STAT 679 Final Project Progress Report

Xiaoyang Wang Ziang Zeng

1 Astronomical Challange

Our research focuses on classifying celestial objects into stars, galaxies or quasars using their spectral characteristics. With the advancement of astronomical technology, we can obtain a large amount of data, including images and spectral information, from telescopes and large-scale photometry.

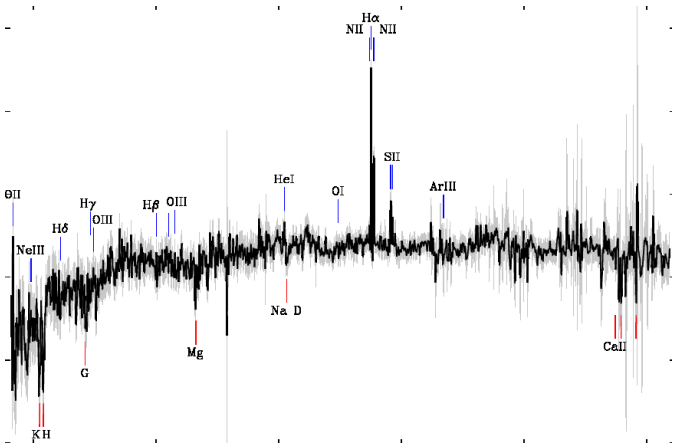
The central question of our research is: "How can we effectively use image and spectral data to accurately classify different types of stellar objects?"

2 Data

We plan to work with the astronomy dataset containing three types of data:

1. Image of the celestial objects:

2. Image of the spectra of the celestial objects.



3. Metadata of the celestial objects, whose variables are as follows:

vars	explanations
objid	Object Identifier
ra	Right Ascension angle (at J2000 epoch)
dec	Declination angle (at J2000 epoch)
u	Ultraviolet filter
g	Green filter
r	Red filter
i	Near Infrared filter
z	Infrared filter
run	Run Number
rerun	Rerun Number
camcol	Camera column
field	Field number
specobjid	Unique ID used for optical spectroscopic objects
class	Object class
redshift	Redshift value based on the increase in wavelength
plate	Plate
mjd	Modified Julian Date
fiberid	fiber ID
plateid	Plate ID

They can be found in this *link*. All the data is obtained from https://www.sdss.org.

3 EDA

First we will check and deal with missing values in the dataset. From figure 1 we can see that there are missing values for **i** and **z**. Then from figure 2, we can see that **i** and **z** are quite scattered. So, we use regression imputation to impute the missing values.

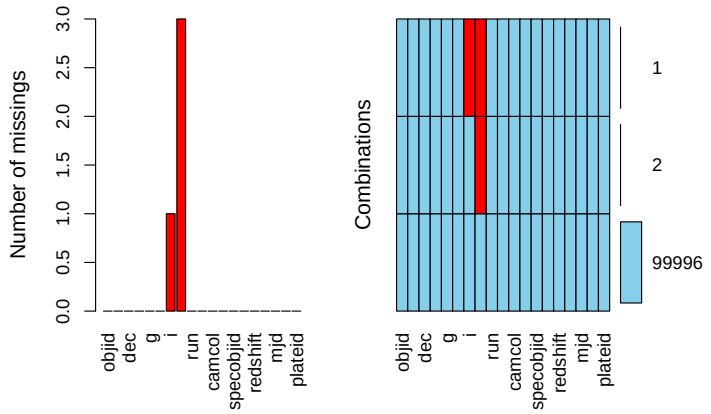


Figure 1: Missing Value Detection

Next, we give several visualizations to better understand the data. Figure 3 gives the distribution of classes, which are 33333 samples each, as selected from the website to make sure the data set is balanced.

Figure 4 gives the spread of stellar across the universe. The x-axis is right ascension angle (at J2000 epoch), and the y-axis is declination angle (at J2000 epoch). We can see some interesting patterns, but it is unclear for classification.

Figure 5 gives the correlationship between varibales. There are some variables that have strong correlationships.

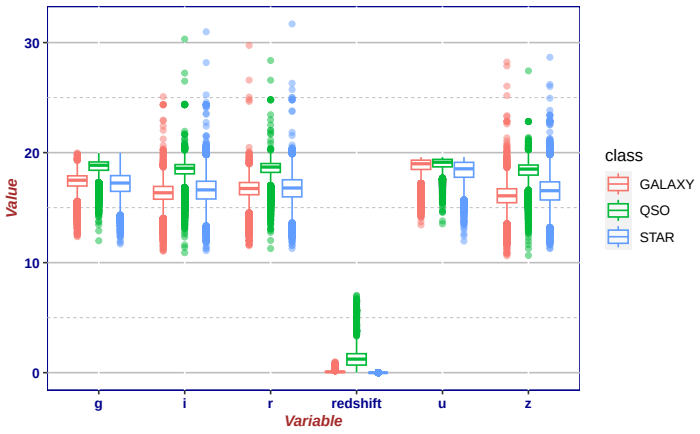


Figure 2: Boxplot

Figure 3: Distribution of Classes

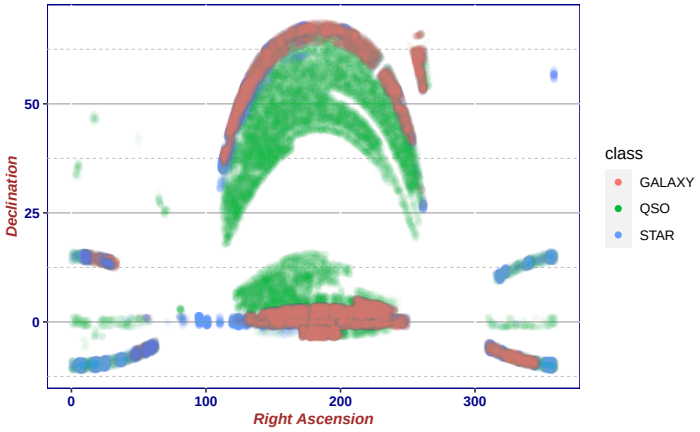


Figure 4: Spread of Stellars

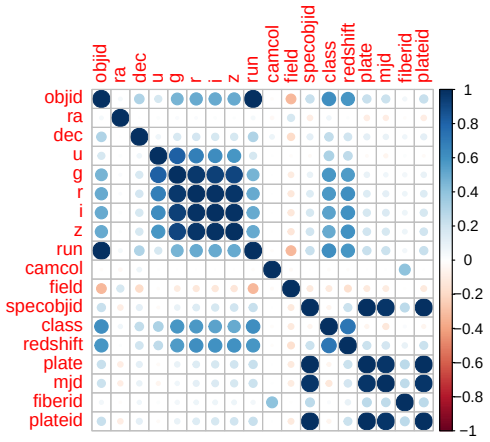


Figure 5: Correlation

4 Methods

Before we get into complex Neural Networks, we firstly try to use a simple CNN to test the performance of NN in this problem. This CNN is quit shallow (337k parameters) with two convolutional layers and a maxpooling in between and followed by three fully connected layers.The competitor is VGG16 which is a much deeper NN with 13 convoluntional layers and 3 fully connected layers(138million parameters).

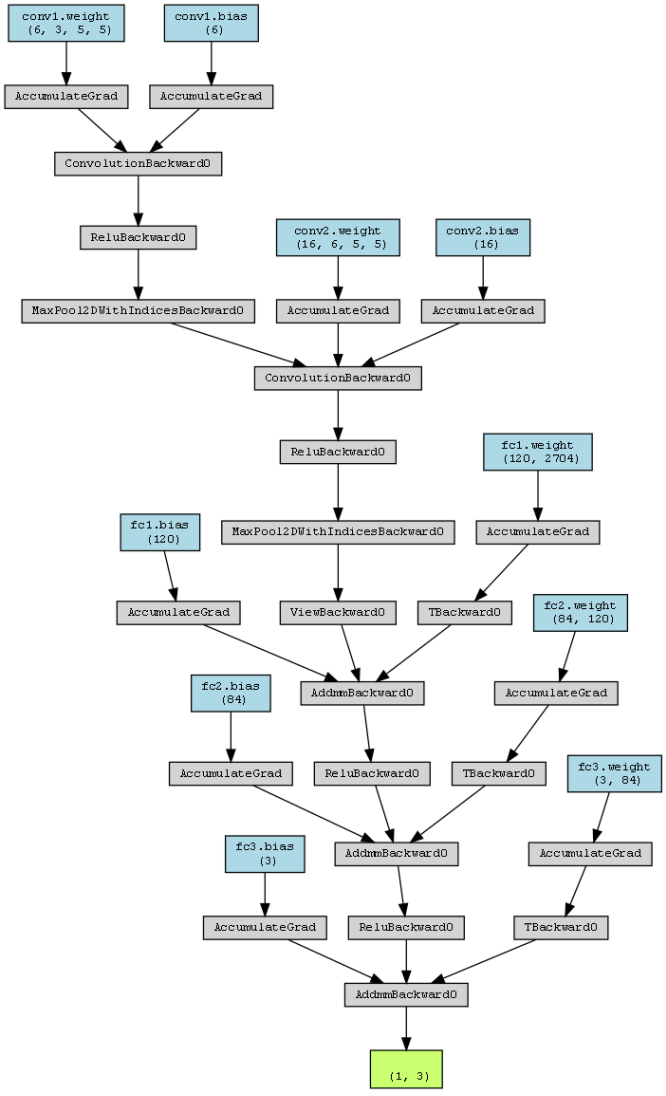


Table 1: Statistics from Confusion Matrix

Statistic	Tree	KNN
Accuracy	0.97255	0.96480
Kappa	0.95882	0.94720
AccuracyLower	0.97019	0.96215
AccuracyUpper	0.97477	0.96731
AccuracyNull	0.33582	0.33582
AccuracyPValue	0.00000	0.00000
McnemarPValue	0.00000	0.00000

5 Results

5.1 CNN

The results of this simple CNN is quite good. The cross entropy loss drop quickly after 10k iterations and get stable around 0.27. And the robustness can also be seen on its 90.2%(SGD)91.8%(Adam) average accuracy after 10 epochs training on validation data. Which actually discourages us to apply deeper NN which is quite time and energy consuming. The training took 20 min and the cross entropy loss is under 0.2 after 3 epochs training and stable at 0.18. On the validation data set, VGG has 94.23% which is better than our simple CNN. However this high accuracy not only consume lot of time but also make it more difficult to improve the performance through the combination of models. With these facts, we plan to modify the simple CNN a little without changing its main structure, and use this simple CNN to build up a final model which is expected to have similar performance as VGG16 while consume less time.

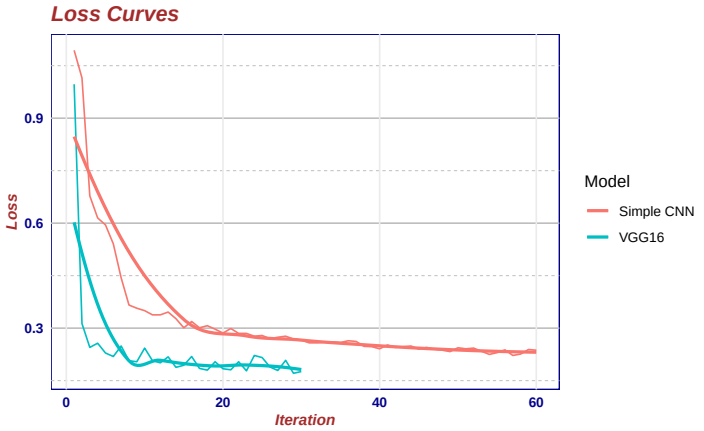


Figure 6: loss curves

5.2 Statistical Model

Regarding statistical models, we intend to use either classification trees KNN or logistic regression models, incorporating ensemble learning methods such as bagging or boosting to improve performance. Our ultimate goal is to develop a hybrid model that combines image classifiers with meta-classifiers, aiming for superior accuracy.

From the confusion matrix and statistics, we see a success in building model through the numerical data. By using several index and the red shift, our simple models reach more than 96% accuracy on validation data.

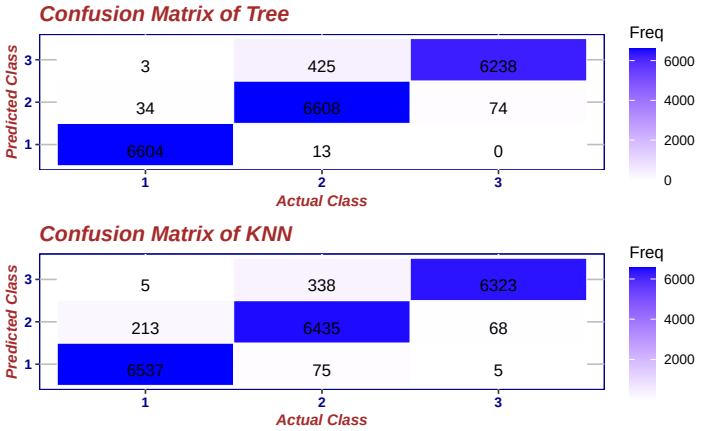


Figure 7: Confusion Matrix

6 Future Work

The next steps are clear. We will first finish the rest models: DNN for spectrum and find better classification model for metadata. Then, we will combine those three models with voting classifier to gain a better classification result.