy Classifier 4.8 Importing the libraries 4.9 Splitting the data 4.10 Generate feature target 4.11 Training the decision tree classifier 4.12 Work on the decision tree classifier 4.13 Random forest classifier 4.14 Summary 	11 11 12 12 13 13 14 14 14 14 14 14
5	PERFORMANCE ANALYSIS <ul style="list-style-type: none"> 5.1 Introduction 5.2 Performance Measures (Table/text) 5.3 Performance Analysis(Graphs/Charts) 5.4 Summary 	15 15 16 18
6	FUTURE ENHANCEMENT AND CONCLUSION	

	6.1 Introduction	18
	6.2 Limitation/Constraints of the System	18
	6.3 Future Enhancements	19
	6.4 Conclusion	19
	References	20

INTRODUCTION

1.1 Introduction

Redshift Regressor: In the current standard cosmological model, the universe began nearly 14 billion years ago, in an explosive event commonly known as the Big Bang. Since then, the very fabric of space has been expanding, so that distant galaxies appear to be moving away from us at very fast speeds. The uniformity of this expansion means that there is a relationship between the distance to a galaxy, and the speed with which it appears to be receding from us. This recession speed leads to a shift in the frequency of photons, very similar to the audio doppler shift that can be heard when a car blaring its horn passes by. If a galaxy were moving toward us, its light would be shifted to higher frequencies, or blue-shifted. Because the universe is expanding away from us, distant galaxies appear to be red-shifted: their photons are shifted to lower frequencies.

Galaxy Classifier: While thinking about space, when we talk about galaxies we observe there are different types of galaxies that are needed to be classified for different purposes. We come across different types of galaxies like Elliptical galaxies, Spiral Galaxies Lenticular galaxies, and Irregular galaxies. So, we basically try to build a model that is able to classify the Galaxies. Elliptical and lenticular galaxies are collectively known as early-type galaxies, because of their simple appearance. Spiral galaxies are known as late-type galaxies, because of their complex structure. Elliptical galaxies have a smooth, spheroidal appearance with little internal structure. They're dominated by a spheroidal bulge and have no obvious thin disk. Most obviously spiral galaxies all show spiral arms, though, when viewed edge on the arms may not be visible. And finally, there is a class of irregular galaxies that don't fit any of the previous described types.

1.2 Motivation for the work

Classification or regression analysis based purely on observational properties of stars and galaxies is an empirical result that can survive regardless of changes in the theoretical interpretations. The amount of information we need to process if we treat each star or galaxy from a large survey as an individual is so overwhelming as to be nearly useless. Machine learning allows us to draw meaningful conclusions from data by abstracting it into categories. Most importantly, classification can lead to a deeper physical understanding of the data.

1.3 About Introduction to the project including techniques

Redshift Regressor: We train a decision tree regressor that builds a model mapping the features, in this case the four colors, to the results, in this case the red shifts measured from spectroscopy.

Galaxy Classifier: We try to build a model that classifies the galaxies. The process of classification involves sorting through the photographs, identifying key features of each galaxy.

For this, we first use decision trees, and then to improve the classification and reduce overfitting we use the random forest classifier.

1.5 Problem Statement

Redshift Regressor: Calculate redshifts of distant galaxies using decision trees for regression.

Galaxy Classifier: Classifying the galaxies into three main categories:

- (i) Elliptical galaxies,
- (ii) Spiral galaxies,
- (iii) Irregular galaxies

1.6 Objective of the work

We then train a decision tree regressor that builds a model mapping the features, in this case the four colors. To the results, in this case the red shifts measured from spectroscopy.

To classify the galaxies based on their shapes and their morphological descriptions like their rate of formation of stars, brightness, etc. into Elliptical, Spiral, and irregular galaxies.

1.7 Organization of the thesis

This thesis consists of a total 6 chapters, followed by the appendices and references. The current chapter gives the introduction about our thesis. Chapter 2 presents the background of the various algorithms that can be used to develop this model, algorithms that we have used in developing the model and other techniques that have been used to generate this model. In Chapter 3, we discuss the disadvantages and the limitations of the existing models, and what our model states, followed by summary. Chapter 4 presents a detailed explanation of design and implementation of the ML model and Chapter 5 gives a pictorial and graphical representation of the performance of the model. In Chapter 6, we discuss how we enhance the existing model followed by conclusion.

1.8 Summary

Redshift Regressor: We build an ML model composed of a decision tree regressor. The model maps the colours of lights emitted from galaxies to their respective spectroscopic redshifts which can further be used to compute the distance of the galaxy by using the Hubble's Law.

Galaxy Classifier: We observe that there are three main types of galaxies and they are (i) Elliptical galaxies, (ii) Spiral galaxies, (iii) Irregular galaxies

To build a model that will be able to classify galaxies into these types of galaxies we take the help of Machine Learning algorithms. We first make use of decision trees and later to improve the accuracy of the model we use Random forest classifier.

LITERATURE SURVEY

2.1 Introduction

There have been several ways to calculate redshifts and classify galaxies using computers and models that have saved time and increased accuracy. By conducting a scientific literature review, we identify and categorize the currently available and the most commonly used methods to calculate redshifts and classify galaxies.

2.2 Core area of the project

Redshift Regressor: In this project we are trying to estimate redshifts using the decision tree regression Model which is based on supervised machine learning. The distance between galaxies can be calculated using redshifts. Hubble's laws states that the velocity of recession between our galaxy and the other galaxies are directly proportional to the distance between them. The velocity of the galaxy which is also known as the redshift is directly proportional to its distance.

Galaxy Classifier: The major task is to identify the type of galaxy a picture from a data set contains. Broadly there are 3 major categories of classification:

- A. The Elliptical Galaxies
- B. The Spiral Galaxies
- C. The Merged Galaxies

2.3 Existing Algorithms

Redshift Regressor:

2.3.1 Deep Neural Networks:

The standard techniques estimate redshifts using post-processed features, such as magnitudes and colours, which are extracted from the galaxy images and are deemed to be salient by the user. This method removes the user from the photometric redshift estimation pipeline. However, we do note that Deep Neural Networks require many orders of magnitude more computing resources than standard machine learning architectures.

2.3.2 Multilayer Neural Network:

Photometric redshifts (photo-z) are crucial to the scientific exploitation of modern panchromatic digital surveys. In this PhotoRApToR (Photometric Research Application To Redshift): a Java/C++-based desktop application capable of solving non-linear regression and multivariate classification problems, in particular, specialized for photo-z estimation. It embeds a machine learning algorithm, namely a multilayer neural network trained by the Quasi-Newton learning rule, and special tools dedicated to pre and post-processing data.

2.3.3 K Nearest Neighbour:

The redshift estimation is performed by comparing predefined regions in the spectra and applying a k nearest neighbor regression model for every predefined emission and absorption region, individually.

Galaxy Classifier:

2.3.4 Galaxy Image Classification and Citizen Science:

There are two main morphological types based on the presence or not of a disk: spiral and elliptical, respectively. However, the multiplicity of hybrid types and the wide range of image conditions depending on factors such as the galaxy brightness, size or distance, turn the classification of this sort of image into a very complex task.

Citizen science has been a partial solution for this problem, with the engagement of amateur people from the general public in this kind of data analysis . Citizen science projects join the endeavors of myriads of volunteers committed to helping with a task that typically is time-consuming and tedious for experts, but also decisive for getting advances in a certain research problem.

2.3.5 Feature Extraction Plus a Classifier:

Among the first, we review the use of physical parameters extracted from the image, whereas the second category is dominated by the WND-CHARM feature set, originally developed as an image analysis tool for the classification of biological images, WND-CHARM represents the state-of-the-art feature extractor for the classification of galaxy images. First, through a FE phase, it computes a set of families of features from the raw images in a one by one fashion. These are categorised as image content descriptors, image transforms, and compound image transforms (transformations of a previous image transformation), composing a feature vector for the image at hand. Depending on the presence or not of colour in the image, two feature sets are available. Then, a feature selection (FS) phase chooses the most informative subset by calculating the Fisher discriminant score of each feature. Finally, the resulting feature vectors are classified using a modified nearest neighbour (NN) rule that weights the distances using the Fisher scores previously computed. Whereas in traditional NN only the closest (or k closest) examples determine the class, with WND-CHARM the distances to all training samples of each class are measured.

2.3.6 Convolutional Neural Networks:

A simpler CNN architecture to distinguish between the two main morphological types. We test this model against a well-established deeper model, ResNet, and explore how their performances are affected by the number and size of the images as well as the presence or not of colour channels. We then compare these results with the classification using FE plus classifier, aiming to investigate in which occasions the different approaches work better. Additionally, we also explore the pre-training of both CNN models considered by exploiting the availability of expert and amateur labels

within citizen science data, which to the best of our knowledge has not been investigated before.

2.4 Other method used in the project

Redshift Regressor: The Decision tree Regression algorithm is used along with K-Fold Cross Validation

Galaxy Classifier: The Decision Tree Algorithm is the main base for classification and after that, The Random Forest Classifier is used for boosting the non-normalised result generated by the Decision Tree Classifier.

2.5 Research issues/observations from literature Survey

Redshift Regressor: Human calculations take time and may contain errors and this same problem was faced while Redshift Regressor manually. Annie Jump Cannon was an extremely productive classifier. Between 1911 and 1915, she classified 5,000 stars a month, resulting in an incredible set of over 200,000 stars that eventually formed the Henry Draper catalogue. Despite Annie Cannon's incredible effort, she wouldn't be able to keep up with today's data rates. In a world where the space telescope will absorb one billion stars, a quick back in the envelope tells us that it would take any of a 16,000 years to classify them by hand.

Researchers have tried to do computations using models and are still trying to develop models with more accuracy and efficiency for Redshift Regressor. The standard techniques estimate redshifts using post-processed features, such as magnitudes and colours, which are extracted from the galaxy images and are deemed to be salient by the user. However most of the existing models have either less accuracy or require more computational resources.

Galaxy Classifier: Amateur people from the general public do the data analysis in the Citizen model. The major issue with the Citizen model is the large dependency over the individual volunteers allotted for the feature based classification and typically is time-consuming and tedious for experts.

In the other model of machine learning with feature extraction, WND-CHARM represents the state-of-the-art feature extractor for the classification of galaxy images. First is the feature extraction (FE) phase, followed by the feature selection phase and finally the resulting feature vectors are classified using a modified nearest neighbour (NN) rule that weights the distances using the Fisher scores previously computed. In the convolutional neural network model an architecture to distinguish between the two main morphological types is developed which is then compared with the ResNet model followed by the FE plus classifier model and Citizen model's results.

2.6 Summary

In the above sections we discussed the core-task that would be performed in the process of calculation of redshifts and in the process of classification of the galaxies. Next, we saw the pre-existing models predicting the redshifts and classifying the galaxies and learnt about some of their short-comings/issues.

SYSTEM ANALYSIS

3.1 Introduction

The ways to calculate redshifts and classify galaxies using machine learning models and human choice selection method for specially classification that have been developed from all over the world. By system analysis we will be able to evaluate the positive as well as negative aspects of the algorithms and models of the same.

3.2 Disadvantages/Limitations in the existing system

Redshift Regressor:

While observing the existing algorithms for Redshift Regressor we could see that either the algorithms require many orders of magnitude and more computing resources or they have less accuracy.

Galaxy Classifier:

The major disadvantage with the citizen model is the large dependency over the individual volunteers allotted for the feature based classification and typically is time-consuming and tedious for experts.

And the other two models of machine learning with feature extraction and convolutional neural network have dependency over model comparison and also have low efficacy and fluctuations in accuracy.

3.3 Proposed System

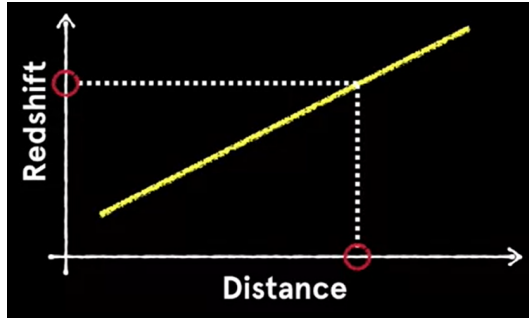
Redshift Regressor:

Now we all know that hot things glow but when we zap some hydrogen gas with electricity and heat, it glows at a very specific set of colors, that is with light of really specific wavelengths. In fact, each element in the periodic table, helium, carbon, oxygen, magnesium, all of them have a unique signature of wavelengths. And we can see their signature in the Universe. We know what the Sun is made of because we see the signature of its elements in sunlight.

And sometimes, we see a familiar signature but it's all shifted to longer or shorter wavelengths. It's hydrogen all right, but all of the wavelengths might be 10% longer than we measure in the laboratory. Because red light has longer wavelengths than blue light, we call this kind of shifting of the wavelengths, red-shift or blue-shift.

But, what could make all the lights shift like this? Well, the interesting thing is, for almost every galaxy in our universe, its light is red-shifted. We see the wavelength signatures of the elements and their stretching. We almost never see them compressed or blue-shifted. And thanks to type Ia supernova, we know something else. The further away a galaxy is, the more its light is red shifted.

If a certain galaxy is red-shifted by 10%, a galaxy twice as far away will be red shifted by 20%, and one three times as far away by 30% and so on, this is known as Hubble's law. But what is it telling us? As light travels from distant galaxies, we can think of its wavelength as stretching with the expanding space. So the further light has to travel, the more it will be stretched along the way, in other words, Hubble's law.



*[Figure 3.3. Hubble's Law :
Relationship between distance of Galaxy and it's Redshift]*

Hubble's law is extremely useful, because measuring redshift is relatively easy and certainly much easier than hoping for a timely supernova. For example, we can measure the wavelengths of light from a distant galaxy and look for the signatures of the elements. Or, we can look at the relative amount of light in broad bands rather than individual wavelengths. We can look at infrared light or red light, green light, blue light, near ultraviolet light, mid-ultraviolet light, and so on.

And we can compare this to standard libraries of galaxies and look for the redshift that gives the best match. And once we've got a match, we can measure the redshift, and, thanks to Hubble's law, map out the galaxies of the cosmos.

Galaxy Classifier:

In our proposed system, we construct a model that classifies galaxies based on their morphological description. We first take images from five sloan filters for each individual galaxy which will be available at sloan digital sky survey. The three main categories of galaxies we consider are Spiral, Elliptical and Irregular galaxies. We use a decision tree algorithm that trains our training set to build a model to map these features to the correct classifications. To improve the accuracy of the model, we use Random forest classifier and use ten-fold cross validation to evaluate the accuracy of our classifier before applying it to unknown data. Then to get the statistics, we plot the confusion matrix.

3.4 Summary

Redshift Regressor: We take a large set of galaxies, say, 50,000, for which we have measured accurate spectroscopic red shifts and derived four photometric colors from these low magnitudes. We then train a decision tree regressor that builds a model mapping the features, in this case the four colors. To the results, in this case the red shifts measured from spectroscopy. We can now run a decision tree regressor on some galaxies for which we don't have spectroscopic redshifts.

Galaxy Classifier:

Our proposed system aims at classifying the galaxies at high accuracy with the help of Decision tree algorithm and Random forest classifier. With the help of our proposed machine learning model, we lay a solid foundation for efficient classification of galaxies by overcoming the limitations of the existing systems and trying to achieve higher accuracy rates.

SYSTEM DESIGN AND IMPLEMENTATION

4.1 Introduction

To build a machine learning model for a redshift regressor and galaxy classifier, a step by step procedure is to be followed which are given in the form of modules below.

The Regression Model:

4.2 Importing Libraries

This task involves importing libraries from packages in python so that we can use the functions like numpy, DecisionTreeRegressor, KFold, pyplot, etc. which are implemented in the code.

4.3 Loading the Dataset and Generate Feature Targets

Dataset is needed to train the model. Here we use the dataset from the Sloan Digital Sky Survey(SDSS). SDSS has measured the flux of each object using optical and infrared filters. These fluxes tell us how much light the object emits in the wavelength range. Feature targets are generated by using the magnitudes from five different Sloan filters. U, G, R, I and Z and subtracting the magnitudes measured in neighboring filters to calculate.

4.4 Implementation of the function median_diff

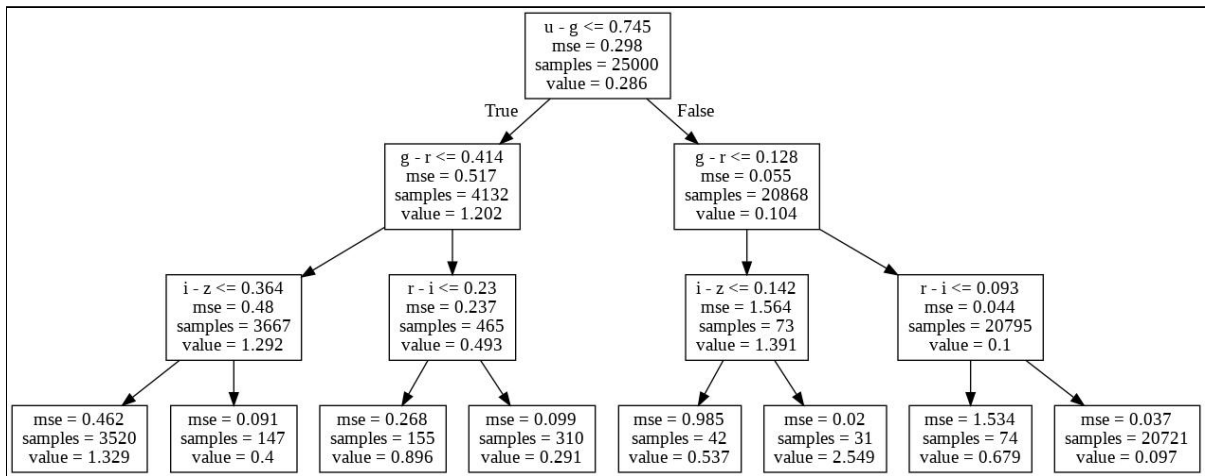
The function should calculate the median residual error of our model, i.e. the median difference between our predicted and actual redshifts. The median_diff function takes two arguments – the predicted and actual/ target values. When we use this function later, these will correspond to the predicted redshifts from our decision tree and their corresponding measured/target values.

4.5 Splitting of data & Validation of decision tree model

The dataset is split into a training set and testing set. Using the training set the model is trained while the testing set is used to test the model's predictions. Here, we will use median_diff from the previous task to validate the decision tree model..

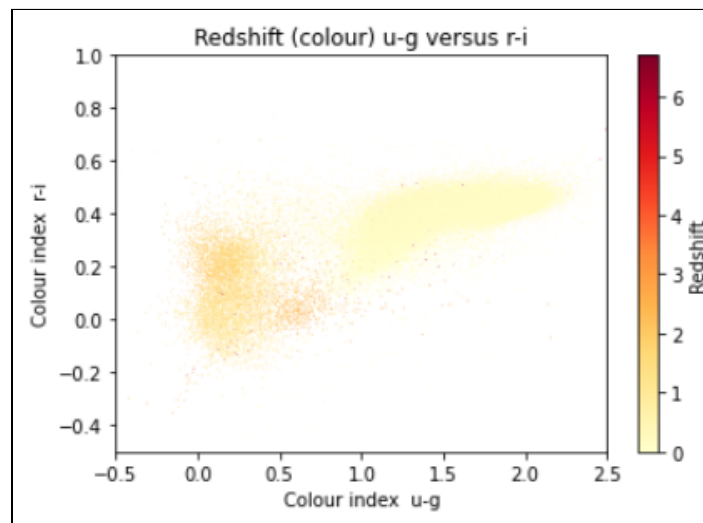
4.6 Plot the decision tree

The decision tree model is then plotted using the export_graphviz function.



[Figure 4.6. Decision Tree upto layer 3]

4.7 Defining color indexes and plotting the redshift graph



[Figure 4.7. u-g vs r-i graph. Redshift color scale on the right.]

The Classifier Model:

4.8 Importing the libraries:

This task involves importing libraries from packages in python so that we can use the functions like `numpy`, `DecisionTreeClassifier`, `confusion_matrix`, `pyplot`, `cross_val_predict`, `RandomForestClassifier` which are implemented in the code.

4.9 Splitting the data:

The data is splitted into 70% training dataset and 30% testing dataset.

4.10 Generate feature target:

We try to generate feature variables here.



4.11 Training the decision tree classifier:

Here, we train the classifier with train features and train targets and get the prediction for test features. We get the information about predicted class and actual class.

4.12 Work on the decision tree classifier:

Here, we first create a function that plots the confusion matrix. Confusion matrix is a table used to measure the performance of a classifier(in this case the Decision tree classifier) on a set of test dataset for which the true values are known. We also make a function that calculates the accuracy.

4.13 Random forest classifier:

We create a function that predicts the actual value, using the 10-fold validation and cross_val_predict. We instantiate random forest classifiers and calculate the accuracy score using the function. After getting the accuracy score, we plot the confusion matrix.

4.14 Summary

To build the model, first we need a good set of data which we take from five sloan filters for each galaxy and split the dataset into training and testing dataset. We then use a decision tree algorithm which uses our training set to build a model which maps the features to the correct classification. But the decision tree classifier suffers from

overfitting and also there are chances that the optimisation is done for only the training set and not the entire dataset. To overcome this problem we introduce Random forest classifier and 10 - fold validation. 10-fold validation helps us to optimise the entire dataset and enhance the accuracy of the model while random forest classifier helps us in reducing the overfitting and also the ability to find the outliers which do not belong in any class. Atlast, the output is visualised using a confusion matrix.





















PERFORMANCE ANALYSIS

5.1 Introduction

In this section we will discuss the results obtained by using the Redshift calculator machine learning model and the Galaxy classifier machine learning model. We will further discuss the accuracy and the correct predictions these models provide.

5.2 Performance Measures (Table/text)

Redshift Regressor:

Method	Accuracy	Interpretability	Simplicity	Speed
Naive Bayes				
Neural nets				
K-nearest				
Decision trees				
Random forests				

 **Bad**

 **Average**

 **Good**

[Table 5.2. Performance of various models]

Galaxy Classifier:

Here we have first calculated the accuracy of the model without fitting the target set and feature set in Random Forest Classifier. This is the non-normalised model result of nearly 78.7%.

The accuracy score: 0.7871794871794872

Now we introduce The Random Forest Classifier into the model which boosts the accuracy of the model to nearly 86.9%.

The accuracy score: 0.8692307692307693

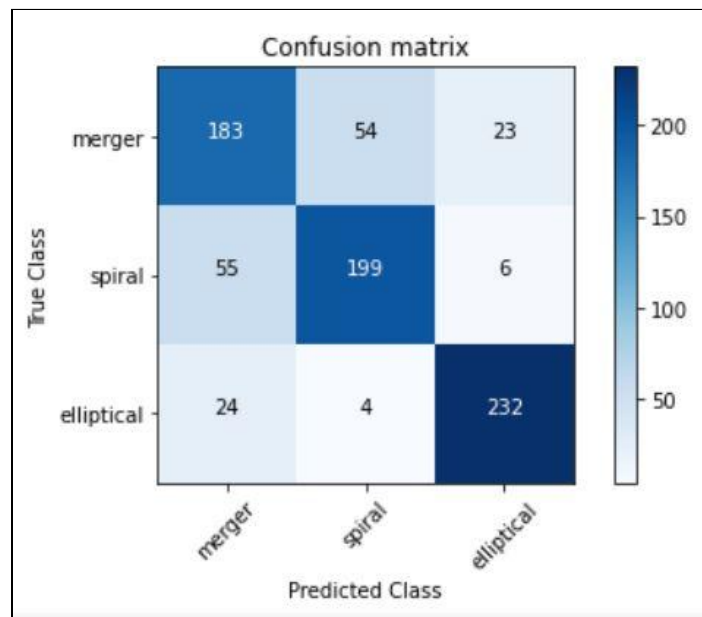
5.3 Performance Analysis(Graphs/Charts)

Redshift Regressor:

The median is the value of the observation for which half the observations are larger and half are smaller. If there are an even number of data points, the mean is taken of the two middle points. For the difference of medians, the median is computed for two samples and then their difference is taken. We are using here the median difference to validate the decision tree model.

Galaxy Classifier:

→ Initial Confusion Matrix of the classification without the use of normalization and Random_Forest_Classifier:



[Fig 5.3.1 Confusion matrix, without normalization]

The accuracy score: 0.7871794871794872

Confusion matrix, without normalization

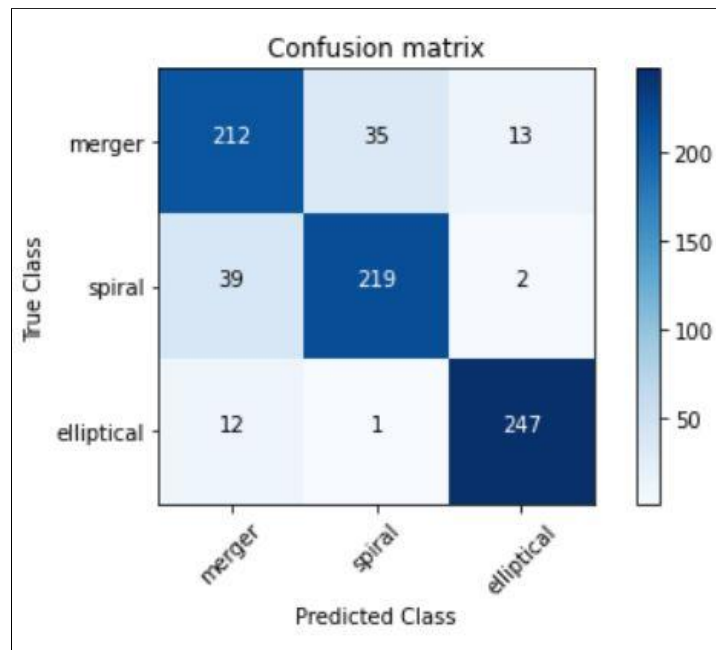
```
[[183 54 23]
```

```
 [ 55 199  6]
```

```
 [ 24  4 232]]
```

This means out of 780 galaxy images a total of 166 galaxy images have been classified wrongly by the model and a total of 614 galaxy images have been classified correctly by the developed machine learning model.

→ Final Confusion Matrix of the classification with normalization and including Random_Forest_Classifier:



[Fig 5.3.2 Confusion matrix, with normalization]

The accuracy score: 0.8692307692307693

Confusion matrix, with normalization

```
[[212 35 13]
```

```
 [ 39 219  2]
```

```
 [ 12  1 247]]
```

This means out of 780 galaxy images only a total of 102 galaxy images have been classified wrongly by the model and a total of 678 galaxy images have been classified

correctly by the developed machine learning model. This boosted the accuracy of the model nearly by 8%.

5.4 Summary

Median difference to validate the decision tree model for the redshift regressor. In galaxy classifier out of 780 galaxy images only a total of 102 galaxy images have been classified wrongly by the model and a total of 678 galaxy images have been classified correctly by the developed machine learning model. This boosted the accuracy of the model nearly by 8%.

FUTURE ENHANCEMENT AND CONCLUSION

6.1 Introduction

No model is 100% accurate and there is always scope for improvement. Hence, we propose a few modifications which will further enhance the accuracy and precision of both the models.

6.2 Limitations/Constraints of the system

Redshift Regressor: Accuracy of the decision tree on the training set gets better as we allow the tree to grow to greater depths. Conversely, the accuracy measure of the predictions for the test set gets better initially and then worse at larger tree depths where it starts to overfit the data. This means it tries to take into account outliers in the training set and loses its general predictive accuracy.

Overfitting is a common problem with decision trees and can be circumvented by adjusting parameters like the tree depth or setting a minimum number of cases at each node.

Galaxy Classifier:

As the dataset is large, the intended model performs slower than expected.

6.3 Future enhancements

Redshift Regressor : Our model can be further enhanced by integrating it to mobile or desktop applications. We can also use the concept of ensemble learning to improve our model. Ensemble method is a machine learning technique that combines several base models in order to produce one optimal predictive model. With the help of stacking, bagging and boosting we can reduce the bias, improve predictions and decrease the variance.

Galaxy Classifier : To enhance this model, we can try integrating it with mobile or small devices and make it available to common users who keep genuine interest in the universe, space and galaxies.

The model can be further enhanced by improving the accuracy.

6.4 Conclusion

The improvements and future enhancements are therefore much necessary for the model to increase its usability, accessibility and efficiency.

REFERENCES

1. <http://ogrisel.github.io/scikit-learn.org/sklearn-tutorial/tutorial/astronomy/regression.html>
2. <https://byjus.com/physics/hubbles-law/>
3. <https://hubblesite.org/science/galaxies>
4. <https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/>
5. <https://www.machinecurve.com/index.php/2020/05/05/how-to-create-a-confusion-matrix-with-scikit-learn/>
6. <https://iopscience.iop.org/article/10.1088/0004-637X/702/2/1502>
7. <https://www.omnicalculator.com/physics/redshift>
8. <https://www.zooniverse.org/projects/zookeeper/galaxy-zoo/classify>
9. https://www.academia.edu/43454704/Predicting_Redshifts_and_Morphology_of_Galaxies_with_Flux_Magnitude
10. https://www.researchgate.net/publication/266319771_Estimating_Spectroscopic_Redshifts_by_Using_k_Nearest_Neighbors_Regression_I_Description_of_Method_and_Analysis
11. https://www.researchgate.net/publication/271325246_Photometric_redshift_estimation_based_on_data_mining_with_PhotoRApToR
12. https://www.researchgate.net/publication/2236709_Estimating_photometric_redshifts_with_artificial_neural_networks