The Astrophysical Journal Supplement Series, 255:24 (11pp), 2021 August https://doi.org/10.3847/1538-4365/ac0893
© 2021. The American Astronomical Society. All rights reserved.

A Deep-learning Approach for Live Anomaly Detection of Extragalactic Transients

V. Ashley Villar[1,2,3,4,10], Miles Cranmer[5], Edo Berger[6,7], Gabriella Contardo[8], Shirley Ho[8], Griffin Hosseinzadeh[6,7], and Joshua Yao-Yu Lin[9]
[1] Department of Astronomy, Columbia University, New York, NY 10027, USA; vav2110@columbia.edu
[2] Department of Astronomy & Astrophysics, The Pennsylvania State University, University Park, PA 16802, USA
[3] Institute for Computational & Data Sciences, The Pennsylvania State University, University Park, PA, USA
[4] Institute for Gravitation and the Cosmos, The Pennsylvania State University, University Park, PA 16802, USA
[5] Princeton University, Princeton, NJ 08544, USA
[6] Center for Astrophysics | Harvard & Smithsonian, 60 Garden Street, Cambridge, MA 02138-1516, USA
[7] The NSF AI Institute for Artificial Intelligence and Fundamental Interactions, 77 Massachusetts Avenue, Cambridge, MA 02139, USA
[8] Flatiron Institute, New York City, NY 10010, USA
[9] University of Illinois at Urbana-Champaign, Urbana, IL 61820, USA
Received 2021 March 17; revised 2021 June 2; accepted 2021 June 3; published 2021 August 10

Abstract

There is a shortage of multiwavelength and spectroscopic follow-up capabilities given the number of transient and variable astrophysical events discovered through wide-field optical surveys such as the upcoming Vera C. Rubin Observatory and its associated Legacy Survey of Space and Time. From the haystack of potential science targets, astronomers must allocate scarce resources to study a selection of needles in real time. Here we present a variational recurrent autoencoder neural network to encode simulated Rubin Observatory extragalactic transient events using 1% of the PLAsTiCC data set to train the autoencoder. Our unsupervised method uniquely works with unlabeled, real-time, multivariate, and aperiodic data. We rank 1,129,184 events based on an anomaly score estimated using an isolation forest. We find that our pipeline successfully ranks rarer classes of transients as more anomalous. Using simple cuts in anomaly score and uncertainty, we identify a pure (≈95% pure) sample of rare transients (i.e., transients other than Type Ia, Type II, and Type Ibc supernovae), including superluminous and pair-instability supernovae. Finally, our algorithm is able to identify these transients as anomalous well before peak, enabling real-time follow-up studies in the era of the Rubin Observatory.

Unified Astronomy Thesaurus concepts: Supernovae (1668); Time series analysis (1916); Astrostatistics techniques (1886); Surveys (1671)

1. Introduction

Wide-field optical surveys such as the Asteroid Terrestrial-impact Last Alert System (Jedicke et al. 2012), the All-Sky Automated Survey for SuperNovae (Shappee et al. 2014), the Panoramic Survey Telescope and Rapid Response System 1 (Pan-STARRS1; Chambers et al. 2016), and the Zwicky Transient Facility (ZTF; Bellm et al. 2018) have exponentially increased the discovery rate of new transient events that vary on day to year timescales. The upcoming Vera C. Rubin Observatory (Ivezić et al. 2019) and its decade-long Legacy Survey of Space and Time (LSST) will greatly accelerate this discovery rate to millions of new transient events annually. However, a limited fraction (likely $\lesssim$0.1%) of all events can be followed up with dedicated spectroscopic and multiwavelength campaigns. Identifying transients worthy of follow-up will be akin to finding needles in a haystack. Adding to the challenge, we will need to identify such events quickly to capture events pre- or near peak to fully optimize the efficiency of follow-up campaigns.

Over the past few years, there have been several initial efforts aimed at photometrically classifying transients to build pure samples of known transient classes (Boone 2019; Muthukrishna et al. 2019; Pasquet et al. 2019; Gómez et al. 2020; Hosseinzadeh et al. 2020; Villar et al. 2020; Sánchez-Sáez et al. 2021). However, even the rarest transients known today, like superluminous supernovae (SLSNe) and tidal disruption events (TDEs), will be discovered by the thousands in the era of LSST (e.g., Villar et al. 2018; Bricman & Gomboc 2020). Detection and classification algorithms sensitive to anomalous transients are essential in order to discover unexpected and even rarer phenomena.

There is a growing literature on anomaly detection for astronomy applications. For supernova (SN) light curves, Pruzhinskaya et al. (2019), Aleo et al. (2020), Martínez-Galarza et al. (2020), and Ishida et al. (2021) used isolation forests and active anomaly discovery on archival data sets. Convolutional autoencoders have recently been used to search for anomalies and glitches in gravitational-wave time series (Morawski et al. 2021). In the broader machine-learning literature, there has been increasing interest in anomaly detection in real time (Chalapathy & Chawla 2019). Typically, these works focus on long and well-sampled single-channel time series with anomalous periods of activity (e.g., Zhang et al. 2018; Li et al. 2019), although some recent work has focused on multivariate series (Zhao et al. 2020). Martínez-Galarza et al. (2020) recently presented a survey of anomaly detection algorithms for univariate, variable light curves. There has been limited focus on anomaly detection in irregularly sampled, aperiodic, multivariate time series. Recently, Soraisam et al. (2020) presented a real-time method to search for anomalies in multivariate data, trained and tested on variable sources. Similarly, Malanchev et al. (2021) presented a dedicated search for anomalies in the ZTF data stream utilizing human-engineered features from complete light curves and four different anomaly detection algorithms.

[10] Simons Junior Fellow.

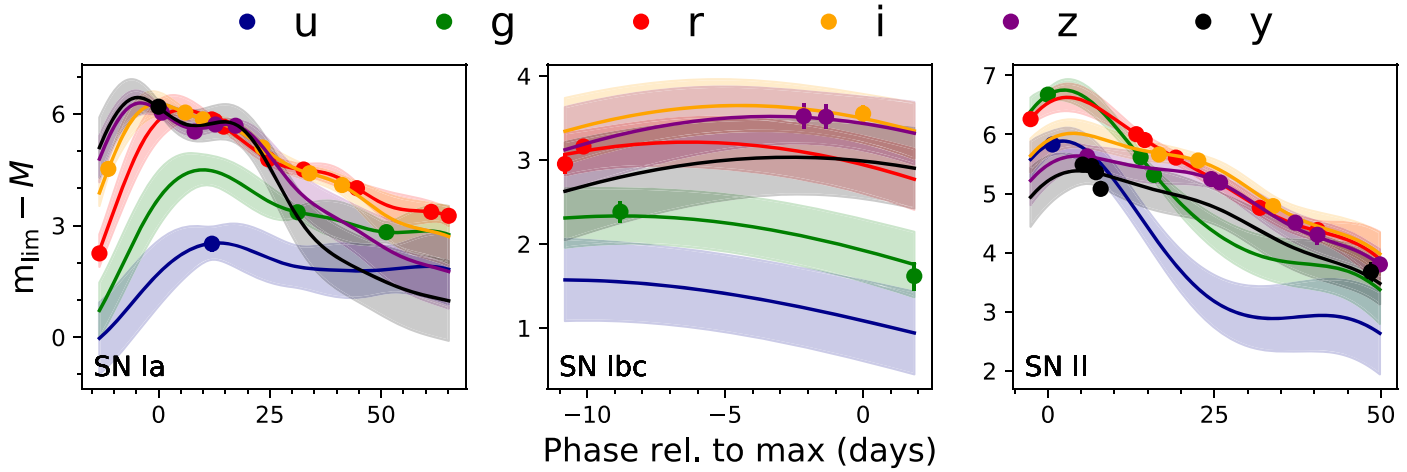

Figure 1. Sample *grizY* light curves of majority SN classes (Types Ia, Ibc, and II). Bold lines represent the 2D GP mean function, with shaded regions representing 68% confidence intervals. Even when entirely missing one or more bands, our method is able to produce reasonable interpolated light curves. Note that the *y*-axis, the magnitude used to train the GP, is designed such that the light curve tends to zero as it approaches the survey magnitude limit.

In this paper, we focus on out-of-distribution anomalies, which appear distinct from all other known transients in some feature space. By taking a completely data-driven approach to anomaly searches, our algorithm is agnostic to physics and therefore sensitive to entirely unexpected phenomena. Our anomaly detection pipeline is based on a variational recurrent autoencoder neural network (VRAENN) with no physical priors, and we search the learned encoded space for out-of-distribution events. The paper is organized as follows. In Section 2, we review the Photometric LSST Astronomical Time-Series Classification Challenge (PLAsTiCC) data set used for training and anomaly detection and the breakdown of SN-like transients used in this study. In Section 3, we present our anomaly detection pipeline and the VRAENN architecture. We discuss our results in Section 4 and conclude in Section 5.

2. Data Set and Preprocessing

In this study, we use PLAsTiCC, a simulation of 3 yr of Rubin Observatory data that includes over 3.5 million transient events from 18 unique physical classes, extending to a redshift of $z \approx 1.5$ (Allam et al. 2018; Kessler et al. 2019). Each event is observed across six broadband filters (*ugrizY*) following the LSST observing strategy at the time the simulation was produced. Sample light curves are shown in Figure 1. Along with the light curves, PLAsTiCC provides metadata including the redshift, Milky Way reddening, physical parameters used to generate the model, and a realistic photometric redshift estimate (see Kessler et al. 2019 for details). For this observing strategy, each event is observed every few days (in any filter) and roughly once a week in the same filter.

The PLAsTiCC data set was originally created for a public data science (Kaggle) competition[11] to classify transients. We repurpose this data set as a training set for anomaly detection in an LSST-like data stream. Here, anomalous events will be determined by the metadata (i.e., if the event comes from a rare astrophysical origin). We remove classes observable only within the Milky Way (e.g., variable stars) and SNe with fewer than three detections in any filter within 300 days of peak brightness. This cut is not necessary, as our algorithm can take light curves of any length; however, light curves with fewer points are very unlikely to be selected for detailed follow-up in reality. We note that extragalactic light curves will be contaminated by Galactic astrophysical sources, but it is straightforward to separate extragalactic events from Galactic events given a photometric redshift estimate (or more sophisticated methods; e.g., Sánchez-Sáez et al. 2021); random associations with unrelated hosts are not modeled in PLAsTiCC. Additionally, we only utilize transients from the wide-fast-deep (WFD) survey and remove events within the deep drilling fields. In total, our data set contains 1,129,184 extragalactic light curves from 13 classes.

1. *Normal Type Ia SNe* arise from the thermonuclear explosions of carbon–oxygen white dwarfs. The models were generated using the standard SALT-II light-curve models (Guy et al. 2007), conditioned on ≈500 light curves from the Joint Lightcurve Analysis (Betoule et al. 2014). Type Ia SNe represent 54.2% of our data set. We consider Type Ia SNe part of the majority classes.
2. *Type II SNe*[12] are the explosions of massive stars that have retained their hydrogen envelopes. They are often characterized by long plateaus in their optical light curves. The models were generated from spectral energy distribution (SED) templates (Kessler et al. 2010; Anderson et al. 2014; Galbany et al. 2016; Sako et al. 2018). Type II SNe make up 25.8% of our data set. We consider Type II SNe part of the majority classes.
3. *Type Ibc SNe* are core-collapse SNe of stars with stripped hydrogen (Ib) and helium (Ic) envelopes. The models were generated using a combination of MOSFiT (Villar et al. 2017; Guillochon et al. 2018) and SED templates (Kessler et al. 2010). Type Ibc SNe make up 5.8% of our sample. We consider Type Ibc SNe part of the majority classes.
4. *Type I SLSNe* are luminous, hydrogen-free events thought to be powered by rapidly spinning, highly magnetized neutron stars. The SLSN models were produced using MOSFiT (Nicholl et al. 2017; Guillochon et al. 2018; Villar et al. 2018). They make up 2.2% of our sample.

[11] https://www.kaggle.com/c/PLAsTiCC-2018

[12] In the original version, PLAsTiCC grouped normal Type II and Type IIn SNe into a single class. We separate these classes due to their distinct physical origins and unique light curves.

We consider SLSNe to be members of the minority classes.

5. *Type Iax* are irregular Type Ia SNe with typically lower luminosities and velocities compared to normal Type Ia SNe (Li et al. 2003). The models were generated using available data in the Open Supernova Catalog (Guillochon et al. 2017). Type Iax SNe make up 1.8% of our data set. We consider Type Iax SNe to be members of the minority classes.
6. *Type IIn SNe*[12] are core-collapse SNe mainly powered by the interaction of the SN ejecta with circumstellar material (CSM). The models were generated using MOSFiT (Villar et al. 2017; Guillochon et al. 2018; Jiang et al. 2020). Type IIn SNe make up 1.8% of our sample. We consider Type IIn SNe to be members of the minority classes.
7. *Type Ia-91bg* are fainter, faster, and redder Type Ia SNe that make up ≈20% of the volumetric Type Ia sample (Filippenko et al. 1992; Graur et al. 2017) and ≈3% of the observational sample (Li et al. 2011). The model light curves are based on the SED templates from Nugent et al. (2002). Type Ia-91bg SNe make up 1.2% of our sample. We consider Type Ia-91bg SNe to be members of the minority classes.
8. *The TDEs* result from the tidal disruption of stars by supermassive black holes (SMBHs; Rees 1988). The TDE models were generated using MOSFiT (Guillochon et al. 2018; Mockler et al. 2019) and make up 0.6% of our sample. We consider TDEs to be members of the minority classes.
9. *The Ca-rich transients* (CARTs) are intermediate-luminosity transients whose spectra appear rich in calcium (Kasliwal et al. 2012). CARTs are modeled using MOSFiT, assuming they are powered by the radioactive decay of ^{56}Ni. We note that this is the same model used to generate Type Ibc SNe but with a distinct parameter space. CARTs make up 0.31% of our sample. We consider CARTs to be members of the minority classes.
10. *Intermediate-luminosity optical transients* (ILOTs) are transients that are brighter than novae but less luminous than SNe (Kasliwal 2012). In this case, we assume that ILOTs arise from CSM interaction with low-energy eruptions (or explosions) of massive stars. The ILOTs have been modeled using MOSFiT (Guillochon et al. 2017; Villar et al. 2017; Jiang et al. 2020) and represent 0.08% of our sample. We consider ILOTs to be members of the minority classes.
11. *Pair-instability SNe* (PISNe) are the explosions of low-metallicity massive stars ($M_{\rm ZAMS} \sim 130$–$260\ M_{\odot}$) that reach core temperatures high enough to form electron–positron pairs (Kasen et al. 2011). Compared to normal core-collapse SNe, PISNe have high kinetic energies and larger ejecta masses. The PISNe are modeled using MOSFiT, assuming that they are powered by the radioactive decay of ^{56}Ni. The PISNe make up 0.07% of our sample. We consider PISNe to be members of the minority classes.
12. *Kilonovae* (KNe) arise from the formation of radioactive rapid neutron capture elements in binary neutron star (and potentially neutron star–black hole) mergers. The models are based on theoretical calculations (Kasen et al. 2017). There are only two KNe in our sample. The KNe are dim and short-lived, making them nearly impossible for LSST to discover in the WFD strategy explored here. We consider KNe to be members of the minority classes.
13. *Active galactic nuclei* (AGN) refer generally to galaxies with active SMBHs from accreting gas. The AGN have a wide range of observed behavior, although they typically vary on timescales of weeks to years at the $\lesssim$10% level. The AGN variability is modeled using a damped random walk as described in MacLeod et al. (2011). The AGN make up 6.3% of our data set. We note that AGN are the only class that is likely not representative of the true LSST data stream; AGN will be much more numerous, with likely millions in the complete sample. Many bright AGN (which are those represented in PLAsTiCC) will be identified within the first year of LSST; however, transient bright flares will be of interest to the community (Graham et al. 2017). Because AGN are distinct from SN-like transients, we consider their effects separately in the Appendix.

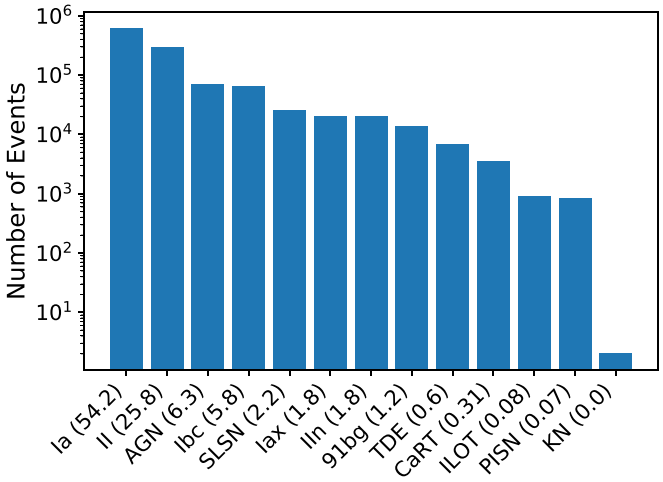

Figure 2. Breakdown of the various transient classes used in this study. Note that AGN are likely highly underrepresented compared to the true breakdown in the LSST data stream; however, AGN are exceptional in their light-curve properties and may be identified early in the survey. The parenthetical numbers associated with each class represent the percentage breakdown. There are a total of two KNe in our sample.

The breakdown of the various classes is shown in Figure 2. The observational rates of each class are a combination of the volumetric rates (intrinsic rarity) and observational effects (e.g., luminosity function, duration). The breakdown used here is designed to match what is empirically expected from a wide-field survey such as LSST (see, e.g., Perley et al. 2020; Villar et al. 2020). We note that this LSST cadence simulation detects just two KNe that pass our cuts in the simulated observations, highlighting the need for target-of-opportunity observations to better capture these rare events.

We define Type Ia, Type II, and Type Ibc SNe as the majority classes because together they make up the bulk of the data set (≈86%). We consider all other classes (excluding AGN) to be minority classes, with fractions of $\lesssim$2% of the sample.

We preprocess the data as follows. Following Villar et al. (2020), we scale the light-curve magnitudes such that zero corresponds to the magnitude limit. This is to aid the Gaussian process (GP) interpolation (which will tend toward zero before and after the SN). We correct the light curve of each event for

time dilation and convert to absolute magnitudes based on the provided photo-z values (see Villar et al. 2020 for details). The photo-z values are based on host galaxy association and use a color-matched nearest-neighbor method presented in Graham et al. (2018). This method is trained on a realistic sample of galaxies that would have spectroscopic redshifts available. About 17% of the redshift estimates are outliers, defined as $|z_{\rm true} - z_{\rm phot}|/(1 + z_{\rm phot}) > 3\sigma_{\rm IQR}$ by Kessler et al. (2019), where $\sigma_{\rm IQR}$ is the typical error for galaxies near $z_{\rm true}$. We make no cuts on redshift uncertainty; instead, we account for redshift uncertainties via a simple Monte Carlo method discussed in Section 3. We caution that our method relies on these host photo-z estimates. Hostless transients will therefore be excluded from detection.

We additionally correct the light curves for Galactic reddening, as outlined in Villar et al. (2020). Finally, we temporally shift each event such that the observed time of peak brightness (in any filter) is considered $t = 0$. This means that as new data are taken during an event's rise, the time of peak luminosity will continue to shift until the true peak luminosity has been observed or the event dims. For example, if a transient first peaks in the g band 10 days postexplosion and then reaches a brighter luminosity in the r band at 15 days postexplosion, the peak used by the neural network will be at 15 days postexplosion; this is independent of the bolometric peak luminosity. We find that this prescription of phase (versus, for example, time since first detection) leads to better performance in the autoencoder.

2.1. Interpolation Using GPs as Preprocessing

The PLAsTiCC light curves are irregularly sampled across time and filters, with no more than one filter observed at any given time. For the architecture discussed below, we require a flux and error estimate for each filter at every observation time. To produce this information, we use a 2D GP to interpolate the light curve over time and filter, with a multivariate Gaussian (MVG) kernel described by

$$\kappa(t_i, t_j, f_i, f_j; \sigma, l_t l_f) = \sigma^2 \times \exp\left[-\frac{(t_i - t_j)^2}{2l_t^2}\right] \times \exp\left[-\frac{d(f_i, f_j)^2}{2l_f^2}\right], \quad (1)$$

where f represents the six ($ugrizY$) filters; l_t and l_f are the characteristic correlation length scales in time and filter, respectively; and $d(f_i, f_j)$ is the Wasserstein-1 distance between each filter's normalized throughput, which we optimize to each specific light curve. This choice in the distance metric loosely measures the similarity between two distributions. Mathematically, we treat each filter as a density function in wavelength. This distance metric is minimized when filters overlap and simplifies to a difference between central wavelengths in the limit of infinitely narrow passbands.

The GP interpolation both accounts for and produces error estimates for each flux measurement. The choice of a Gaussian kernel is physically well motivated in this case; in the Arnett model, an SN light curve is loosely described as the convolution between an input luminosity function and a Gaussian filter whose width is set by a diffusion timescale (Arnett 1982). Furthermore, without using a 2D GP (i.e., if the filters were not correlated), events that were unobserved in a given passband would be filled with the mean function (the limiting magnitude in our case). The 2D GP is therefore necessarily to produce reasonable light curves. Flux uncertainties are utilized in both the encoding method and anomaly scoring steps of our algorithm (see Section 3) in order to make our algorithm robust to low-confidence outliers. Finally, we note that during testing, we implemented a similar (though less physically motivated) interpolation scheme in Villar et al. (2020) and found that the interpolation methods led to visually similar light curves. The data used in that work, the Pan-STARRS1 Medium Deep Survey, have a similar cadence to the light curves explored here. We also note that a similar method has already been applied to PLAsTiCC data for classification (Boone 2019). Because this preprocessing is similarly employed on all light curves, we do not have reason to expect this step to significantly bias the results, even if the GP-interpolated light curves differ from the ground truth.

We implement the GP preprocessing using `sklearn`, optimizing the Gaussian width for each light curve independently via the `minimize` function from `scipy`, which uses the Broyden–Fletcher–Goldfarb–Shanno optimization algorithm (Fletcher 1986). We assume flat logarithmic priors over the wavelength (10^{-3}–$10^{4.5}$ Å) and temporal GP widths (10^{-6}–10^{4} days). Sample light curves are shown in Figure 1. Even in cases of poorly sampled light curves or light curves in which a band is completely missing, the GP produces reasonable flux and error estimates across all filters.

We note that our GP utilizes the complete light curve for interpolation. In reality, only the light curve up to the most recent observation will be available in real time. One may be concerned that because our GP has been conditioned on the entire light curve, it has more information than what will be available for real-time usage. This likely has no effect on our results, as each observation heavily anchors the GP prediction (see Figure 1), and the learned GP kernel sizes are similar to our priors (see, e.g., Villar et al. 2020 for typical SN values).

3. VRAENN: Architecture, Training, and Anomaly Detection

Following preprocessing of the training set, our anomaly detection pipeline includes two steps. First, we learn an encoded form of each light curve by training a VRAENN on the full data set. Then, we use an isolation forest to rank each light curve's encoded form by an anomaly score. We utilize a simple Monte Carlo to estimate the uncertainty on this score. In this section, we describe the VRAENN architecture, the training process, the isolation forest, and our error estimation method. Our code, along with all chosen hyperparameters of our model, is available via Github.[13]

Broadly speaking, a variational autoencoder is a probabilistic model used to encode high-dimensional data (Kingma & Welling 2013). Standard autoencoders simultaneously train an encoder and a decoder to learn a low-dimensional representation of the data set. However, the learned latent space is not guaranteed to be continuous (e.g., the same SN class may separate into several clusters in latent space), which is a beneficial property when searching for anomalies. Variational autoencoders solve this problem by learning smooth and continuous latent space by design.

Our novel VRAENN architecture is well suited to the problem of searching for unknown, anomalous transients. While more

[13] https://github.com/villrv/vraenn

traditional feature extraction via model fitting (e.g., Villar et al. 2019; Hosseinzadeh et al. 2020) or using predefined quantities (Boone 2019) has been successful for classification, dozens of features are required. We want to avoid searching for anomalous events in high-dimensional data, in which distance metrics are more challenging to meaningfully define (Liu et al. 2012). Additionally, the VRAENN architecture is built without any physical models, meaning that it is sensitive to new and unexpected physical processes, which are observationally distinct from known transients.

The architecture specifically used here has three additional benefits: (i) the ability to handle unevenly sampled light curves across time using recurrent neurons, (ii) the ability to produce extrapolated and interpolated light curves, and (iii) an insensitivity to noisy data and spurious outliers due to the variational architecture and error estimation.

Rather than using a static vector as a bottleneck, our VRAENN learns a distribution of encoding vectors. In particular, the encoding layer consists of two vectors: one that represents the mean of an MVG and one that represents its diagonal covariance matrix. For each SN passed into the VRAENN, we randomly select an encoding from the MVG defined by this learned mean and variance. The "variational" aspect of our architecture refers to this process of learning a distribution of encodings rather than a singular encoding. This is helpful in generating a smooth encoding space in which most events will cluster, allowing us to more easily pick out anomalous events. We emphasize that although the latent space is more well behaved, we do not claim that this space is interpretable. Villar et al. (2020) presented a similar method and compared the learned latent space to a number of hand-engineered features to highlight how such latent spaces can be correlated to observable properties.

The VRAENN architecture, based on the model presented in Villar et al. (2020), uses recurrent neurons to read in the GP light curve and estimated errors and encodes this light curve as a vector. This is achieved by encoding the light curve into a series of smaller matrices until the information reaches a small bottleneck layer of size 1×10. This is an unoptimized choice of size, although Villar et al. (2020) found a similar result after a hyperparameter search of a similar architecture. This layer is known as the encoded layer, and the layers preceding it are known as the "encoder." Each of these layers uses gated recurrent unit (GRU) neurons with a combination of hyperbolic tangent, sigmoid, relu, and linear activation functions. The GRU is a memory-efficient version of the long short-term memory, the standard for recurrent neural networks (Cho et al. 2014).

Before being passed into the second half of the autoencoder (the "decoder"), the encoded layer is repeated N times, with each time appended as a phase relative to maximum light. This can be thought of as evaluating an SN model at specific times, where the model is specified by 10 free parameters, with an 11th parameter specifying the time. The unique repeat layer of our architecture allows us to evaluate the light curve at times not included in the real data, i.e., interpolating and extrapolating the light curve if desired (although this feature is not used in this work).

The decoder then produces the light curves at the N times specified in the repeat layers. As with the encoder, the decoder uses GRU neurons with hyperbolic tangent activation functions. Figure 3 illustrates our full pipeline, including a schematic of the neural network architecture used. The VRAENN is optimized using a loss function combining the log of the weighed mean squared error and the standard Kullback–Leibler divergence (which measured how well our MVG represents our latent variable space):

$$\mathcal{L} = \log \sum_{i=1}^{N} \frac{[F_{i,\mathrm{True}}(t,f) - F_{i,\mathrm{Predicted}}(t,f)]^2}{N} + \sum_{i=1}^{N} - 0.5(1 + \log(\sigma_i^2) - \mu_i^2 - \sigma_i^2). \quad (2)$$

We minimize the loss function using the Adam optimizer (Kingma & Ba 2014) with standard learning parameters $\alpha = 10^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ for 1000 epochs using Keras (Chollet 2015) with a TensorFlow back end (Abadi et al. 2016). We train our VRAENN on 1% of the sample (12,159 events), reflecting (for example) the small data set that will be available within months of LSST coming online. The full model takes roughly 20 hr to train on a standard CPU. We note that once trained, our algorithm takes less than a second per object to encode the light curve, pass it through the VRAENN encoder, and calculate the associated anomaly score and uncertainty (assuming 10 random draws for error estimation).

A sampled subspace of our encoding vectors is shown in Figure 4. The majority of events cluster near zero, with anomalous events (like SLSNe) forming a cloud outside of the main distribution.

3.1. Scoring Anomalies with an Isolation Forest

Once our VRAENN is trained, we can encode any PLAsTiCC light curve, partial or complete, as a 1×10 vector. We then pass these encodings into an isolation forest (Liu et al. 2012). The isolation forest works by generating a series of decision trees over a random subset of attributes. Each tree recursively splits the set. Out-of-distribution anomalous events will be isolated with very few splits, while an average event will cluster with similar events, even after many splits. The number of splits is inversely related to an anomaly score. For the sake of interpretability, we then convert this raw score to a percentile. We use `sklearn` to implement the isolation forest using 1000 base estimators.

We identify several sources of possible error in the anomaly score.

1. Flux uncertainty due to Poisson noise. This is provided by PLAsTiCC as a standard deviation for each flux measurement.
2. Uncertainty in the flux estimates from the GP, which is also estimated by the GP.
3. Photometric redshift error, reported as a standard deviation. This affects the entirety of the light curve as both an overall multiplicative flux term and a time dilation adjustment.
4. Model uncertainty from the neural network converting the light curve into an encoded vector.

We account for the first three using a simple Monte Carlo method. For each transient, we generate 10 light curves that have photo-z and flux values drawn from Gaussian distributions described by the reported mean and errors. We do not account for noise from the neural network itself, which could be accomplished via an ensemble of networks. We find that, in general, the error estimated from this method is sufficient to eliminate anomalous events arising from poor data quality and

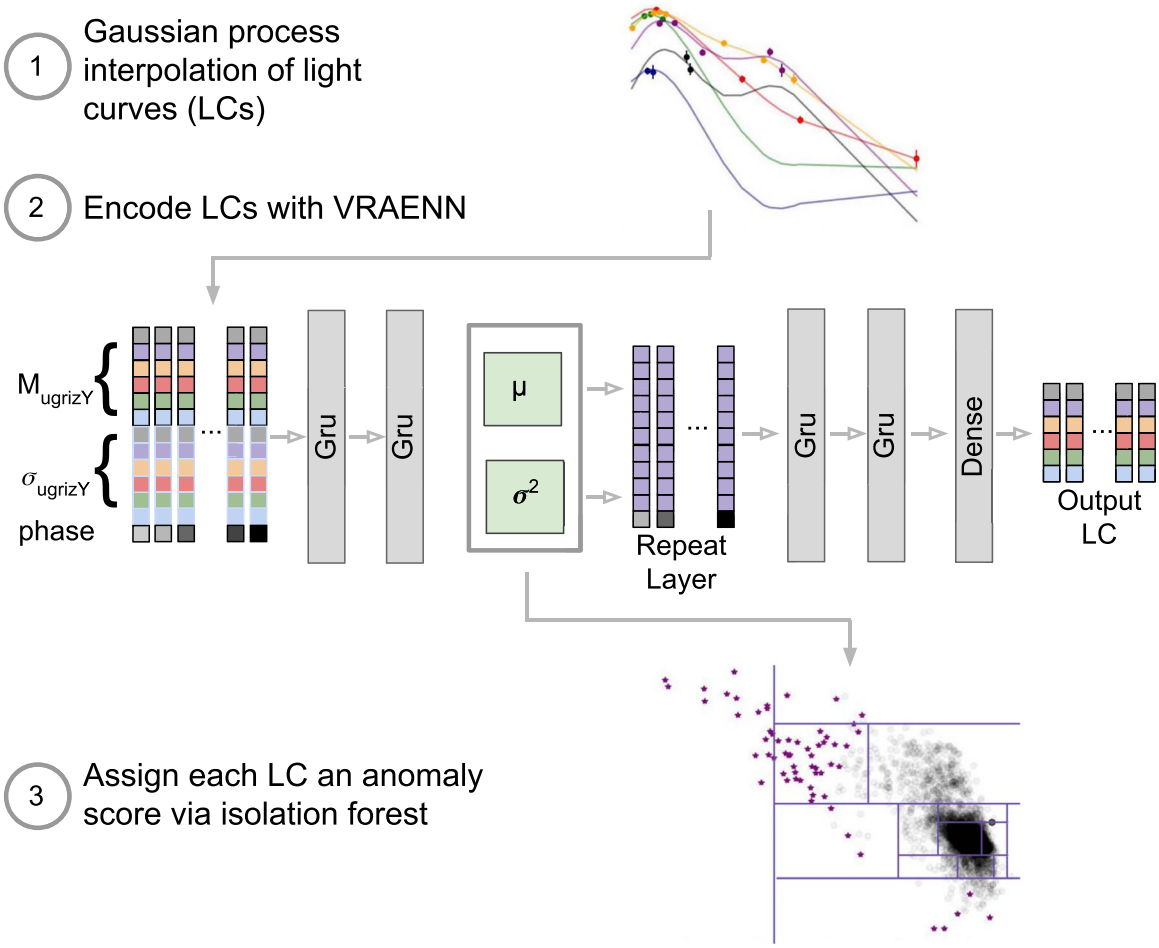

Figure 3. Summary of our anomaly detection pipeline. (1) We interpolate each *ugrizY* light curve using a 2D GP. (2) We train the variational recurrent autoencoder in an unsupervised manner. Light curves are represented as a time series, in which each epoch is represented by 13 features: six flux values, six estimated error values, and one time. This time series (consisting of N points) is encoded. The encoding layer is repeated N times, each time appended with the associated time value. We use this network to encode each light curve. (3) We use an isolation forest to assign an anomaly score for each light curve using the encoded vectors. The isolation forest is represented here in one subspace of the encoded space, with anomalous events highlighted as purple stars.

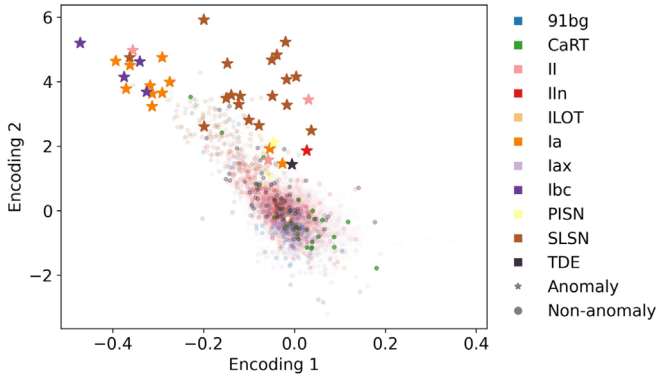

Figure 4. Scatter plot of representative encoding space for various classes. Events with high anomaly scores are shown as stars. Most events cluster near the origin, while more anomalous classes are seen as clouds outside the main distribution. The anomalous Type Ia SNe are those with incorrect photo-z estimates.

incorrect photo-z estimates by making a cut on the anomaly score uncertainty.

We show how the anomaly score uncertainty changes as a function of both time and number of data points in Figure 5. We first examine how the error changes as a function of number of observations. The error grows until the light curve

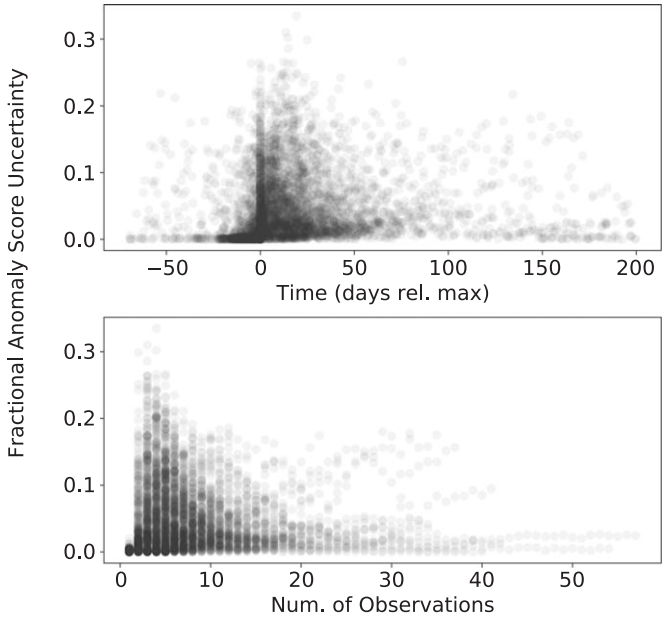

Figure 5. Time evolution of the relative anomaly score error for a representative subsample of our test set. In the upper panel, relative errors are low before peak luminosity and subsequently rise; however, the lower panel reveals that the relative error drops with an increased number of data points.

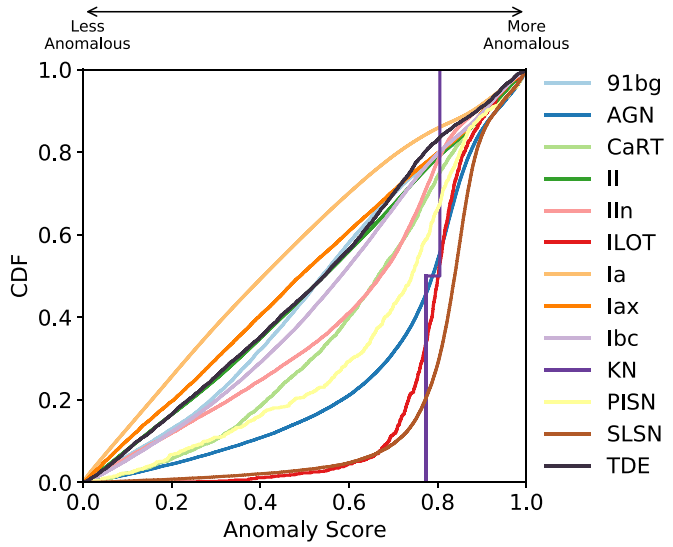

Figure 6. The CDF of the anomaly score for various classes of transients. The PISNe, SLSNe, ILOTs, KNe, and AGN are clustered toward the high end of the anomaly score.

reaches ≈three observations and then decreases. The error is initially small because the autoencoder, with limited information, returns the mean encoding vector from the training set (i.e., only one number is needed to encode a single observation). These light curves are not flagged as anomalous. However, once the light curve reaches a sufficient number of data points, the fractional uncertainty follows the expected trend of decreasing with increasing number of observations. When translated into fractional error versus time, the fractional error rises near peak (where most light curves have few data points) and declines at late times.

Finally, we additionally test utilizing the reconstruction loss (the mean square error of the VRAENN model versus the GP-interpolated light curve) of the autoencoder as an input feature. We find similar results as those discussed in the following section and the Appendix (i.e., anomalous samples dominated by SLSNe and AGN).

4. Results and Discussion

Our anomaly classification algorithm generates a ranked list of anomalous events given the full data set. We are interested in which events the anomaly detection pipeline ranks highly, but we are also interested in understanding when and why an event becomes anomalous.

We first explore the anomaly scores of the full test set. In Figure 6, we show the cumulative distribution function (CDF) for each class in our set. Even for events that would not be labeled as anomalous by our pipeline, the anomaly score distribution matches expectations. Type Ia–like SNe (including 91bg and Iax) are the least anomalous, on average. Type Ibc SNe and TDEs are largely distributed evenly across scores, with a slight tail at the upper end. While Type II SNe are, on average, more anomalous, only a small fraction have high anomaly scores. CARTs and Type IIn SNe are also more anomalous, on average. The PISNe, SLSNe, ILOTs, and AGN most drastically cluster at the high end of the anomaly scores, indicating that they are the most likely to be classified as anomalies with our algorithm. Only two KNe were in our sample; both had moderately high anomaly scores around the 80th percentile. Reassuringly, the events that prefer higher anomaly scores do not cluster in any obvious section of observational phase space (e.g., luminosity or duration), implying that our VRAENN has picked up on more fundamental features.

In practice, we will be limited to a small fraction (∼0.1%) of the LSST transients for follow-up. We therefore investigate several thresholds on the anomaly scores and uncertainties to search for anomalous events. The results presented here are summarized as histograms of anomaly sample breakdown and abundances in Figure 7. In short, our anomaly sample is pure, with $\lesssim$10% contamination from majority (Type Ia, Ibc, and II SNe) classes. The anomaly sample significantly overrepresents minority classes.

Many of the nonanomalous transients (e.g., Type Ia SNe) with high anomaly scores have fractional uncertainties on the anomaly scores of $\sigma_A/A \gtrsim 0.1$ (due to large uncertainties on photo-z estimates). We first make a strict cut on the full sample, looking only at events in the top 90th, 95th, and 99th percentiles of the anomaly scores and with $\sigma_A/A < 0.01$. Within these cuts, we find that SLSNe make up a majority of the remaining sample. For the 95th percentile cutoff, the anomalous SLSNe that pass our cut make up ≈5% of the original sample. In contrast, just 0.03% of the original input sample for Type Ia and Type II SNe remain in our 95th percentile anomaly score cutoff. In other words, our 95th percentile cutoff removed 95% of the SLSN sample but 99.97% of the Type Ia and Type II SN samples (leading to SLSNe being overrepresented in the anomalous sample by a factor of about 4.5). Similarly, PISNe make up a small fraction (≈1%) of our anomalous sample, but we retain ≈5% of the original sample in the 95th percentile cutoff. The results are similar for higher percentiles, with an even larger bias toward the more anomalous classes. This is highlighted as an abundance measurement in Figure 7.

We then explore a higher cutoff in the anomaly score fractional error, keeping events with $\sigma_A/A < 4 \times 10^{-4}$ (a threshold chosen by hand to maximize the ratio of rare events, shown in the rightmost column of Figure 7). Even without cuts on the anomaly score itself, SLSNe dominate the sample, followed by PISNe and Type Ia–like SNe.

Noting that stringent cuts in the anomaly score uncertainty lead to pure samples of minority classes, we investigate if incorrect photo-z estimates are responsible for false-positive detection of anomalous events in majority classes. We find that removing events with $|z_{\rm phot} - z_{\rm true}|/z_{\rm true} > 2$ does improve the purity of anomalous samples for cuts with $A > 99\%$ and 99.9%, although it has a lesser effect for the $A > 95\%$ cut. For all cuts, we find that removing events with incorrect photo-z estimates drastically decrease the number of Type Ia SNe that pass our anomaly thresholds. For example, in the $A > 95\%$ sample, the sample fraction of Type Ia SNe drops from ≈10% to ≈3%. Reducing the number of outlier photo-z estimates in LSST will likely substantially improve the purity of our anomaly sample.

We investigate why some Type Ia SNe are within the most anomalous events even with stringent cuts on the anomaly score and uncertainty. All of the events that pass our 99% cutoff for the anomaly score have catastrophically incorrect photo-z values. Specifically, we find that these events are typically injected at relatively low redshifts ($z_{\rm true} \lesssim 0.05$) yet have reported photometric redshifts $z_{\rm phot} \approx 2$ with small reported uncertainties—again highlighting the need for reliable photo-z estimates.

Our anomaly detection algorithm is biased toward bright events, which begs the question, Is our algorithm making a

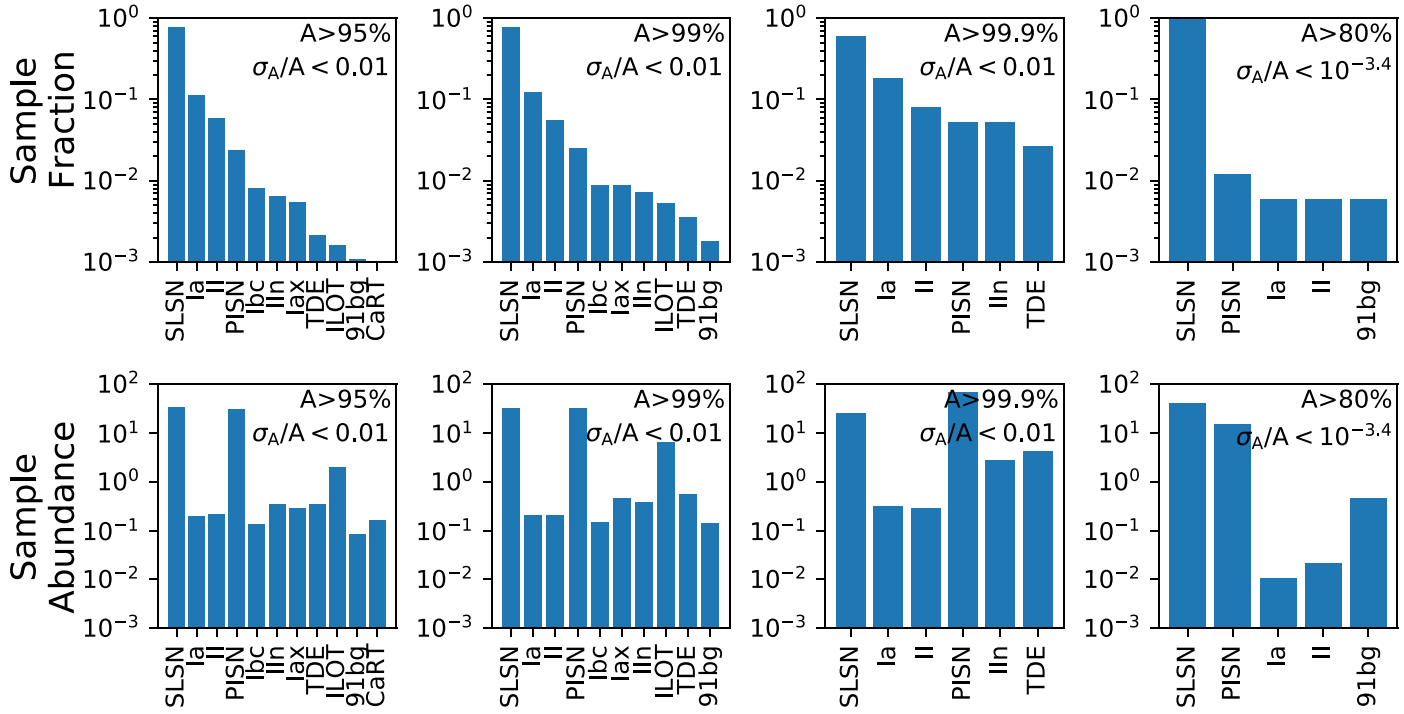

Figure 7. Top row: breakdown of anomalous events, given the cuts listed in each panel. The top number represents the anomaly percentile threshold, while σ_A/A refers to the anomaly score fractional error threshold. The rightmost panel shows optimized cuts on the anomaly score and uncertainty to minimize contamination from majority classes. Overwhelmingly, SLSNe make up the majority of our anomaly sample. Bottom row: relative abundance of the anomaly samples shown in the top row. Each panel shows the "abundance" (ratio between the number of "anomalous" events in each class and the expected number based on input fraction). The minority classes are heavily overrepresented, while the majority classes are underrepresented.

trivial cutoff on luminosity to search for anomalies? We test this hypothesis by making a simple baseline comparison. We rank each event by peak luminosity (in any filter), rise time (time from first detection to peak flux), and decay time (time from peak to final detection), excluding AGN. Such a baseline requires the full light curve and is therefore not an entirely fair comparison to our proposed model. If we keep the top 1% of the brightest events (neglecting timescales), 50% of the sample is made up of Type Ia SNe, and only ≈4% are SLSNe. We can further make a cut on the photometric redshift uncertainties. Even with extremely aggressive cuts ($\sigma_z/z < 0.05$) that remove 99.99% of the sample, only ≈10% of the sample are SLSNe. The plurality, ≈40%, are Type Ia SNe. Next, we include timescale cuts in our baseline. We keep transients within the bottom 10 and top 10 percentiles for the luminosity and rise and decay timescales; we find that more stringent cuts remove nearly all of the sample. In this case, we find that the plurality of the sample are Type Ia SNe (27%), with Type II and Ibc SNe and SLSNe each making up the next highest fractions, all at ≈15%. Given these results, our proposed algorithm is seemingly learning more complex features beyond peak flux (e.g., timescales) and performs better than simple filters alone.

4.1. Anomaly Detection in Live Streaming Data

Our analysis thus far has focused on the full light curves, rather than real-time follow-up. We next turn to how the anomaly scores evolve over time for a representative subsample of $\approx 10^3$ SNe. As shown in Figure 5, most events are identified near peak luminosity ($t \approx 0$). We focus on how anomaly scores vary for the majority classes versus the minority classes. Events from the minority classes are much more likely to be triggered as anomalous before peak. In our representative sample, ≈60% of Type Ia/Ibc/II SNe identified as anomalous are first marked as such after peak magnitude. Type Ia and Type Ibc SNe are typically flagged just around peak, while Type II SNe are flagged, on average, about a week postpeak. In contrast, ≈65% of anomalies from minority classes are identified as anomalous before peak. They are flagged, on average, about 1 week before peak. However, short-lived anomalous transients (such as ILOTs and CARTs) are flagged around or postpeak. We visualize our findings for a representative sample of transients in Figure 8. The background histogram of the vector plot in Figure 8 shows the overall density of the anomaly scores over time. Most curves have just a few points around $t = 0$ and low anomaly scores. The arrows show that the anomaly scores typically rise before peak but plateau after peak. This implies that, on average, the scores are steady postpeak.

Finally, we investigate whether light curves often begin as anomalous and then drop to less anomalous over time. For this test, we use a 99th percentile cutoff in the anomaly score. We show a selection of representative anomaly curves over time in Figure 8; in this figure, gray curves are SNe that never reach the anomaly threshold, blue curves are members of the minority classes that reach the anomaly threshold, and orange curves are members of the majority classes that reach the anomaly threshold. Light blue/orange curves represent members of the minority/majority classes that reach the anomaly threshold but drop below threshold before the end of the event. In contrast, dark curves are those that remain anomalous until the end of the event. We find that about 7% of events reach this threshold at least once (with, by definition, 1% remaining anomalous by the final observation). As previously mentioned, we find that members of the minority classes are typically flagged as anomalous before peak, while members of the majority classes are flagged after peak; this is regardless of whether or not those events drop below the anomaly

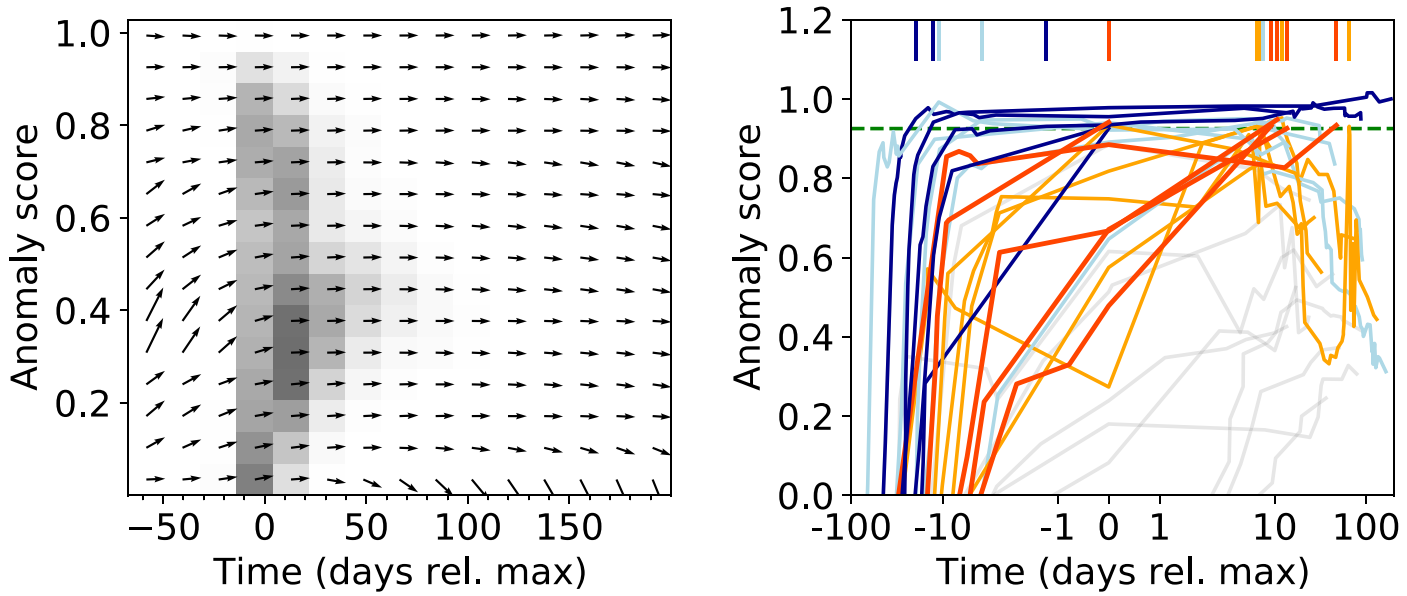

Figure 8. Evolution of anomaly scores as a function of time for a representative set of SNe. Left: vector plot showing the flow of anomaly scores over time. Arrows represent the average gradients of the anomaly scores. Right: anomaly score curves for a representative set of SNe. The green dashed line is the 99th percentile threshold for the final values (having an anomaly score of $A \approx 0.95$). Gray curves are SNe that never cross the anomaly threshold. Blue and orange curves represent events from the minority and majority classes, respectively. The light orange/blue curves drop below the anomaly threshold before the final observation, while darker curves remain above the anomaly threshold until the end. The colored vertical dashes above the anomaly curves represent the times at which the anomalous events first cross the anomaly threshold. Members of the minority class are much more likely to cross this trigger threshold before maximum light, while erroneous anomalies from the majority classes are triggered after peak luminosity.

threshold by the final observation. The fact that the minority classes are flagged before peak is useful in practice, as we are much more likely to follow events caught before peak luminosity.

5. Conclusions

We presented an anomaly detection pipeline for SN-like transients in an LSST-like filtered data stream. We repurpose the PLAsTiCC data set to train and test our algorithm, allowing us to analyze how, why, and when events are tagged as anomalous. Our key results are as follows.

1. We present a novel VRAENN architecture that encodes SN-like light curves in real time into a low-dimension vector. We train this neural network on 1% of 1,129,184 events from the PLAsTiCC data set.
2. We pair this neural network with an isolation forest to assign every transient an anomaly score. We use a Monte Carlo method to estimate our uncertainty on this score.
3. We examine the efficacy of our algorithm through a series of percentile and uncertainty cuts. We find that our algorithm is successful in identifying anomalous classes, especially luminous events.
4. We find that our algorithm is often limited by the photometric redshift estimate. Catastrophically incorrect redshift estimates of Type Ia SNe are especially challenging to remove from our anomaly samples.
5. We find that members of minority classes (i.e., SNe that are not Type Ia, Type Ibc, or Type II) are likely to be identified before peak luminosity. In contrast, erroneously flagged members of the majority classes are more likely to be flagged postpeak.

Much is left to be done to sufficiently prepare for the deluge of data that will come in the new era of the Rubin Observatory. An algorithm like the one presented here must be integrated into Rubin Observatory Alert Brokers, such as ANTARES (Matheson et al. 2021) and Alerce (Förster et al. 2021). These brokers will ingest the life LSST alert packets, run user-defined filters such as ours, and provide the community with a tagged and curated data stream. Importantly, this work must be tested on real data with an active follow-up campaign to validate the proposed method and better understand real false positives. Furthermore, it is possible to use the anomaly score designed here in classification methods to increase the purity of rare transient samples at the cost of completeness.

We thank F. Bianco, L. Garrison, and K. Malanchev for insightful conversations and comments on this work. We additionally thank an anonymous referee for constructive feedback that improved the quality of this manuscript. V.A.V. is supported by the Simons Foundation through a Simons Junior Fellowship (No. 718240). The Berger Time Domain group at Harvard is supported in part by NSF and NASA grants, as well as the NSF under Cooperative Agreement PHY-2019786 (the NSF AI Institute for Artificial Intelligence and Fundamental Interactions, http://iaifi.org/). This work made use of the Habanero cluster at Columbia University. This research made use of the following software packages: `numpy` (Harris et al. 2020), `scipy` (Virtanen et al. 2020), `jupyter` (Kluyver et al. 2016), `sklearn` (Pedregosa et al. 2011), `matplotlib` (Hunter 2007), `tensorflow` (Abadi et al. 2016), and `astropy` (Astropy Collaboration et al. 2018).

Appendix

Here we investigate the AGN included in the PLAsTiCC simulation. In Figure 9, we show the same anomaly samples shown in Figure 7 but including AGN. The AGN dominate the anomaly sample in all cuts. Our algorithm overrepresents AGN in our anomaly sample at the same rate as SLSNe and PISNe.

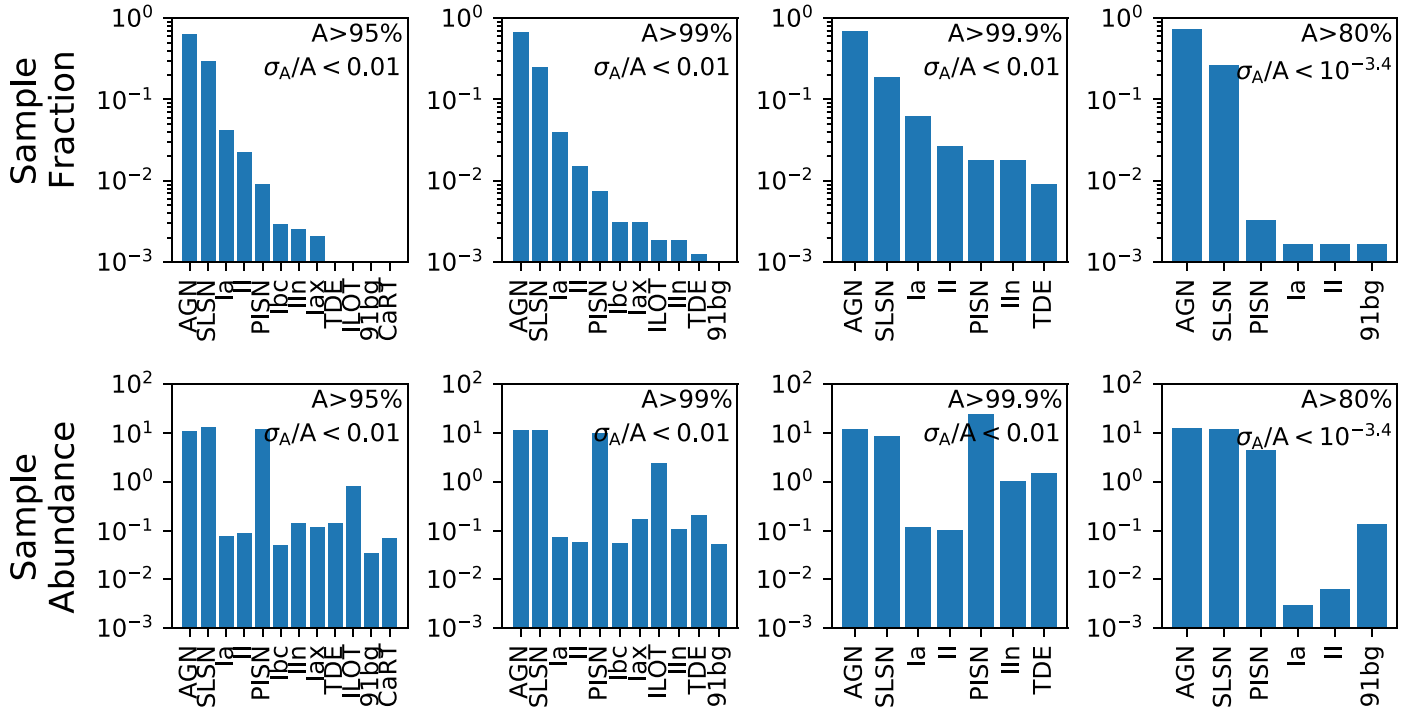

Figure 9. Top row: breakdown of anomalous events, given the cuts listed in each panel. Cuts are identical to those of Figure 7 but now include AGN. The AGN dominate the samples in all cuts. Bottom row: relative abundance of anomaly samples shown in the top row. The AGN are, like members of the minority class, highly overrepresented in the anomaly sample.

However, this is likely not representative of reality. The PLAsTiCC AGN are generated via a damped random walk with a structure function to define the correlation between each filter, as described in MacLeod et al. (2011). We find that, among the AGN identified as "anomalous" in this study, all AGN have at least 20 data points spanning at least 60 days in duration. These events would very likely be identified as AGN via other classification methods and removed from our anomaly data stream. These events are an example of a simple anomaly detection, in which simple filters would likely also pick them out as distinct from SNe (see a summary of similar pitfalls in Wu & Keogh 2020).

Finally, we note that being able to identify AGN in the LSST data stream is an open problem (Shemmer et al. 2018). At first glance, our algorithm is seemingly very successful at identifying AGN; however, these are likely AGN that will be quickly identified by other means. Future studies will need to identify extreme outbursts of AGN (not explicitly simulated by PLAsTiCC) and AGN with substantial dust extinction (likely not identified in the LSST alert stream unless they undergo a significant outburst). Searching for these elusive AGN continues to be an open problem that requires further development of specialized classification methods.

ORCID iDs

V. Ashley Villar https://orcid.org/0000-0002-5814-4061
Miles Cranmer https://orcid.org/0000-0002-6458-3423
Edo Berger https://orcid.org/0000-0002-9392-9681
Gabriella Contardo https://orcid.org/0000-0002-3011-4784
Griffin Hosseinzadeh https://orcid.org/0000-0002-0832-2974
Joshua Yao-Yu Lin https://orcid.org/0000-0003-0680-4838

References

Abadi, M., Barham, P., Chen, J., et al. 2016, arXiv:1605.08695
Aleo, P. D., Ishida, E. E., Kornilov, M., et al. 2020, RNAAS, 4, 112
Allam, T., Jr, Bahmanyar, A., Biswas, R., et al. 2018, arXiv:1810.00001
Anderson, J. P., González-Gaitán, S., Hamuy, M., et al. 2014, ApJ, 786, 67
Arnett, W. D. 1982, ApJ, 253, 785
Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., et al. 2018, AJ, 156, 123
Bellm, E. C., Kulkarni, S. R., Graham, M. J., et al. 2018, PASP, 131, 018002
Betoule, M. E. A., Kessler, R., Guy, J., et al. 2014, A&A, 568, A22
Boone, K. 2019, AJ, 158, 257
Bricman, K., & Gomboc, A. 2020, ApJ, 890, 73
Chalapathy, R., & Chawla, S. 2019, arXiv:1901.03407
Chambers, K. C., Magnier, E. A., Metcalfe, N., et al. 2016, arXiv:1612.05560
Cho, K., van Merrienboer, B., Gulcehre, C., et al. 2014, arXiv:1406.1078
Chollet, F. 2015, Keras, https://github.com/fchollet/keras
Filippenko, A. V., Richmond, M. W., Branch, D., et al. 1992, AJ, 104, 1543
Fletcher, R. 1986, Practical Methods of Optimization (New York: Wiley)
Förster, F., Cabrera-Vives, G., Castillo-Navarrete, E., et al. 2021, AJ, 161, 242
Galbany, L., Hamuy, M., Phillips, M. M., et al. 2016, AJ, 151, 33
Gómez, C., Neira, M., Hernández Hoyos, M., Arbeláez, P., & Forero-Romero, J. E. 2020, MNRAS, 499, 3130
Graham, M. L., Connolly, A. J., Ivezić, Ž, et al. 2018, AJ, 155, 1
Graham, M. J., Djorgovski, S., Drake, A. J., et al. 2017, MNRAS, 470, 4112
Graur, O., Bianco, F. B., Modjaz, M., et al. 2017, ApJ, 837, 121
Guillochon, J., Nicholl, M., Villar, V. A., et al. 2018, ApJS, 236, 6
Guillochon, J., Parrent, J., Kelley, L. Z., & Margutti, R. 2017, ApJ, 835, 64
Guy, J., Astier, P., Baumont, S., et al. 2007, A&A, 466, 11
Harris, C. R., Millman, K. J., van der Walt, S. J., et al. 2020, Natur, 585, 357
Hosseinzadeh, G., Dauphin, F., Villar, V. A., et al. 2020, ApJ, 905, 93
Hunter, J. D. 2007, CSE, 9, 90
Ishida, E. E., Kornilov, M. V., Malanchev, K. L., et al. 2021, A&A, 650, A195
Ivezić, Ž, Kahn, S. M., Tyson, J. A., et al. 2019, ApJ, 873, 111
Jedicke, R., Tonry, J., Veres, P., et al. 2012, AAS/DPS Meeting, 44, 210.12
Jiang, B., Jiang, S., & Villar, V. A. 2020, RNAAS, 4, 16
Kasen, D., Metzger, B., Barnes, J., Quataert, E., & Ramirez-Ruiz, E. 2017, Natur, 551, 80
Kasen, D., Woosley, S., & Heger, A. 2011, ApJ, 734, 102
Kasliwal, M. M. 2012, PASA, 29, 482
Kasliwal, M. M., Kulkarni, S., Gal-Yam, A., et al. 2012, ApJ, 755, 161

Kessler, R., Conley, A., Jha, S., & Kuhlmann, S. 2010, arXiv:1001.5210
Kessler, R., Narayan, G., Avelino, A., et al. 2019, PASP, 131, 094501
Kingma, D. P., & Ba, J. 2014, arXiv:1412.6980
Kingma, D. P., & Welling, M. 2013, arXiv:1312.6114
Kluyver, T., Ragan-Kelley, B., Pérez, F., et al. 2016, in Positioning and Power in Academic Publishing: Players, Agents and Agendas, ed. F. Loizides & B. Schmidt (Amsterdam: IOS Press), 87
Li, D., Chen, D., Shi, L., et al. 2019, arXiv:1901.04997
Li, W., Filippenko, A. V., Chornock, R., et al. 2003, PASP, 115, 453
Li, W., Leaman, J., Chornock, R., et al. 2011, MNRAS, 412, 1441
Liu, F. T., Ting, K. M., & Zhou, Z.-H. 2012, ACM Trans. Knowl. Discov. Data, 6, 3
MacLeod, C., Brooks, K., Ivezić, Ž, et al. 2011, ApJ, 728, 26
Malanchev, K., Pruzhinskaya, M., Korolev, V., et al. 2021, MNRAS, 502, 5147
Martínez-Galarza, J. R., Bianco, F., Crake, D., et al. 2020, arXiv:2009.06760
Matheson, T., Stubens, C., Wolf, N., et al. 2021, AJ, 161, 107
Mockler, B., Guillochon, J., & Ramirez-Ruiz, E. 2019, ApJ, 872, 151
Morawski, F., Bejger, M., Cuoco, E., & Petre, L. 2021, Mach. Learn. Sci. Technol., 2, 045014
Muthukrishna, D., Narayan, G., Mandel, K. S., Biswas, R., & Hložek, R. 2019, PASP, 131, 118002
Nicholl, M., Guillochon, J., & Berger, E. 2017, ApJ, 850, 55
Nugent, P., Kim, A., & Perlmutter, S. 2002, PASP, 114, 803
Pasquet, J., Pasquet, J., Chaumont, M., & Fouchez, D. 2019, A&A, 627, A21
Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, J. Mach. Learn. Res., 12, 2825
Perley, D. A., Fremling, C., Sollerman, J., et al. 2020, ApJ, 904, 35
Pruzhinskaya, M. V., Malanchev, K. L., Kornilov, M. V., et al. 2019, MNRAS, 489, 3591
Rees, M. J. 1988, Natur, 333, 523
Sako, M., Bassett, B., Becker, A. C., et al. 2018, PASP, 130, 064002
Sánchez-Sáez, P., Reyes, I., Valenzuela, C., et al. 2021, AJ, 161, 141
Shappee, B., Prieto, J., Stanek, K., et al. 2014, AAS Meeting, 223, 236.03
Shemmer, O., Richards, G., Brandt, N., et al. 2018, LSST AGN Science Collaboration Roadmap, https://agn.science.lsst.org/sites/default/files/LSST_AGN_SC_Roadmap_v1p0.pdf
Soraisam, M. D., Saha, A., Matheson, T., et al. 2020, ApJ, 892, 112
Villar, V. A., Berger, E., Metzger, B. D., & Guillochon, J. 2017, ApJ, 849, 70
Villar, V. A., Berger, E., Miller, G., et al. 2019, ApJ, 884, 83
Villar, V. A., Hosseinzadeh, G., Berger, E., et al. 2020, ApJ, 905, 94
Villar, V. A., Nicholl, M., & Berger, E. 2018, ApJ, 869, 166
Virtanen, P., Gommers, R., Oliphant, T. E., et al. 2020, NatMe, 17, 261
Wu, R., & Keogh, E. J. 2020, arXiv:2009.13807
Zhang, C., Song, D., Chen, Y., et al. 2018, arXiv:1811.08055
Zhao, H., Wang, Y., Duan, J., et al. 2020, in 2020 IEEE Int. Conf. on Data Mining (ICDM) (Piscataway, NJ: IEEE), 841