Classifying Foreground Dwarfs and Background Giants in the Perseus Cluster using Machine Learning

Steven Li* and Dan Rachou†
Department of Physics and Astronomy, San Jose State University.
(Dated: May 31, 2019)

When ever you look at a picture of a galaxy far far away, it is almost always unambiguous if that galaxy is a giant but very far away making it seem like a dwarf or it is a dwarf very close making it seem like a giant. The only truly way to know what type of galaxy it is, is to know it's true redshift but that isn't always available. Utilizing Machine Learning algorithms in scikit-learn's machine learning libraries, we are able to use other parameters that do not include the spectroscopic redshift to see if a combination of those parameters are able to make a concrete classification similar to spectroscopic redshifts. We have found that indeed a combination of those more accessible parameters can also produce very high accuracy results.

I. INTRODUCTION

Within the Perseus cluster we are able to tell the difference between foreground dwarfs from background galaxies with photometric redshift 0.08. However, we have a hard time differentiating foreground galaxies from background galaxies with redshift between [**0.04-0.07**]. In a previous research assignment, Dan compared visually foreground dwarf galaxies to background giant galaxies in the Perseus cluster with redshifts that falls within this particular range. He found that there were a lot of similarities, especially in shape, but you could also tell the difference by comparing the size and intensity of color between the two galaxy groups. However, those were mere observations and there was no clear quantifiable way to discern the differences. Due to the fact that there is an overlap of both background giants and foreground dwarfs between a photometric redshift of [**0.04-0.07**], we determined that there was no clear-cut cookie cutter way to differentiate the two without performing rigorous detailed analysis. Visual representation shown in **Figure 1**.

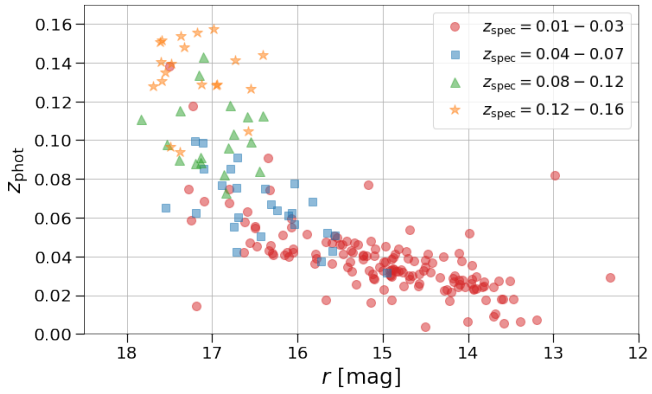

FIG. 1. Plot of photo-z vs. magnitude produced by Dr. Aaron Romanowsky that shows an overlap of two distinct z-spec ranges. Red dots representing galaxies in the foreground of the Perseus cluster.

From that, we wondered if there are any other single parameter or combination of parameters which are more accessible, abundant, and does not create a gray area where we are not able to differentiate between the two galaxy groups.

We have decided to aid with completing the Perseus cluster catalog by using the scikit-learn library to build a classification model, and perform data analysis to see if there are any other parameters besides spectroscopic redshift that can discriminate between foreground dwarf galaxies in the Perseus cluster and giant background galaxies.[8]

II. METHODS

Within the field of machine learning, there are two types of tasks: supervised and unsupervised. The main difference is that in supervised learning we train the model using data that is well labeled; in other words, we already know what the what output from the model should be. In unsupervised learning, the training data is unlabeled, and the goal is to infer the natural structure present within the set of data points. The system basically tries to learn without a teacher.

The biggest challenge in both supervised and unsupervised learning is getting enough data to train and test the model accurately. According to professor Romanowsky, we know about approximately 100 instances of foreground dwarfs and background giants in the same redshift range inside the Perseus cluster. The large amounts of data is the main factor in building an accurate model.

After discussing the two types of tasks, we decided to go with supervised learning. The reasoning behind our decision is that we are able to substitute the lack of data within the Perseus cluster by utilizing known data from two nearby clusters, NGC 383 & 507.

* steven.li@sjsu.edu
† dan.rachou@sjsu.edu

As previously mentioned, we have a hard time differentiating foreground dwarfs from background galaxies by only looking at the the photometric redshifts. To label galaxies as foreground dwarfs or background galaxies we need to look at the true redshift, and the spectroscopic redshift is the closest one to that. The SDSS library misses a lot of spectroscopic redshifts of different stellar objects. Fortunately, we can retrieve many of the missing spectroscopic redshifts from NED.[1] In NED we are limited to a search radius of 60 arcminutes. We combined the data set from the NED search with the SDSS library using CrossID.[6] In CrossID, we set the search radius of each object to be 0.03 arcminutes to make sure that we CrossID the right objects from SDSS and NED. We then merged the spectroscopic redshifts from NED with the resulting data set from CrossID. We ended up with data sets ready for data analysis.

The data analysis of these two clusters were done by taking the whole data set for each specific cluster and plotting them on a histogram, and then zooming in on first range of spike. The spike represents the relevant data. The data analysis is a required step to filter out outliers such as the one at a redshift of 7 for NGC 507, which is clearly an error that we extracted from SDSS or NED shown in **Figure 2**.

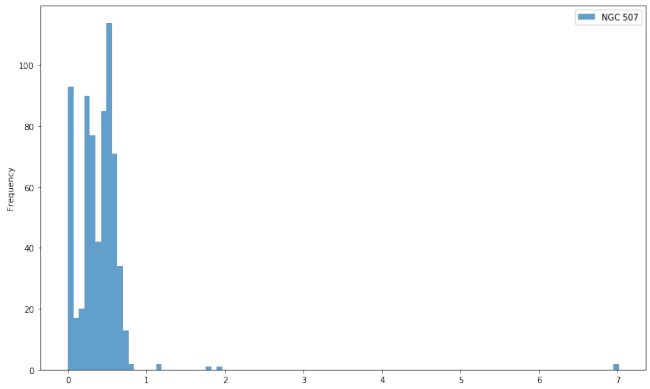

FIG. 2. A histogram of all the redshift found within the whole NGC 507 without filtering any outliers.

After zooming into the first spike, zoomed in even further around the redshift value of the center of the each specific cluster which we obtained from NED to be **0.01700±0.00004** for NGC 383 and **0.01646±0.00002** for NGC 507. We found a Gaussian shape denoting the foreground galaxies within. Both the original histogram and zoomed in histogram for each specific cluster is shown in **Figure 3** and **Figure 4**.

From that, we take the starting and ending redshift of the Gaussian to determine the range of redshifts that each cluster's foreground galaxies are found in. The data analysis on the redshifts of NGC 383 data set shows us that galaxies with redshifts between [**0.0125-0.0225**] are in the foreground of the cluster. We performed the same data analysis on the NGC 507 data set and found the

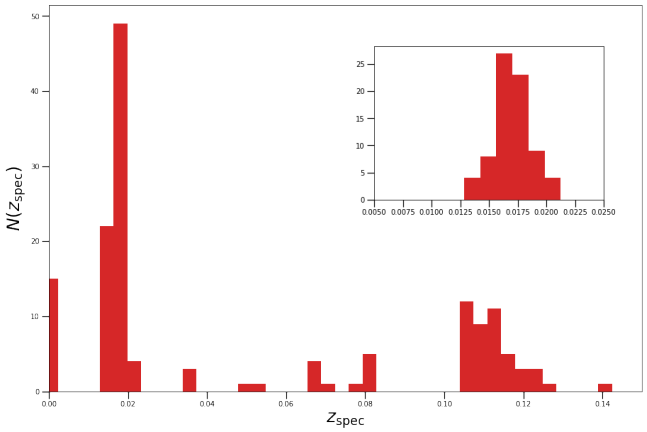

FIG. 3. A histogram of the range of redshift that NGC 383 foreground galaxies are found within.

redshift range of foreground galaxies to be from [**0.012-0.022**].

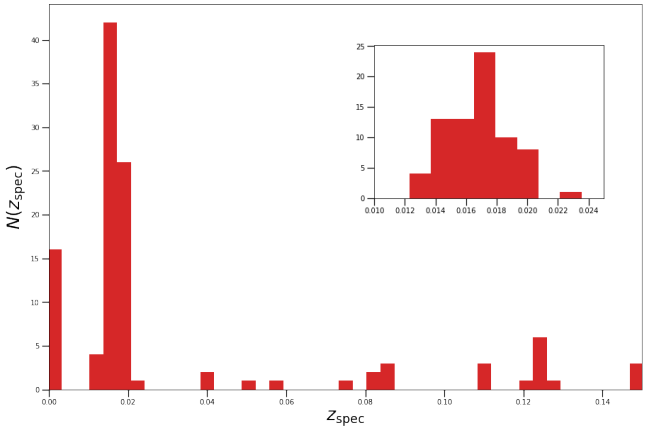

FIG. 4. A histogram of the range of redshift that NGC 507 foreground galaxies are found within.

Our next step was to use these redshift ranges to create data sets of foreground dwarf galaxies and background giants. Looking at **Figure 1** we can see instances of both background giants and foreground dwarfs in the magnitude range of [**15-17.5**]). For our training data set we collected galaxies of similar magnitude. Any other galaxies that do not fall in this range of magnitude are not considered as relevant because they are not part of the problem area we are targeting.

Combing the number of background galaxies from Perseus and NGC 383 & 507, we got a total of 160 background galaxies. We plan to increase the data set in the future by performing a bigger radius search around Perseus in SDSS.

The same process was done for the foreground dwarf galaxies from these three clusters, netting us a data set of 56 dwarf galaxies which is not enough to train an accurate classification model. We increased the data set by per-

forming an all-sky query in SDSS. We filtered the search by similar spectroscopic redshifts, magnitudes and color. In preparation for an all-sky query we did data analysis to find the distribution of the parameters just mentioned. We already knew that the range of the magnitude should be 15-17.5. Next was to find the range of redshifts and color. We used the dwarf data set to calculate the color distribution by subtracting the U-band parameter with the Z-band parameter. The redshifts were already within the data, so we only needed to make a plot using a histogram to find out the range. The resultant two histograms utilizing the new data set can be seen in **Figure 5**. The majority of the dwarf galaxies found in Perseus, NGC 383 and 507 have a U- and Z-band difference within the range of [**2.5-3.4**]. Since red sequence dwarfs in the outer regions of the Perseus Cluster are bluer than the ones found in the Perseus cluster, we decided to include this color range as one of the filters in our all-sky search for similar dwarf galaxies.

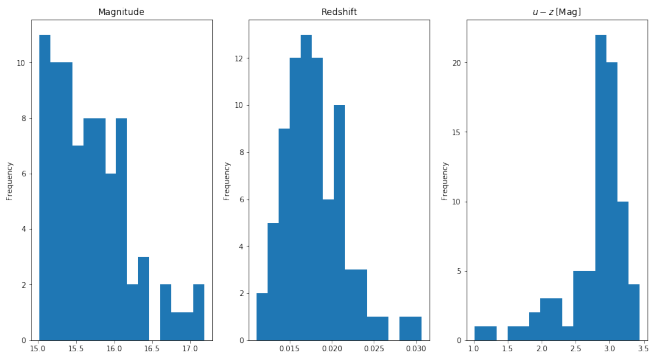

FIG. 5. 3 Histogram showing the magnitude, redshift and color frequency on the combined Perseus, NGC 383 & 507 Cluster.

We could not use the redshift range from Perseus, NGC 383 or 507 redshift range due to the individual nature inside each cluster. Instead, we looked at redshift of the center of each cluster and used median ± **10%**. This resulted in a Redshift range: [0.0153-0.0187]. Narrowing our all-sky search with the information above, we netted a total of 413 dwarf galaxies to use in our training data set.

The next phase was to cherry-pick the parameters we believed to have an effect in differentiating background giants from foreground dwarfs. We began by grabbing all the primary and secondary objects in the PhotoObjAll table from SDSS, which contains all the parameters for each photometric object in SDSS, and then carefully removing the least discriminating features. [2] We narrowed down the total number of parameters from 509 down to 137 for the training data set, and a small subset of the data set can be seen in Table I. The 137 parameters we decided on were included in the final CrossID query.

On the data set generated by the two final CrossID queries, we performed Support Vector Machine Learning

	psfMag_u0	psfMag_g0	psfMag_r0	psfMag_i0	psfMag_z0
0	21.08928	19.37086	18.25864	17.81117	17.51476
1	21.41094	19.66853	19.04671	18.49730	18.12082
2	21.27979	19.46498	18.33810	17.92343	17.63418

TABLE I. Sample of the parameter used in the whole training dataset

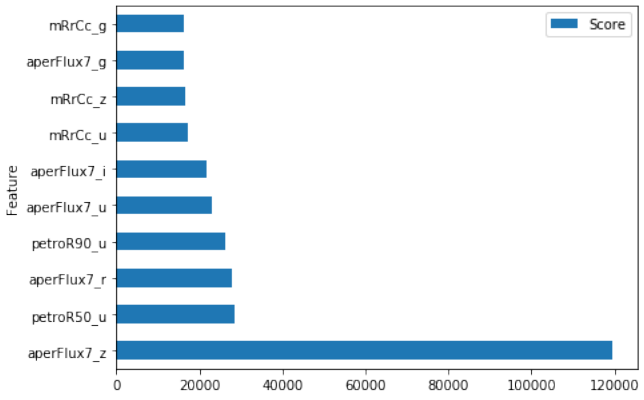

FIG. 6. Univariate selection on training dataset to select the top 10 features that have the strongest relationship with the output.

classification algorithm (SVM) which its main advantage is the effectiveness in high dimensional spaces.[5] In order to improve the accuracy of the classification model, we conducted feature engineering such as Univariate Selection, Feature Importance, and Correlation Matrix with Heatmap before building the model. Heatmap[9].

Univariate selection can be used to select those features that have the strongest relationship with the output variable. From the Univariate selection, we ran SelectKBest to extract the top 10 best features that are shown in **Figure 6**. [4]

Feature importance is provided by the feature importance property of the Extra Tree Classifier. It gives us a score for each feature of our data where the higher the score the more important or relevant is the feature towards our output variable. The Extra Tree Classifier from the scikit-learn library utilizes a meta estimator to fit a number of randomized decision trees on various samples of our training data set and then uses averaging to improve the overall accuracy and reduce over-fitting within the training set.[3] The top 10 most important features obtained by feature importance is shown in **Figure 7**.

The methods above resulted in a list of the most important features and best scoring features which we used with SVM to determine the combination of parameters that would give us a scoring that reduces over-fitting, training time, and accuracy alongside the model cross-validation. Lastly, Correlation Matrix with Heatmap gave us a visual representation of how each parameter correlates with each other. In this case, we used Correlation Matrix on the top 10 best features obtained by

TABLE II. Model Type, Features and Accuracy

Kernel Model	All Features	10 Best Features & Most Important Features	10 Best Features	10 Most Important Features	20 Most Important Features
Gaussian	0.84 (±0.26)	0.87 (±0.27)	0.87 (±0.27)	0.91 (±0.15)	0.91 (±0.15)
Sigmoid	0.50 (±0.15)	0.49 (±0.37)	0.49 (±0.37)	0.66 (±0.02)	0.66 (±0.02)
Polynomial (n=1)	0.90 (±0.09)	0.87 (±0.06)	0.87 (±0.11)	0.93 (±0.16)	0.94 (±0.16)
Polynomial (n=2)	0.90 (±0.16)	0.89 (±0.10)	0.87 (±0.13)	0.94 (±0.09)	0.95 (±0.10)

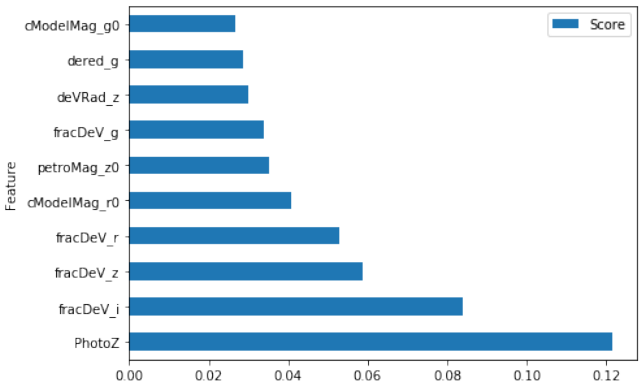

FIG. 7. Feature Importance using Scikit-Learn's Extra Tree Classifier to determine the top 10 most important/relevant feature to our output variable.

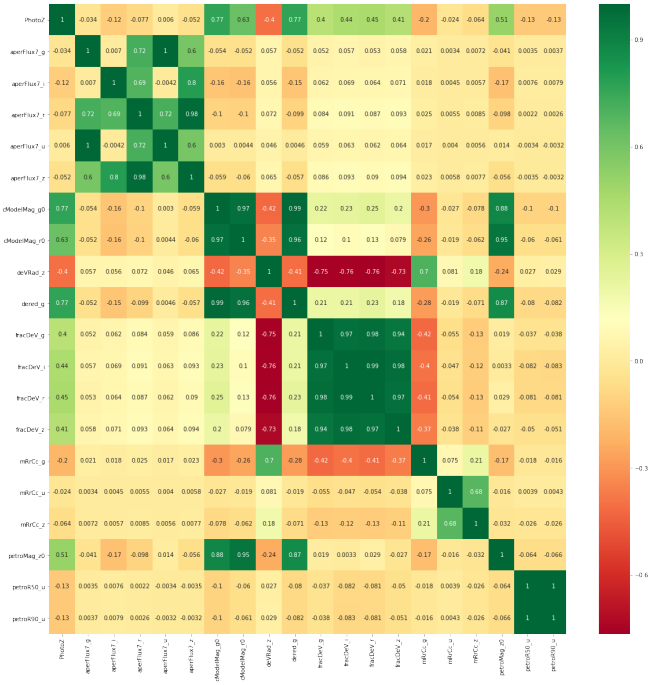

FIG. 8. Correlation Heatmap of the top 10 strongest relationship features and the top 10 most important features with the output.

Univariate Selection and the top 10 most important features obtained by Feature Importance that is shown in **Figure 8**.

A correlation score of 1 means that the two features are exactly correlated with each other while something close to 1 implies strong correlation. A correlation score revolving around 0 means minimal to no correlation between the two features while a correlation score approaching -1 means inverse correlation between the two features.

With this information, all that was left was to pick the model with the best accuracy score and feed it the Perseus cluster prediction data set, and compare the results with known dwarfs from Perseus by counting the duplicates between the two lists. Lastly, we generated a thumbnail search of the resultant data set for a visual check and further analysis.

III. RESULTS

We tested multiple models with different sets of features from the refined parameter list. The models of Support Vector Machine that our tests used were the Gaussian kernel, Sigmoid kernel and the Polynomial kernel of degree 1 and 2. Our feature sets ranged from all features, top 10 best features, top 10 most important features and many more different combinations of the two. The scores that resulted from utilizing scikit-learn's cross valence for cross validating scoring on a given model are all depicted in **Table II**.

Out of all the models and parameters used, the one that gave us the best accuracy from running cross-validation on the training data set was the polynomial model with degree 2 on the 20 Most Important Features. It achieved an accuracy score of **0.95(±0.10)**. This is the highest scoring model with the lowest error bar that we have trained using our data set. We used this model to find new dwarfs in the Perseus Cluster.

After picking the model, we imported the prediction data set of Perseus galaxies in the magnitude range of **[15.5-16.5]**, which is the most troublesome magnitude range in our targeted problem area, and fed the chosen Machine Learning model the curated data set. The model took the data set and predicted whether the galaxy is in the foreground or in the background by denoting a 0 for background giant and a 1 for foreground dwarf into an

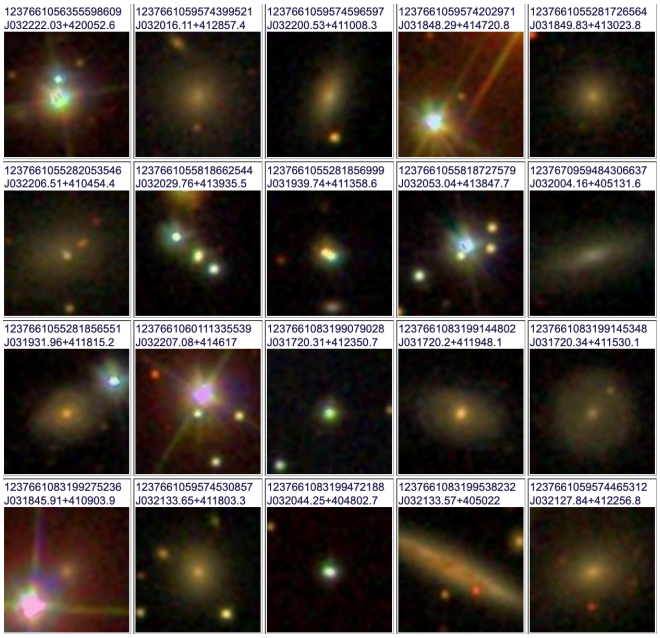

FIG. 9. Visual subset of the Machine Learning classified dwarf galaxies generated by a thumbnail search of the subset in SDSS image list format.

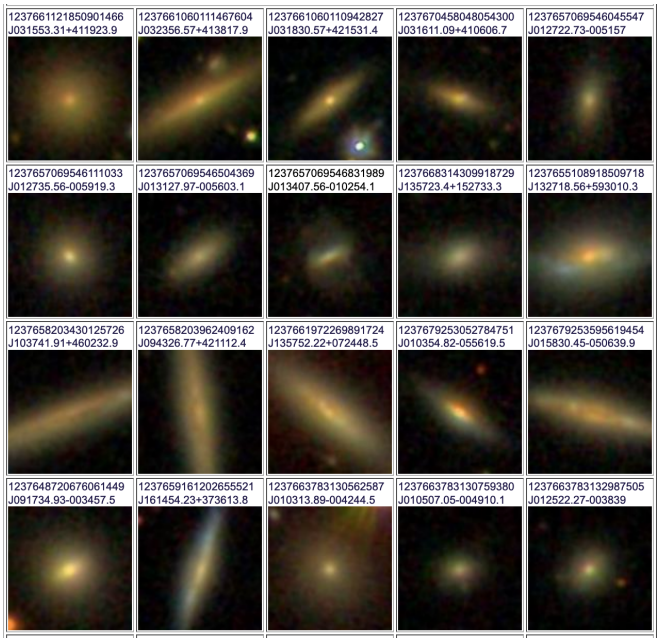

FIG. 10. Visual subset of known dwarf galaxies generated by a thumbnail search of the subset in SDSS image list format.

array equal to the input test data set size. The prediction data set consisted of 121 galaxies from the Perseus cluster that can either be classified as foreground dwarfs or background giants, and after having it fed into the machine learning model it returned a result of 84 foreground dwarfs and 37 background giants.

We took that resultant array of 0s and 1s, and concatenated it to the original Perseus prediction data set. Then we dropped all galaxies that were not classified to be foreground dwarf galaxies. After that, we took that array and concatenate it with the data set of 183 known Perseus dwarf galaxies, resulting in a total of 267 dwarf galaxies. Then we dropped all the duplicated entries on the objID parameter. Our ML model ended up classifying 68 new potential dwarfs. A visual subset of the galaxies that the machine learning model deemed as foreground dwarf galaxies are shown in **Figure 9**. The image list was generated by taking the subset and converting it in the a SDSS image list search format then having it input into SDSS DR13 Image List Tool. [7]

IV. DISCUSSION

From **Figure 9** we see that our machine learning model has definitely predicted galaxies within the Perseus cluster prediction set correctly as some of the thumbnails look very close to known foreground dwarfs shown in **Figure 10** .

However, some of the objects in our prediction set are undoubtedly non-galaxies. We blame SDSS for wrongly categorizing non-galaxies as galaxies because we specified the object type to galaxies in our SQL query. For other thumbnails, it is not clear what it is and additionally some of them have interference that either blocks a portion of the galaxy or completely hinders the galaxy to be shown. We see it necessary in the future to include a data-cleaning step before feeding the prediction data set into our classification model.

Although our model has predicted some galaxies correctly, we still cannot trust the model completely in prediction. We decided to look for false positives in our result data set before making a final verdict. To determine if any of the predictions have been falsely flagged, we compared the resultant machine learning array with the data sets of known background giants and foreground dwarfs. If a known dwarf galaxy was classified as background giant or vice versa, we have found a false positive.

After comparing the output objects classified as dwarfs to the known background galaxies, we found that 3 of the total 68 objects were actually background galaxies shown in **Figure 11**, leaving us with a total of 65 new potential foreground dwarfs that we didn't already know about. If we can prove that the remaining objects are Perseus dwarfs, we have build a classification model that can classify foreground dwarfs with an accuracy rate of **95.58%**.

We found that 6 out of the 37 classified background giants were already known, leaving 31 potentially new background galaxies. Out of those 31 potentially new background giants, none of them were falsely identified. If we can confirm that the 31 galaxies are indeed background giants then the model prediction on this set of Perseus cluster has an accuracy **100.00%** in classifying giant background galaxies.

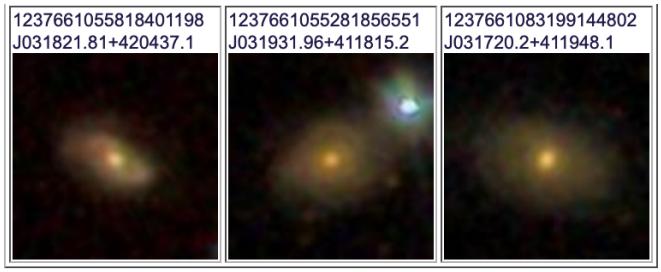

FIG. 11. Visual set of the Machine Learning dwarf galaxies that were actually known background giants falsely labeled as foreground dwarfs generated by a thumbnail search of the set in SDSS image list format.

V. CONCLUSION

From our attempt at classifying foreground dwarfs and background giants in the Perseus cluster using machine learning, we found that Support Vector Machine is a very viable machine learning algorithm to perform this type of classifications. SVM has indeed found other parameters besides spectroscopic redshift that can discriminate between foreground dwarf galaxies in the Perseus cluster and giant background galaxies.

With a cross-validation score of **95±10%**, we found that our model created using SVM could have a potential ceiling of an accuracy **95.58%** in classifying foreground dwarf galaxies and **100.00%** in giant background galaxies if in the future we are able to validate that the classifications were indeed correct. Only the 20 Most Important Parameters are included in our model but even with just using all the relevant parameters, we get a cross-validation score of **91±6%**. So even without utilizing feature engineering to improve accuracy, over-fitting and training time the model should still give decent results.

Lastly, this research can be extended by performing different supervised machine learning algorithms such as Random Forest or Linear regression with Principal Component Analysis to reduce the number of parameters. More research can also be done by using unsupervised machine learning such as k-means or Apriori algorithm to see if they can give similar results.

[1] **NASA/IPAC Extragalactic Database**. https://ned.ipac.caltech.edu/.

[2] **PhotoObjAll Parameter Table**. https://skyserver.sdss.org/dr12/en/help/browser/browser.aspx?cmd=description+PhotoObjAll+U#&&history=description+PhotoObjAll+U.

[3] **Scikit-learn: ExtraTreesClassifier**. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html.

[4] **Scikit-learn: SelectKBest**. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html.

[5] **Scikit-learn: Support Vector Machines**. https://scikit-learn.org/stable/modules/svm.html#support-vector-machines.

[6] **SDSS CrossID for DR12**. https://skyserver.sdss.org/dr12/en/tools/crossid/crossid.aspx.

[7] **SDSS DR13 Image List Tool**. http://skyserver.sdss.org/dr13/en/tools/chart/listinfo.aspx.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. **Scikit-learn: Machine Learning in Python**. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] Raheel Shaikh. **Feature Selection Techniques in Machine Learning with Python**. https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-pytho 2004.