Simon Louisin

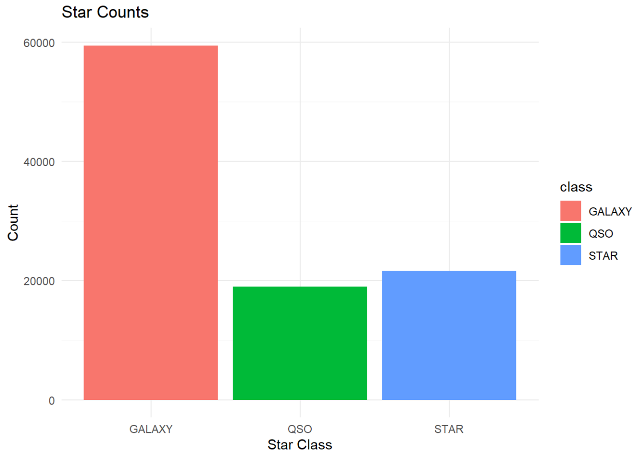
Introduction

One area of study that depends heavily on classifying objects is Astronomy. From planets to stars to galaxies there is a lot to identify in space, with a seemingly infinite amount of capacity. Many researchers rely on powerful telescopes,photography, and satellite technology techniques to observe space far and wide. My research is based on identifying particular objects in space, specifically Galaxys, Stars, and Quasars, based on quantitative spectral observations. A Star is a fixed luminous object in space, a Galaxy is a system or collection of stars held together by gas, and a Quasar is a large collection of luminous gas being pulled by a black hole.

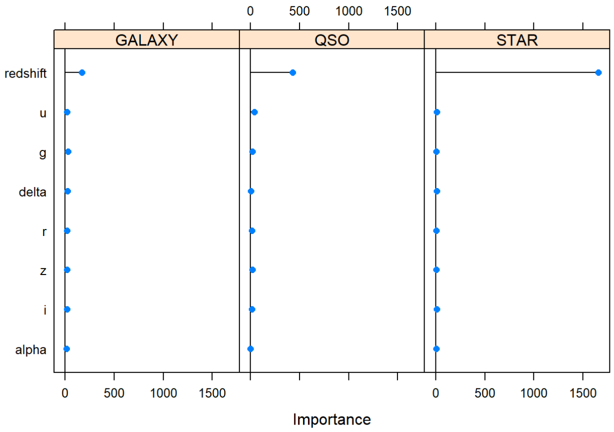
The data I analyzed is named “Stellar Classification Dataset-SDSS17” on Kaggle from user “FEDESORIANO”. It consists of 100,000 observations on 17 features and 1 response classifier. The response has 3 levels and the predictors are all of numeric type. The goal I hope to achieve is to build models that can accurately predict the class of object and then of those models identify the one that works best based on the misclassification rate. Due to the nature of the response being a factor, I rely on using classification models.

Exploratory Data Analysis

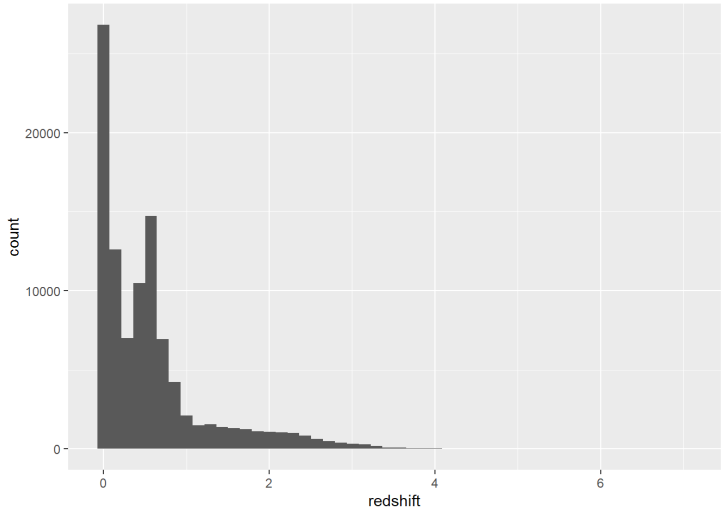
An early question I had that is crucial to the problem is how the different classes are spread in the data. I was hoping for a reasonably even spread, as a classification problem with a heavy skew is likely to not have useful results. In this case, Galaxy’s hold the majority of observations near 60% of the total, with the other 2 making up 20% each. Due to the size of the data, I thought this was reasonable to continue the analysis.



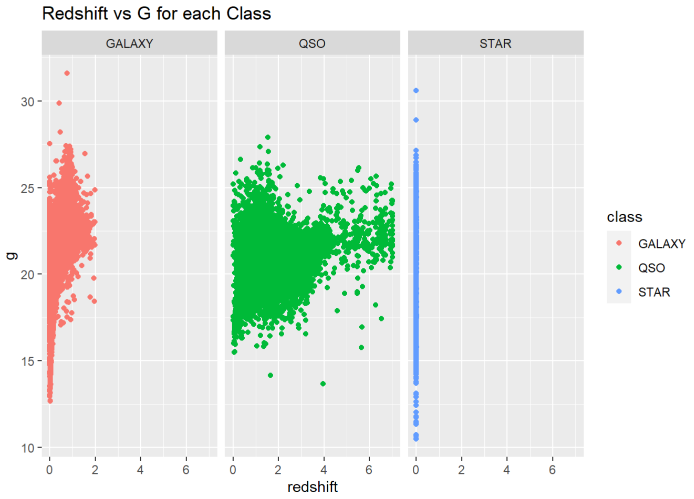
Looking at the data descriptions, I realized that some of the initial 17 features were not useful as predictors as they held information that could not be used to identify the classes like I'D names of equipment and date of photographs. For these reasons, I excluded 9 of the 17 features, and worked with the 8 features left for modeling purposes.

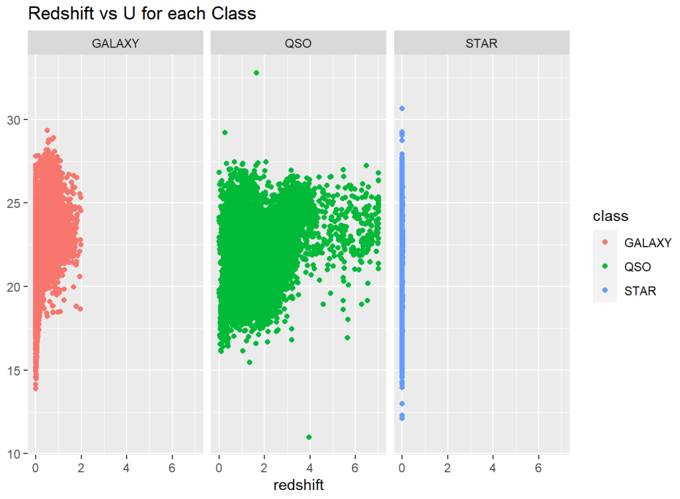


Another question I had that I was interested in is if some features are more important than others at classifying objects. I used a random forest to find the Gini Index for variables which helps determine their importance in classifying the response. It was found that there was one feature in particular that was most important for determining every class named redshift. The redshift describes how an object is moving in space relative to the earth measuring wavelength. This was interesting to find because one predictor has this huge influence on each class, with the others much lower in comparison. Furthermore, the histogram of this predictor reveals a right skew with most counts between 0 and 1, eventually dwindling at around 4. The maximum value of redshift observed in the data was 7. This tells me that many observations have 0 redshift, basically they are not moving compared to us in space. This may be an underlying reason this certain predictor can explain the class well.



I explored the redshift's interaction with the other top variables according to their respective importance. The feature 'G' represents the green filter in the photometric system and 'U' represents the Ultraviolet filter in the photometric system. We can see that Stars have a value of 0 for redshift, with Quasars and Galaxies both having values of 0 and greater.





Stars and Galaxies seem to have higher ranges of G and U compared to Quasars, but Quasars contain a larger spread for redshift.

Model Building

To begin my modeling process, I separated the dataset into a training and testing set. The training dataset contained 85% of the data at 85,000 observations, with the testing set containing the other 15% at 15,000 observations.

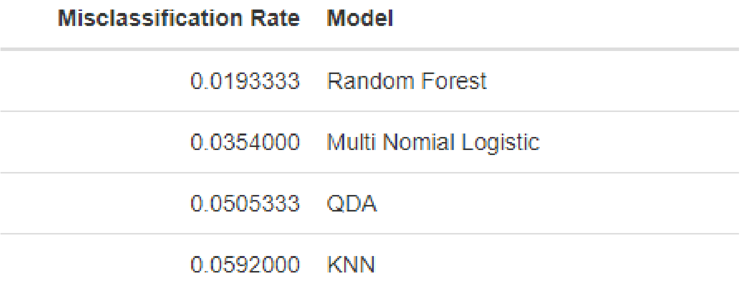
The first model I chose to work with was a K-Nearest Neighbors Classification (KNN). KNN is a non parametric model that identifies for each point its k nearest neighbors. It then uses the average class for all its neighbors to determine its predicted classification. I used the caret package to train the model using the class as the response, and all 8 other features as predictors. I used a cross fold validation technique with 5 folds. The tuning parameter of k, which is the total neighbors to calculate distances from, I had set to test between 5 and 11. For KNN, the predictors were centered and scaled. The accuracy of the model on the training set was near 93.5% with the best tune at k = 5.

The second model I worked with was a Random Forest. A Random Forest is a number of classification trees in this data on bootstrap training samples where each time the data is split, it pulls m random predictors as candidates from the full sample of p predictors to split the data. I used the caret package to train the model using class as the response, and all 8 other variables as predictors. I used the cross fold validation with 5 folds , with an mtry from 1 to 10, which had a best tune of mtry at 6. I had used 100 Trees, I would have liked to do more but the computation time was very long. The accuracy on the training set was 97.9%.

The third model I used was a Quadratic Discriminant Analysis (QDA). QDA uses an estimated covariance matrix for each class using the predictors. Using the caret package to train, the validation approach involved cross fold validation with 5 folds. For QDA, the predictors were centered and scaled. This led to a 94.7% accuracy for the training set.

The fourth model I chose was a Multinomial Logistic Regression. Multinomial Logistic Regression is an extension of Logistic Regression, but allows for levels in the response than just binary. It calculates the probability that a point is in a given class, and then the highest probability is assigned as the predicted classification. I used the caret package to train the model. The validation approach had cross fold validation with five folds. The predictors were centered and scaled. The accuracy for the training set was 96.2%

Model Comparison

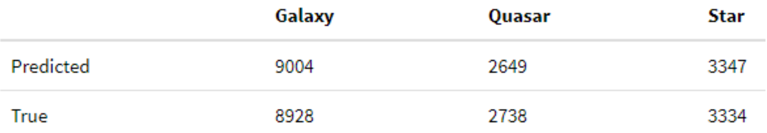


Misclassification Rate	Model
0.0193333	Random Forest
0.0354000	Multi Nomial Logistic
0.0505333	QDA
0.0592000	KNN

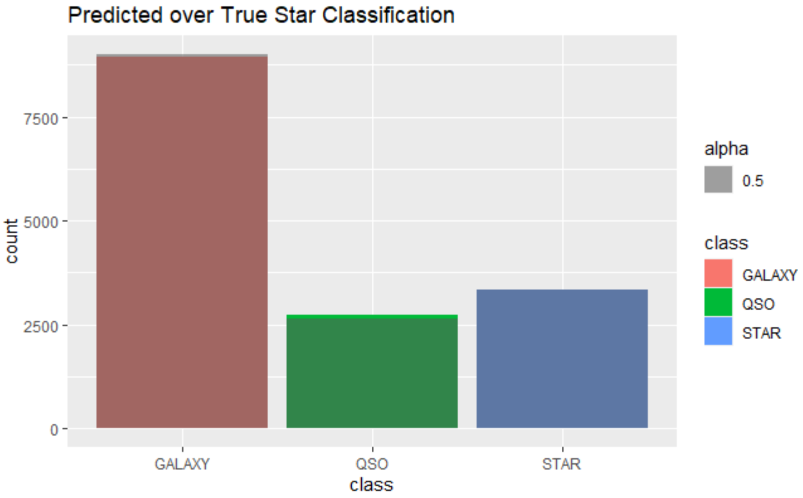
The breakdown of each model's performance using its Misclassification rate is stated above using predictions on the testing set. The best model here is the Random Forest as its Misclassification rate is the lowest at just under 2%. This estimate means that for 100 new observations, it will misclassify only about 1 or 2 observations. The other models in this analysis are slightly behind, with the highest misclassification rate being about 6% for the KNN Classification model.

Conclusion

Overall, the best model tested for the classification of Stars, Galaxies, and Quasars was the Random Forest model. There may be other models that were not considered that fit this data better, but with a misclassification rate of about 2%, the random forest model is not too bad at predicting the correct class. It appears that it can slightly over predict the Galaxies, and under predict Quasars. For Stars it does quite well.



	Galaxy	Quasar	Star
Predicted	9004	2649	3347
True	8928	2738	3334


Predicted over True Star Classification
count
GALAXY
QSO
STAR
class
alpha
class
GALAXY
QSO
STAR