Draft version November 21, 2019
Typeset using LaTeX **twocolumn** style in AASTeX63

Host Galaxy Based Supernova Classification

Noel Chou,[1] V. Ashley Villar,[1] Edo Berger,[1] David Jones,[2] and Michelle Ntampaka[3,1]

[1] *Center for Astrophysics | Harvard & Smithsonian, Cambridge, MA 02138, USA*
[2] *Department of Astronomy and Astrophysics, University of California, Santa Cruz, CA 95064, USA*
[3] *Harvard Data Science Initiative, Harvard University, Cambridge, MA 02138, USA*

ABSTRACT

Upcoming optical surveys such as the Large Synoptic Survey Telescope will discover supernovae at rates far out-pacing feasible spectroscopic classification. It is therefore critical that we optimize alternate classification methods using all available information. The use of host galaxy data for classification has not been fully developed, despite well-known trends between host galaxy properties and supernova types, and in particular there is an absence of machine learning methods. Using Pan-STARRS1 Medium-Deep Survey (PS1-MDS) images, we trained machine learning algorithms to predict supernovae types using solely contextual information. In particular, we present a random forest classifier using known host galaxy properties, and a convolutional neural network using host galaxy images. Classifying between types Ia, Ibc, II, IIn, and superluminous, we find the convolutional neural network performs significantly better than the random forest classifier. They achieve average accuracies per class of 60% and 45% respectively. Classifying between supernovae Ia and all other types, we achieve 73-74% accuracy with both algorithms. This is a significant improvement from existing host-galaxy based classification work (Foley and Mandel 2013). Future work includes combining our algorithms with photometric classification pipelines for optimized classification.

Keywords: astronomical databases: surveys — supernovae: general — galaxies: general

1. INTRODUCTION

Supernovae are traditionally classified according to their spectra. However, optical surveys are discovering supernovae at exponentially increasing rates with which spectroscopic follow-up will be unable to keep up. There is therefore an urgent need for alternative algorithms to classify the remaining supernovae. Much work has been done in developing photometric methods, for example Boone (2019); Lochner et al. (2016); Möller & de Boissière (2019); Muthukrishna et al. (2019); Pasquet et al. (2019); Villar et al. (2019). However, far less work has been done taking into account host galaxy information. This is despite known correlations between host galaxy properties and supernova types. For exam-

ple, core collapse types of supernovae generally come from massive, young stellar progenitors, and they are therefore less likely to occur in elliptical galaxies where there is little star formation. In contrast, Type Ias, having white dwarf progenitors, frequently occur in both star forming and non-star forming galaxies. Foley & Mandel (2013) create a classification algorithm to distinguish between Type Ia and core collapse supernova using direct correlations between individual properties of the host galaxy and the supernova type. However, there remains a lack of algorithms that can make use of more complicated correlations between combinations of properties and supernova types. There is also a lack of host galaxy based algorithms classifying between different core-collapse types of supernovae.Machine learning techniques are well suited to this purpose because of their ability to learn and extrapolate complex trends which may not be obvious to the human eye.

Our purpose here is to further develop the use of host galaxy data towards supernova classification, utilizing machine learning, motivated by the importance of optimizing non-spectroscopic galaxy classification through all available means. We train our algorithms using the Pan-STARRS medium-deep survey which has enough depth for us to detect hosts for a range of supernova types and redshifts. It furthermore has similar filter, depth, and cadence properties to the upcoming Large Synoptic Survey telescope, making our algorithm easily transferable to use with LSST data. We first train a random forest classifier on known galaxy properties such as magnitude, ellipticity, and radius, which we extract from Pan-STARRS images surrounding the locations of 470 spectroscopically classified supernovae. A random forest classifier uses an ensemble of decision trees which each classify based on a subset of galaxy properties, leading to a collective decision. With this algorithm as a baseline, we develop a convolutional neural network trained directly on host galaxy images. A convolutional neural network works by taking a filter matrix and convolving it with the input images, optimizing the filter matrix to best lead to accurate classification. It therefore has the advantage of being able to detect and make use of context properties which have not been noticed or defined by humans.

This paper is structured as follows. In Section 2 we introduce the Pan-STARRS dataset used to train our algorithms. In Section 3 we describe our methods, including the host galaxy detection and selection process in 3.1, the Random Forest Classifier in 3.2, and the Convolutional Neural Network in 3.3. In Section 4 we describe our results in both classes, in classifying between the five types as well in classifying Type Ias vs. core collapse types. In Section 5 we compare our algorithm to the previous Foley & Mandel (2013), in Section 6 we discuss the limitations of our algorithms, and we conclude in Section 7.

2. DATA DESCRIPTION

Our data comes from the Pan-STARRS1 (PS1) wide-field survey telescope in Haleakala, Hawaii. The telescope has a primary mirror of diameter 1.8m and a 1.4 gigapizel camera (GPC1) with a 7.1 deg^2 field of view and a 0.”258 pixel scale, made up of 60 4800 $\times$ 4800 pixel

detectors. The survey included $g_{P1}r_{P1}i_{P1}z_{P1}y_{P1}$ broadband filters, though we only use the $g_{P1}r_{P1}i_{P1}z_{P1}$ since the y_{P1} filter data has less depth and cadence frequency. Filter details can be found in Stubbs et al. (2010) and Tonry et al. (2012).

We used the results of its Medium-Deep Survey (PS1-MDS, 2010-2014). This survey covered around 70 deg^2 in ten single-pointing fields, taking up about 25% of observing time Chambers et al. (2016). Each filter excluding y_{P1} reached a 5σ depth of $\approx$ 23.3 magnitude per observation. In the y_{P1} filter, observations only reached a 5σ depth of $\approx$ 21.7 which is insufficient for our purposes, and therefore we do not use this filter.

The Pan-STARRS1 Image Processing Pipeline (IPP; Magnier et al. (2016a,b); Waters et al. (2016)) performed the reduction, astrometry, and stacking of the MDS images. Transients were then found using the `photpipe pipeline`, which has also been used for the SuperMACHO and ESSENCE surveys (Rest et al. (2005), Miknaitis et al. (2007)). See Rest et al. (2014), Scolnic et al. (2018), Jones et al. (2018) for further details of the analysis process. The images from all years of the survey, excluding the year in which the supernova occurred were then aligned and stacked to generate deep images of the background only. For example, for a supernova occurring in 2011, our stack would include all images from 2009, 2010, 2012, 2013, and 2014.

A total of 5235 likely supernovae were found (Jones et al. (2017, 2018)), more than 500 of which were spectroscopically classified using the MMT 6.5-m, Magellan 6.5, and Gemini 8-m telescopes and the SNID software package Blondin & Tonry (2007). Those of Types Ia, IIP, and I SLSNe were published by Jones et al. (2017), Sanders et al. (2015), and Lunnan et al. (2018) respectively. We train our algorithm on 470 events, of which 333 are of Type Ia, 16 are of Type Ibc, 85 are of Type II, 24 are of Type IIn, and 12 are superluminous. Note: we are trying to recover the remaining and will hopefully include over 500 in the final draft of this paper. We also identified a few other rare types of transients in the survey, but not in sufficient quantities for us to be able to meaningfully include them in our classification algorithm.

We realize there may be inaccuracies in the spectroscopic classification of our supernovae. We visually verified each instance of the types with smaller samples sizes, namely Type Ibc and superluminous, to make sure our smaller samples were sufficiently pure.

3. METHODS

3.1. *Host Detection*

Taking Pan-STARRS images of one square arc-minute surrounding the location of each classified supernova, we use `SEP` (Barbary (2016)), a Python library based on the `Source Extractor` software (Bertin & Arnouts (1996)), to detect host-galaxy candidates.

Though it is often clear from an image which galaxy is the host of a supernovae, at other times the host is more difficult to identify. At times the supernova is outside of the visible radius of its host, and multiple galaxies from different depths may all appear to be near the event location due to projection. Furthermore, at higher redshifts, the host may be too dim to detect in certain filters.

To ensure we are identifying the most likely host in ambiguous cases, we choose our host based on a formula from Bloom et al. (2002) to calculate the chance C_c that each galaxy is in the vicinity of the supernova by coincidence and therefore is not the host:

$$P_{cc} = 1 - e^{-\pi R_e^2 \sigma(\leq m)}.$$

R_e is the effective radius, given by

$$R_e = \sqrt{R^2 + 4R_h^2}.$$

R is the offset between the event and the center of the host candidate, and R_h is the radius of the host candidate. $\sigma(\leq m)$ is the density of galaxies brighter in magnitude than m, given by

$$\sigma = \frac{10^{0.33(m-24)-2.44}}{0.33\log(10)}.$$

The galaxy with the lowest chance of coincidence was chosen as the host by default. However, if the supernova location fell outside of the object with the lowest chance coincidence, and different object(s) were found to have a photometric redshift matching the redshift of the event, the object with the lowest chance coincidence that also had a matching redshift was used. Objects coincident with the location of a star identified in the Sloan Digital Sky Survey were excluded from host galaxy candidacy.

3.2. *Random Forest Classifier*

A random forest (RF) classifier (Breiman (2001)) is a machine learning algorithm which performs classification by developing a collection of decision trees. In each tree, a series of decisions, each decision based on a value of the input which in our case is properties of the host galaxy, create a path to a classification. The collection of trees, each trained on a subset of the values of the input, then vote to output a single type. This improves accuracy and decreases the risk of overfitting as compared with classifcation made by a single tree.

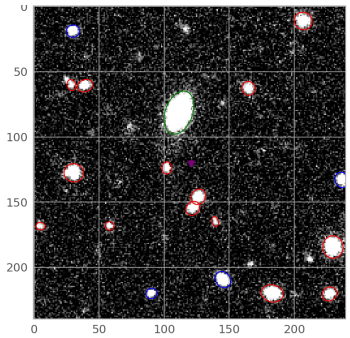

Figure 1. Example of host galaxy identification. Purple triangle marks location where the supernova occurred. Green oval marks the most likely host calculated from the chance coincidence. Objects circled in blue are stars identified using SDSS, all other objects are in red.

We used the `scikit-learn` python package to build our RF classifier. We found it to be successful when set to use 200 decision trees. All other parameters were left to the `scikit-learn` default values.

We had significantly uneven sample sizes for different types of supernovae, with many more Type Ias than the other types. Uneven samples would naturally lead to an algorithm biased towards classifying inputs as the more common types, since by doing so they could increase their overall accuracy. To avoid this problem, we used Synthetic Minority Over-sampling Technique (SMOTE, Chawla et al. (2002)) to create synthetic data to add to the less common classes. SMOTE randomly creates artificial data points falling on the line segments which

would connect the real input points. Since Type Ia is our largest class, other classes were up-sampled to match its size.

We extract the following known galaxy properties from the PS1-MDS images to form the inputs for the RF classifier: apparent and absolute magnitude, ellipticity, separation (distance between supernova location and host center), separation/$\sqrt{area}$, pixel rank, and the chance of coincidence of the most likely host. Event redshift was also included; see the Section 6 for a discussion of redshift use. Pixel rank was used as a measure of whether the event happened in a brighter or dimmer part of galaxy, which can be an indicator of whether the supernova occurred in a star-forming region of the galaxy. Pixel rank was calculated as the percent of host galaxy pixels which were dimmer than the pixel corresponding to the event location if the event was located within the host object, and 0 otherwise. Magnitudes from each of the 4 filters (g, r, i, z) were used, so color information is captured in the magnitude differences between the filters.

In any filter where the likely host galaxy could not be detected, the image's limiting magnitude + 1 was used for the magnitude. For images where no likely host could be detected to a 3σ threshold in any of our filter bands, we used average values, as is literature standard.

3.3. *Convolutional Neural Network*

We also built a convolutional neural network (CNN), a type of a deep neural network which works well for classification based on images. We trained it directly on the PS1-MDS images of the 0.67 arcminute by 0.67 arcminute square surrounding the location where each supernova occurred. Unlike our RF algorithm, in which potentially valuable information contained in the images is lost in the extraction process, the CNN takes in the entire image directly and therefore can make use of potentially complex host galaxy properties and host-event correlations, even those which we might not think of.

The input consisted of images with 160 × 160 pixels capturing 0.67 × 0.67 arcminutes. Each pixel containing 6 values: the $g_{P1}r_{P1}i_{P1}z_{P1}$ values, the event redshift, and a value set to $1 - C_c$, the chance coincidence of the most-likely host galaxy if the pixels was part of the most likely host, and 0 otherwise.

CNNs work by taking an initially-random filter matrix, convolving it with the input image, and passing on the output matrix to the next layer of the neural network. We use 2 dimensional 3 by 3 convolution filters which are convolved across each layer of pixel values. In the architecture we found to work well, there are three convolution layers, each followed by a rectified linear unit (relu, Nair & Hinton (2010)) activation layer which zeros out negative values, and then a pooling layer, which reduces the output by taking the average of each 2x2 square of values. This is followed by an additional pooling layer taking the average of every 5x5 square of values. It then ends with two fully connected layers of neurons, which each take the dot product of its inputs and its weights vector, pass the output through a non-linear activation function (Softmax Jang et al. (2016)), and then output the result. The final output of the last fully connected layer corresponds to the pre-

diction of the supernova type. The algorithm then optimizes the filter matrices as well as the neuron weight to maximize classification accuracy. It does this by Adam optimization (Kingma & Ba (2014)) with learning rate 5×10^{-3}. We trained our data on batches of size 58 samples at a time, for 200 epochs. Note: I may further optimize hyperparameters; architecture may change in final draft.

We divided our data into training, validation, and testing sets, as is standard machine learning practice. Within the training set, we used image augmentation to balance our sample sizes for each type to prevent our algorithm from developing biases due to uneven sample sizes. We did this by adding additional randomly rotated and reflected copies of the less common classes from within the same training set. The test, validation, and training sets therefore still contain completely distinct events from one another. Images are all cropped from 240×240 pixels to 160×160 pixels after rotation to eliminate empty corners.

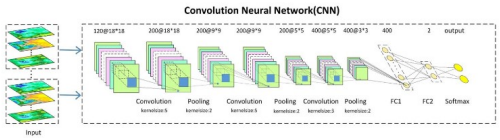

Figure 2. Convolutional Neural Network. Filter matrix (blue) is convolved with inputs in covolution layers, size is decreased by taking the average of square pools in pooling layers, fully connected layers (yellow) lead to a classification. NOTE: for my final paper this figure will be placed with a high-quality similar diagram which I create myself and reflects the specific architecture I will use. The current placeholder image was stolen from Zhang et al. (2018).

4. RESULTS

Note: Accuracy rates may change in final draft due to tuning adjustments.

4.1. *RF results*

The random forest classifier, using only known galaxy properties, serves as a baseline for of machine learning host galaxy based classification. With the random forest classifier, we were able to achieve an average accuracy per class of 0.47, which is significantly better than random which would yield 20%. Figure 4 shows our confusion matrix, a visualization of our success and misclassification rates for each of the types. The random forest classifier has significant success in classifying Type Ia and superluminous supernovae, with success rates of 77% and 89% respectively, but it struggles to classify Types Ibc and IIn, achieving only 0% and 7% accuracy in those categories.

We also separately trained it to classify just between Type Ia and other, and achieved 73% accuracy with this. The *feature importances* of a RF are a metric of how significant features of the input are to the algorithm for classification, relative to one another; features of higher importance were more heavily used in determining the output. Figure 3 shows a bar graph of our feature importances.

4.2. *CNN results*

With the convolutional neural network, we achieve an average accuracy per class of 63%, a significant improvement over the random forest. The main improvement mainly comes from an improvement in identifying Types IIn and Ibc, in which it achieved 100% and 75% accuracy, it performed worse at the other categories. Note: this analysis will likely change in the final draft because the algorithm easily trades off accuracy in one class for

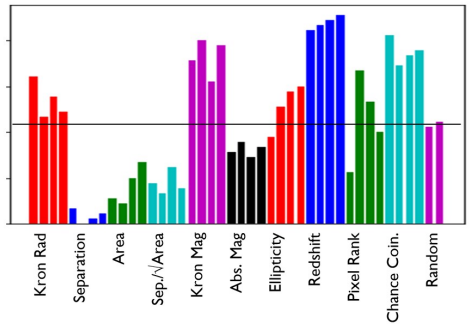

Figure 3. Feature Importances from RF Classifier. Importance values are relative and useful only for comparison with one another. Note: in the final draft of this paper I will improve the graph, ditinguishing between the four bars of each feature and create a legend. For each feature, the four bars represent the importance of the feature in the input corresponding to the g, r, i, z filters from left to right. In the case of redshift, the identical values were input with the information from each filter, so the differences between the redshift importances are due to noise within the classifier. Chosen host galaxy locations were cross-referenced between filters to ensure that the same host was chosen in all filters, though magnitudes, radii, etc. may still vary between filters. On the far right are two random values added for comparison, and their height is marked by the horizontal line. We can see that several properties, such as separation and area, were less important than the random number meaning they were unuseful for classification. Redshift, apparent magnitude (Kron mag), and chance coincidence are the most important features, followed by radius, pixel rank, and ellipticity. We do not have a good explanation for why the algorithm relied upon apparent magnitude and redshift instead of absolute magnitude, since these capture equivalent information.

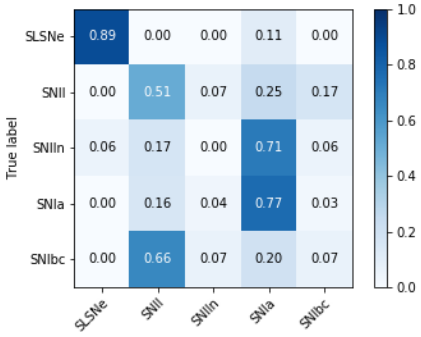

Figure 4. Random Forest Classifier Confusion Matrix showing success and misclassifications. Rows correspond to actual types and columns correspond to classifier predicted types. Numbered boxes indicate the fraction of row type which were classified as column type, e.g. 11% of SLSNe were misclassified as SNIa.

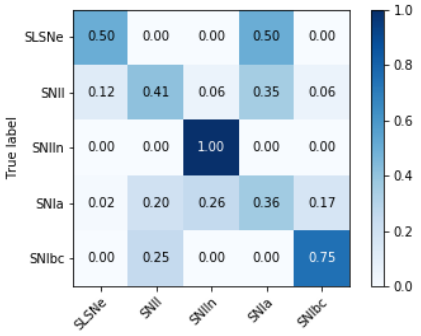

Figure 5. Convolutional Neural Network Confusion Matrix.

accuracy in another, and I will continue to tune the algorithm.

4.3. *Ia vs. Other*

We also separately trained our algorithms to classify between Type Ias and all other types, since this is a useful classification for cosmological research. We found similar success rates (73-74%) in both algorithms. This is significantly better than previous host galaxy based classification (see Section 5 for comparison).

5. COMPARISON TO PREVIOUS HOST-GALAXY BASED CLASSIFICATION

In their *galsnid* classification algorithm, Foley & Mandel (2013) use direct correlations between individual host-galaxy properties and supernova types to predict whether a supernova is of Type Ia or a core collapse type. Their algorithm is trained on LOSS survey data Leaman et al. (2011).

Both *galsnid* and our random forest classifier include in their input pixel rank, and offset. *Galsnid* additionally takes in K band magnitude and $B_0 - K$ color. We instead use *griz* magnitudes, from which the algorithm should be able to learn any colors which are combinations of those bands. Though Foley and Mandel use host galaxy morphology as one of their inputs, our sample is

of generally higher redshifts than theirs, and therefore do not have the resolution to morphologically classify the galaxies in our sample. However, we do include magnitude and we expect the algorithm to learn color, which should give it some information about morphology.

Our algorithm improves off Foley & Mandel (2013) firstly by extending to also classify between core collapse types of supernovae. Furthermore, their algorithm assumes all galaxy properties are uncorrelated, and they suggest that improvement could be made by taking into account correlations. Our random forest algorithm implements this improvement, as it makes no assumptions about relationships between properties or lack thereof. Machine learning is furthermore capable of detecting relationships between complex combinations of inputs and the corresponding outputs, giving it further advantage over the use of direct correlations.

The CNN receives even more complete information, allowing it to learn even more complicated correlations, since it receives the images themselves. It therefore has an advantage over both the random forest classifier and *galsnid* in that it can make use of galaxy properties which have not been noticed or defined by humans. This likely contributes to its higher success rate.

Foley and Mandel used a Figure of Merit (FoM) from Kessler et al. (2010) to quantify their success. It is given by efficiency times psuedopurity:

$$FoM = \epsilon_{Ia} \times PP_{Ia}$$

where efficiency is

$$\epsilon_{Ia} = N_{Ia}^{Sub}/N_{Ia}^{Tot}$$

with N_{Ia}^{Tot} being the actual number of Type Ia supernova in the sample, N_{Ia}^{Sub} being the number of Type Ia supernova which get correctly classified as Type Ia. Psuedopurity is given by

$$PP_{Ia} = \frac{N_{Ia}^{Sub}}{N_{Ia}^{Sub} + 5N_{Non-Ia}^{Sub}}$$

with N_{Non-Ia}^{Sub} being the number of non-Ia supernova incorrectly classified as Type Ias. By adjusting their classification weighting, they achieve a maximum FoM of 0.269 when weighting heavily towards classification as core collapse. Even without adjusting weighting to maximize our FoM, we achieved FoM of 0.45 with neutral weighting. This is a significant improvement from *galsnid*, likely due to the differences discussed earlier in this section.

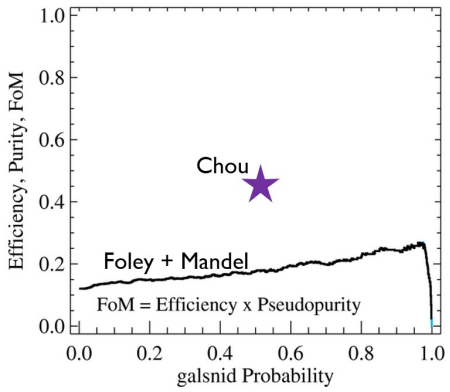

Figure 6. Comparison with Foley & Mandel (2013).

6. LIMITATIONS

WE use redshift in both of our classificatino algorithms. For the RF, this enables us to calculate absolute

magnitudes and distances, which in turn should enable the algorithm to pick up on trends between true physical galaxy properties and supernova types. For the data set we used to train the algorithms, we had access to the spectroscopically derived redshift of each supernova. Though we typically will not have spectroscopic redshifts for unclassified supernova or their hosts from future surveys, in LSST we will have access to photometric redshifts for all galaxies brighter than 27.5 magnitude, which we will be able to use instead. Less than 10% of these redshifts are expected to be outliers, with galaxies brighter than 25.3 magnitude having an expected root-mean-square scatter $\sigma_z/(1+z) \lesssim 0.05$. The additional uncertainty of photometric redshifts should not be a limiting factor for our algorithms.

Though our use of redshift would hypothetically inhibit our ability to classify events deeper than the training set, our algorithms would probably not be useful on such events anyway since the host galaxies at such high redshifts would not be well visible in images.

Another limitation to our algorithms is that they can only classify between the five types. They could easily be extended through retraining on a data set including other types, which can be done quickly and easily. They would still, however, be unable to identify new and unforeseen classes.

We do not take into account any biases that may exist in our spectroscopically classified training set. Though this could be addressed in the random forest classifiers by adjusting weighting within the training set, there is no clear solution for our convolutional neural network.

Our classification likely creates biased samples of each of the supernova types if correlations exist between properties of supernova of a given type and their host galaxie. For example, Type Ias occurring in star-forming galaxies could have differences in typical properties from Type Ias occurring in quenched galaxies. If using sets of supernova created using our algorithms, one would need to be mindful of biases caused by the nature of our classification. Clearly, samples created using our algorithm should not be used to investigate correlations between supernovae and host galaxies.

We are also limited by our small sample size for certain classes, especially the superluminous supernovae. This makes it difficult for our algorithms to learn trends. This limitation should be eliminated once a significant number of supernovae have been discovered by LSST and classified, creating larger samples at high redshifts.

We furthermore face the inherent physical limitations of classifying supernovae using only contextual information. The context of a supernova does not completely determine its type. Even if perfectly using all available contextual information, it would still be impossible for us to reach 100% accuracy. This can be partially addressed by combining our algorithm with photometric methods, which is an important future direction for our algorithm. Our classification methods should be able to significantly improve photometric methods especially early in the supernova, when photometric datapoints are limited.

7. CONCLUSIONS

Optimizing non-spectroscopic methods of supernova classification is currently important, as the rate of supernova discovery increases beyond the capacities ofr spectroscopic classification. It is well-known that supernova types correlate with various host galaxy properties. We develop two machine learning algorithms to classify supernova solely based on contextual data. To take advantage of these correlations, we train our algorithms on 470 spectroscopically classified supernova from the Pan-STARRS Medium-Deep Survey. Our first algorithm is a random forest classifier, which we train on extracted host galaxy properties including magnitude, ellipticity, offset between host galaxy and supernova location, and others. We use SMOTE resampling to balance the sample sizes of each type. This algorithm achieves an average accuracy per class of 45% and serves as a baseline with which to compare our second and more successful algorithm, a convolutional neural network. This is trained directly on PS1-MDS images of the area surrounding the supernova, as well as redshift and a mask conveying the most likely host and its chance of coincidence. It uses rotations and reflections to balance the sample sizes of each type. It achieves a significant average accuracy per class of 60%, performing better on Types IIn and Ibc. Our algorithm is the first that we are aware of to use contextual data to classify between all 5 types of supernovae, and it significantly improves off of previous host-galaxy based Ia versus core collapse supernova classification Foley & Mandel (2013). Future directions include combination with photometric classification methods.

Software: Astropy (Price-Whelan et al. (2018)), Keras (Chollet et al. (2015)), Matplotlib (Hunter (2007)), NumPy (Van Der Walt et al. (2011)), Pandas (McKinney (2010)), Scipy (Oliphant (2007)), Scikit-learn (Pedregosa et al. (2011)), SEP (Barbary (2016), Bertin & Arnouts (1996)), Tensorflow (Abadi et al. (2015))

REFERENCES

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/

Barbary, K. 2016, J. Open Source Software, 1, 58

Bertin, E., & Arnouts, S. 1996, Astronomy and Astrophysics Supplement Series, 117, 393

Blondin, S., & Tonry, J. L. 2007, The Astrophysical Journal, 666, 1024

Bloom, J. S., Kulkarni, S. R., & Djorgovski, S. G. 2002, The Astronomical Journal, 123, 1111, doi: 10.1086/338893

Boone, K. 2019, arXiv preprint arXiv:1907.04690

Breiman, L. 2001, Machine learning, 45, 5

Chambers, K. C., Magnier, E., Metcalfe, N., et al. 2016, arXiv preprint arXiv:1612.05560

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. 2002, Journal of artificial intelligence research, 16, 321

Chollet, F., et al. 2015, Keras, https://keras.io

Foley, R. J., & Mandel, K. 2013, ApJ, 778, 167, doi: 10.1088/0004-637X/778/2/167

Hunter, J. D. 2007, Computing in science & engineering, 9, 90

Jang, E., Gu, S., & Poole, B. 2016, arXiv preprint arXiv:1611.01144

Jones, D., Scolnic, D., Riess, A., et al. 2017, The Astrophysical Journal, 843, 6

—. 2018, The Astrophysical Journal, 857, 51

Kessler, R., Bassett, B., Belov, P., et al. 2010, Publications of the Astronomical Society of the Pacific, 122, 1415

Kingma, D. P., & Ba, J. 2014, arXiv preprint arXiv:1412.6980

Leaman, J., Li, W., Chornock, R., & Filippenko, A. V. 2011, Monthly Notices of the Royal Astronomical Society, 412, 1419

Lochner, M., McEwen, J. D., Peiris, H. V., Lahav, O., & Winter, M. K. 2016, The Astrophysical Journal Supplement Series, 225, 31

Lunnan, R., Chornock, R., Berger, E., et al. 2018, The Astrophysical Journal, 852, 81

Magnier, E., Schlafly, E., Finkbeiner, D. P., et al. 2016a, arXiv preprint arXiv:1612.05242

Magnier, E. A., Chambers, K., Flewelling, H., et al. 2016b, arXiv preprint arXiv:1612.05240

McKinney, W. 2010, in Proceedings of the 9th Python in Science Conference, ed. S. van der Walt & J. Millman, 51 – 56

Miknaitis, G., Pignata, G., Rest, A., et al. 2007, The Astrophysical Journal, 666, 674

Möller, A., & de Boissière, T. 2019, arXiv preprint arXiv:1901.06384

Muthukrishna, D., Narayan, G., Mandel, K. S., Biswas, R., & Hložek, R. 2019, arXiv preprint arXiv:1904.00014

Nair, V., & Hinton, G. E. 2010, in Proceedings of the 27th international conference on machine learning (ICML-10), 807–814

Oliphant, T. E. 2007, Computing in Science & Engineering, 9, 10

Pasquet, J., Pasquet, J., Chaumont, M., & Fouchez, D. 2019, Astronomy & Astrophysics, 627, A21

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, Journal of machine learning research, 12, 2825

Price-Whelan, A. M., Sipőcz, B., Günther, H., et al. 2018, The Astronomical Journal, 156, 123

Rest, A., Stubbs, C., Becker, A. C., et al. 2005, The Astrophysical Journal, 634, 1103

Rest, A., Scolnic, D., Foley, R. J., et al. 2014, The Astrophysical Journal, 795, 44

Sanders, N. E., Soderberg, A. M., Gezari, S., et al. 2015, The Astrophysical Journal, 799, 208

Scolnic, D., Jones, D., Rest, A., et al. 2018, The Astrophysical Journal, 859, 101

Stubbs, C. W., Doherty, P., Cramer, C., et al. 2010, The Astrophysical Journal Supplement Series, 191, 376

Tonry, J., Stubbs, C. W., Lykke, K. R., et al. 2012, The Astrophysical Journal, 750, 99

Van Der Walt, S., Colbert, S. C., & Varoquaux, G. 2011, Computing in Science & Engineering, 13, 22

Villar, V., Berger, E., Miller, G., et al. 2019, arXiv preprint arXiv:1905.07422

Waters, C., Magnier, E., Price, P., et al. 2016, arXiv preprint arXiv:1612.05245

Zhang, W., Jiang, L., & Han, L. 2018, in Ninth International Conference on Graphic and Image Processing (ICGIP 2017), Vol. 10615, International Society for Optics and Photonics, 106155E