The Astrophysical Journal Supplement Series, 236:9 (26pp), 2018 May https://doi.org/10.3847/1538-4365/aab781

Machine-learning-based Brokers for Real-time Classification of the LSST Alert Stream

Gautham Narayan[1,13], Tayeb Zaidi[2], Monika D. Soraisam[3], Zhe Wang[4], Michelle Lochner[5,6,7], Thomas Matheson[3], Abhijit Saha[3], Shuo Yang[4], Zhenge Zhao[4], John Kececioglu[4], Carlos Scheidegger[4], Richard T. Snodgrass[4], Tim Axelrod[8], Tim Jenness[9,10], Robert S. Maier[11], Stephen T. Ridgway[3], Robert L. Seaman[12], Eric Michael Evans[4], Navdeep Singh[4], Clark Taylor[4], Jackson Toeniskoetter[4], Eric Welch[4], and Songzhe Zhu[4]
(The ANTARES Collaboration)

[1] Space Telescope Science Institute, 3700 San Martin Dr., Baltimore, MD 21218, USA; gnarayan@stsci.edu
[2] Department of Physics and Astronomy, Macalester College, 1600 Grand Ave., Saint Paul, MN 55105, USA
[3] National Optical Astronomy Observatory, 950 N. Cherry Ave., Tucson, AZ 85719, USA
[4] Department of Computer Science, University of Arizona, 1040 E. 4th St., Tucson, AZ 85721, USA
[5] African Institute for Mathematical Sciences, 6 Melrose Rd., Muizenberg 7945, Cape Town, South Africa
[6] SKA South Africa, 3rd Floor, The Park, Park Rd., Pinelands, 7405, South Africa
[7] Department of Physics and Astronomy, University College London, Gower St., London WC1E 6BT, UK
[8] Steward Observatory, University of Arizona, 933 N. Cherry Ave., Tucson, AZ 85720, USA
[9] Department of Astronomy, Cornell University, Ithaca, NY 14853, USA
[10] LSST Project Office, 950 N. Cherry Ave., Tucson, AZ 85719, USA
[11] Department of Mathematics, University of Arizona, 1040 E. 4th St., Tucson, AZ 85721, USA
[12] Lunar & Planetary Laboratory, University of Arizona, 1629 E University Blvd., Tucson, AZ 85721, USA

Received 2017 October 19; revised 2018 February 21; accepted 2018 February 22; published 2018 May 11

Abstract

The unprecedented volume and rate of transient events that will be discovered by the Large Synoptic Survey Telescope (LSST) demand that the astronomical community update its follow-up paradigm. Alert-brokers—automated software system to sift through, characterize, annotate, and prioritize events for follow-up—will be critical tools for managing alert streams in the LSST era. The Arizona-NOAO Temporal Analysis and Response to Events System (ANTARES) is one such broker. In this work, we develop a machine learning pipeline to characterize and classify variable and transient sources only using the available multiband optical photometry. We describe three illustrative stages of the pipeline, serving the three goals of early, intermediate, and retrospective classification of alerts. The first takes the form of variable versus transient categorization, the second a multiclass typing of the combined variable and transient data set, and the third a purity-driven subtyping of a transient class. Although several similar algorithms have proven themselves in simulations, we validate their performance on real observations for the first time. We quantitatively evaluate our pipeline on sparse, unevenly sampled, heteroskedastic data from various existing observational campaigns, and demonstrate very competitive classification performance. We describe our progress toward adapting the pipeline developed in this work into a real-time broker working on live alert streams from time-domain surveys.

Key words: methods: data analysis – methods: statistical – stars: variables: general – supernovae: general – surveys – virtual observatory tools

1. Introduction

The Large Synoptic Survey Telescope (LSST; Ivezić et al. 2008) will revolutionize astrophysics, probing deeper than the previous generation of wide-field surveys and replacing static maps with a continuous movie of the night sky, producing ∼20 terabytes of raw images every single night. This is approximately the same data volume as all of the imaging data obtained by the Sloan Digital Sky Survey (SDSS; Abolfathi et al. 2017) over a decade. However, despite the dramatic increase in depth and data volume, ongoing surveys, including the Dark Energy Survey (DES; Dark Energy Survey Collaboration et al. 2016) and the newly commissioned Zwicky Transient Facility (ZTF; Law et al. 2009; Rau et al. 2009; Ofek et al. 2012; Smith et al. 2014, and references therein), still visually inspect candidate detections of source variability, commonly referred to as "alerts," to determine the most promising targets for follow-up studies.

Visual inspection does have merits: humans are very capable in distinguishing pathological data from interesting astrophysical behavior, can make inferences despite sparse or missing information, and can combine and derive complex contextual information, which is incorporated into their final classification decision. But as the volume of alerts grows, the efficacy of visual inspection by humans decreases, and the process of classification by visual inspection becomes increasingly inconsistent and rate-limiting. Consequently, rare and extremely scientifically interesting objects often go unstudied because detailed follow-up could not be prioritized in time, or simply because they were not identified as unusual from sparse early phase observations.

The limitations of human inspection have been recognized for some time, but the effort to replace eyeballs with algorithms at different stages of the analysis is not a simple task. As reported by various transient surveys, candidate transient sources flagged by the difference imaging pipelines include "bogus" artifacts, overwhelming the number of bona fide objects detected in difference images by an order of magnitude

[13] Lasker Fellow.

or more. Increasingly complex automated filtering is being applied to winnow down the alert streams and separate real astrophysical sources from artifacts, e.g., with SDSS: du Buisson et al. (2015); Pan-STARRS: Wright et al. (2015); DES: Goldstein et al. (2015); the Intermediate Palomar Transient Factory (iPTF): Brink et al. (2013) and Masci et al. (2017); Hyper Suprime-Cam Survey: Morii et al. (2016); notably, the Optical Gravitational Lensing Experiment (OGLE): Klencki et al. (2016) using an unsupervised hierarchical self-organizing map (SOM); and the High cadence Transient Search (HiTS): Cabrera-Vives et al. (2017) using deep learning with a rotationally invariant convolutional neural network (Deep-HiTS).

Improvements in real–bogus categorization are necessary, but will not address the classification of alerts that enables an investigator to select follow-up targets matching particular criteria. The key distinction between real–bogus categorization and alert classification is that the former functions on features extracted from individual difference images—a snapshot—whereas the latter considers the time evolution of the source—a sequence—along with contextual information. Alerts from different images, potentially from different surveys, must be cross-matched and combined, tagged with contextual information and the results of any spectroscopic follow-up, and this combined alert packet for each astrophysical source must be characterized, and if possible, definitively classified. Human screening of an alert package can take several seconds. LSST is expected to produce $\sim 10^7$ alerts per night, a rate that far exceeds the capacity of visual inspection. LSST will require a software system capable of (1) automated real-time classification of alerts and (2) filtering and distributing alerts to allow astronomers to focus on objects that are relevant to their scientific interests—an “alert-broker.”

The Arizona-NOAO Temporal Analysis and Response to Events System (ANTARES) is an alert-brokering system that we are developing to meet these requirements. ANTARES is designed to sift through the alert stream and characterize events, with the goal of identifying phenomena that are exceedingly infrequent, as well as those of interest to the broader astronomical community (Saha et al. 2016, 2014). This feature distinguishes ANTARES from existing broker services, which use human inspection to serve early-phase transient alerts to the community. While both automated classification and alert distribution systems exist, they have seldom been combined, and even the few automated alert-brokers that have been developed have never been operated at LSST scale.

1.1. An Overview of this Paper

In this work, we will present progress toward developing ANTARES, describing some of the challenges posed by alert-broker development, as well as highlighting how a system may be useful to different studies. We illustrate this by posing three problems, focused on solving three different scientific questions:

1. Real-time filtering on extremely sparse data to enable recognition, categorization, and rapid follow-up of unusual phenomena,
2. General alert characterization to provide different science interests with feeds of relevant objects,
3. Stringent retrospective classification to identify members of a specific class of objects.

These three use cases drive the choices we make for feature extraction and classifier development in this work.

Following a broad review of automated classification and alert-brokering in Section 2, we describe the design and current status of the ANTARES classification pipeline in Section 3. We structure the subsequent sections of this work to reflect the development cycle for our pipeline, with each section discussing a different component of the broker system. In Section 4, we describe the data sets, real and simulated, that are used to develop and validate the classification pipeline. In Section 5, we describe feature extraction from the time-series data, prior to the development of the multiple stages we use to process the light curves in Section 6. In Section 7, we describe the training and application of our suite of classification algorithms, and we introduce metrics to evaluate their performance. We summarize our conclusions in Section 8 and discuss future avenues of development for ANTARES in Section 9.

2. An Overview of Automated Classification and Alert-brokering

We provide a brief overview of the existing literature on automated variable and transient classification systems and alert-brokering systems in the following subsections.

2.1. Variable Classification

The automated classification of variable stars has a long history, beginning with the development of methods to determine the periods of pulsating and eclipsing variables, including the Lafler–Kinman statistic (Lafler & Kinman 1965), the Lomb–Scargle periodogram (Lomb 1976; Scargle 1981), and several Fourier power spectrum methods—Analysis of Variance (AOV; Schwarzenberg-Czerny 1996), Phase Dispersion Minimization (PDM; Stellingwerf 1978), Bayesian Evidence Estimation (Gregory & Loredo 1992), Conditional Entropy (Graham et al. 2013b), as well as hybrid methods (Saha & Vivas 2017). Despite the differences in these techniques, Graham et al. (2013a) found that most methods exhibited comparable performance on realistic data, and there was no single optimum algorithm. The most accurate algorithm for period determination depended on the astrophysical class being studied—information that is not available for most sources. Indeed, variable classification is one of the purposes for which period determination is used in the first place.

While the period is one of the most important features in discriminating between different classes of variables, there are many features that are sensitive to light-curve shape. Early work by Eyer & Blake (2005) showed that even small sharp features in the light curves could improve the accuracy of period determination. Together with clustering techniques and naive Bayes classifiers, these light-curve shape features could be used to label large data sets much faster than would have been possible by visual inspection. Debosscher et al. (2007) successfully applied machine learning techniques to a very diverse set of variables, with over 20 different classes, drawn from several different surveys, showing that the classification features and the technique were extremely robust and could be used to label new data sets (Debosscher et al. 2009; Sarro et al. 2009).

Richards et al. (2011, hereafter R11) vastly expanded the set of features that are sensitive to the shape of the light curve to include many metrics that are more robust in the presence of noisy or spurious data. R11 found that including these robust features and a hierarchical taxonomy of labels could dramatically improve classification performance on the Debosscher et al. (2007) data set. Richards et al. (2012) utilized these features, together with iterated active learning—prioritizing follow-up of objects whose inclusion into the training sample maximally helps classification—to reduce sample selection biases. R11 remains the conceptual basis for many contemporary methods applied to large data sets, such as Masci et al. (2014), as well as many software packages for variable star classification, such as Kim et al. (2014)[14] and FATS[15] and its derivative, feets.[16] Unfortunately, as many of the observational programs that discover variable stars only observe with a single filter, many of the features that are employed by these software packages are designed with the assumption of single-band photometry. These packages make limited use of multicolor information, despite its utility in discriminating between different classes of variables.

2.2. Transient Classification

While variable classification is retrospective—it can be applied well after the observations have been taken—one of the main goals of transient classification is to operate in real time to select objects for spectroscopic follow-up while they are still active. Early "flash" spectroscopy is particularly important for understanding the physics of the progenitor systems (e.g., Khazov et al. 2016). Even programs that derive cosmological constraints from photometric samples of Type Ia supernovae (SNe Ia) use spectroscopy to assess their contamination (Jones et al. 2017), and have a relatively narrow window to obtain spectroscopy with sufficient S/N.

This need for real-time rapid response drove some of the earliest advances in using machine-learning techniques for transient classification. Poznanski et al. (2007) attempted to distinguish between SNe Ia and core-collapse SNe using single-epoch photometry along with a photometric redshift estimate from the probable host galaxy. Many contemporary techniques, such as by Foley & Mandel (2013) and the sherlock package,[17] can operate on sparse, or only contextual information, allowing for spectroscopic follow-up while the transient rises to maximum light.

The Supernova Photometric Classification Challenge (hereafter SNPhotCC) in 2009 was one of the earliest efforts to employ machine-learning algorithms for light-curve classification along the lines of the work done on variable stars. The challenge simulated SN light curves with the properties of the DES, and aimed to determine which techniques could distinguish SNe Ia from several other classes of SNe. The data for the SNPhotCC was provided in full along with the original types chosen for generation by Kessler et al. (2010b).

The techniques used to classify for the challenge varied widely, from basic spline fitting and semi-supervised learning, to much more complex methodologies that fit light curves with a variety of templates (such as Guy et al. 2010 or parametric descriptions such as Karpenka et al. 2013), and compared classification results using an ensemble of classification schemes (Kessler et al. 2010a). A measure of the value of this exercise is that the SNPhotCC data set is still used as the reference standard to benchmark contemporary SN light-curve classification schemes, such as Lochner et al. (2016, hereafter L16). In this paper, we integrate the nonparametric transient classification framework developed in L16 as a component of the ANTARES pipeline. An updated version of SNPhotCC, now including more classes than just different types of SNe, and with simulations appropriate for LSST, the "Photometric LSST Astronomical Time-series Classification Challenge" (PLAsTiCC), is in development.[18]

2.3. Alert Filtering and Distribution Systems—Brokers

An alert-broker is a software system to rapidly characterize and filter alerts. Brokers must be capable of producing large and pure samples of known astrophysical classes, and must therefore be effective at distinguishing between classes, as well as being able to identify rare and novel sources within each class. As they operate in real time, alert-brokers must cope with complex streaming input from different astronomical facilities studying different parts of the electromagnetic spectrum. This can include objects with pathological and/or missing data and that can be contradictory—e.g., spectra of the same object from two groups with two different classifications. The data for each object will span a different range in phase, so unlike classification algorithms, assembling a consistent input feature vector across all events is simply not possible. This means brokering systems must necessarily adopt some sort of hierarchical classification scheme, depending on the amount of data available for a given object. This reflects the R11 finding that adopting a hierarchical taxonomy of variable stars improves classification.

There are a few notable examples of alert-brokers that have classified variables and transients in real time on survey data. The Oarical system (Bloom et al. 2012) employed during the PTF survey was one of the first automated transient brokers. However, alerts were not made public, but rather supplied to PTF members through "Marshals"—an alert-brokering systems tailored to the needs of specific projects within the PTF collaboration. These alerts were still visually inspected before follow-up. The Oarical code has also never been made public, and while elements of it have been adapted to iPTF alerts and will likely be applied to ZTF alerts, Marshals still employs visual classification to select targets for spectroscopic follow-up. The Catalina Real-time Transient Survey (CRTS) operated an automated classifier (Djorgovski et al. 2014) for a period in 2015; however, current optical transients are all reported as human classified. 4PiSky (Staley & Fender 2016)[19] and the (now-defunct) Williams et al. (2009) serve alerts in the VOEvent format[20] (Seaman et al. 2011) defined by the International Virtual Observatory Alliance.

Brokers do not require machine-learning classification schemes at all, and indeed many brokering systems simply annotate the alert stream. One of the most successful, though underappreciated, brokering systems is the Rochester Supernova Web page[21] maintained solely by amateur astronomer David Bishop. The Rochester page has provided alerts on

[14] https://github.com/dwkim78/upsilon
[15] https://github.com/isadoranun/FATS
[16] https://github.com/carpyncho/feets
[17] https://github.com/thespacedoctor/sherlock
[18] https://plasticcblog.wordpress.com/
[19] https://4pisky.org/
[20] http://www.ivoa.net/documents/VOEvent/index.html
[21] http://www.rochesterastronomy.org/supernova.html

potential SNe to a vast number of follow-up teams for more than two decades, and is often updated with the results of follow-up observations before the International Astronomical Union's (IAU) Transient Name Server (TNS),[22] itself another example of a transient alert-broker. However, the critical issue for large survey projects like LSST in the coming years is having a way to cope with the quantity and the rate of data. Brokers must be able to utilize the capacity they have to keep pace with the data rate from LSST, which will produce a new image every 37 s. To contend with the high event rate and volume, we will have no alternative but to adapt to using machine-learning algorithms to process the alert stream.

3. ANTARES

ANTARES consists of two components: (1) a pipeline that will use automated classification techniques to provide real-time characterization and annotation of alerts, and (2) an alert distribution system that will allow these annotated alerts to be searched and filtered by astronomers.

The classification pipeline must be capable of handling data that is incomplete, most often because of the limited observing cadence and lost observing time due to poor weather conditions. It must be able to make a preliminary classification using only the beginning of the light curve, in order to enable rapid spectroscopic follow-up and find interesting new classes of transients, as well as to probe unexplored regimes of the evolution of known classes of objects. Even without high-confidence photometric classification, criteria such as the association of contextual information and categorization can all be used to select objects for early-time studies.

The brokering system wraps around the classification pipeline. It ingests alerts into the system and controls the flow of data through the pipeline. The brokering system also coordinates the bookkeeping of the system: the storage of extracted features and annotations in a database, as well as serving the annotated alert stream to downstream brokers, publishing rare and interesting alerts to Web pages, and allowing users to search and filter the database of all annotated alerts.

Building this system requires expertise from both astronomers and computer scientists. We began active development of the project in 2014 December, adapting existing platforms and services wherever possible, and constructing new tools where none were available. The current architecture of ANTARES is depicted in Figure 1 and described below.

3.1. Architecture of ANTARES

The ANTARES system consists of several components (any shape with a black outline in Figure 1) to process alerts from LSST. Some of these components are part of the pipeline (encompassed by the dashed black outline) run on every alert package from LSST. Components that are interfaces to the community or are external systems are indicated outside this outline, and are not run on every alert package.

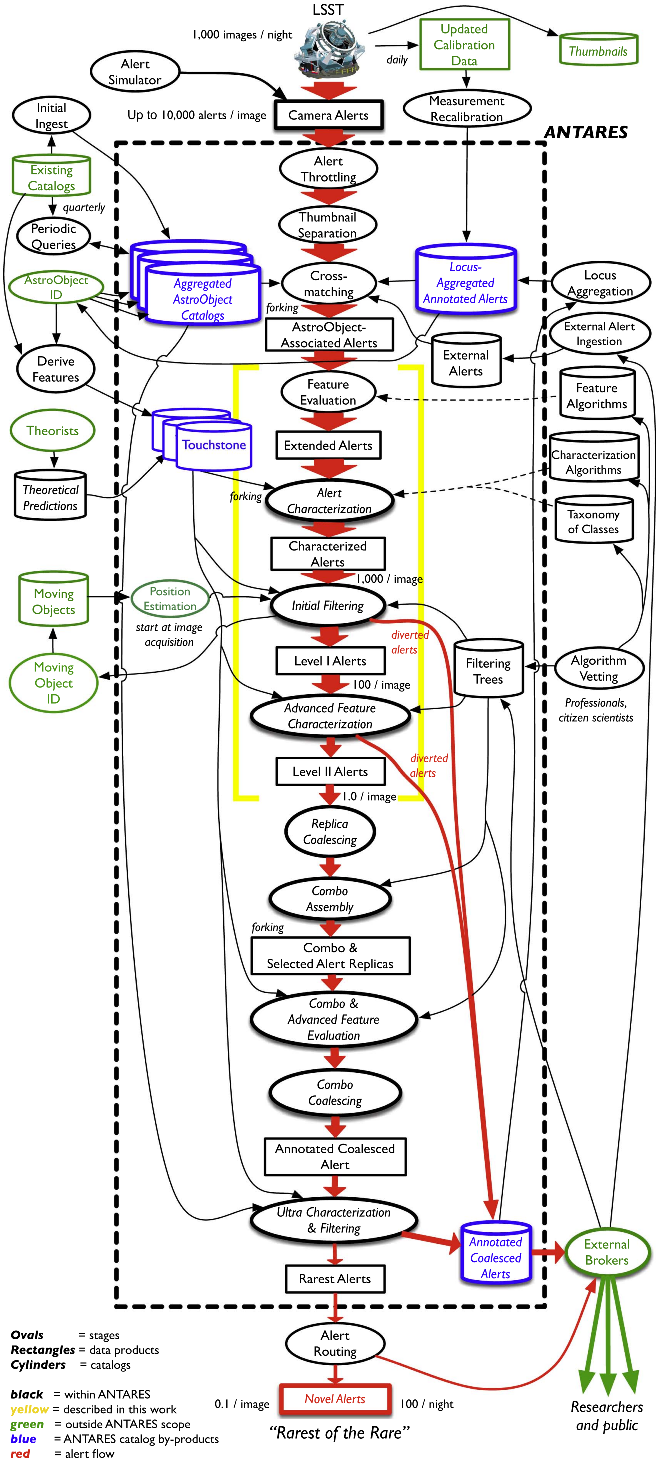

Figure 1. Schematic of the ANTARES architecture. The processing pipeline is enclosed by the black dashed line. The core machine-learning stages described in this work and depicted in Figure 10 are bracketed in yellow.

Prior to ingestion by ANTARES, alert packets from LSST will be filtered through the LSST alert real–bogus discrimination system and the LSST moving object pipeline to remove difference-imaging artifacts and sources that exhibit significant motion against the background sky. The contents of an LSST alert packet are described in the LSST Data Products Definition Document.[23] The filtered difference image alerts produced by the LSST pipeline are ingested into the ANTARES pipeline (the dashed outline) at the top. The alert stream can be throttled by S/N if the volume exceeds our processing capacity. Extraneous information, such as difference-imaging thumbnails are

[22] https://wis-tns.weizmann.ac.il/

[23] http://ls.st/dpdd/

separated from the alert packet. Alerts are cross-matched against static catalogs (referred to as "AstroObjects"), as well as a database of previous history at the same position on the sky (a "locus"), including any prior calculations and evaluation by ANTARES itself. Features are derived from all of the alerts. The contextual information and computed features are stored together with the original data in the LSST alert packet as an "extended alert." Where a feature or contextual attribute cannot be uniquely determined, a duplicate of the alert is created (indicated in Figure 1 by the text label "forking"), and these different copies are assigned the different possible values of the feature or contextual attributes. We term these different copies "replicas."

All replicas of all the alerts are passed on to the various filtering and characterization "stages." Some of these stages score and label the replicas, while other stages trigger actions if the replicas meet certain predefined conditions of interest. These stages (denoted by the yellow bracketed region in Figure 1) are largely designed to compare the alert to a library of astrophysical events with similar properties and known classifications—the "Touchstone." Some stages are more computationally intensive than others, and it would be prohibitive to run these stages on all the alerts from LSST. However, these stages are designed to do more fine-grained classification (e.g., determining the type of SN, rather than discriminate between SNe and variable stars), and only need to be run on a subset of the alerts that match certain criteria after initial filtering. We "divert" alerts that do not meet these criteria, and do not execute these computationally intensive stages on them. We term the alerts that are retained for processing after the initial diversion "Level I Alerts," and the alerts that are retained after the computationally intensive "Advanced Feature Characterization" stages are executed "Level II Alerts." After all the stages are run, all of the replicas of each alert are coalesced, and the ensemble of the classifiers is used to annotate the alert, thereby accounting for different possibilities even when attributes or contextual information cannot be uniquely determined. It is these core classification stages that are the focus of this work.

In some cases, alerts may prove interesting in combination with other external alerts at the same location on the sky. For example, a tidal disruption event (TDE) may be indicated by an optical trigger in a galaxy that has not previously exhibited AGN activity, but has strong ongoing soft X-ray or IR emission. We term such structures "combos" and define much more specialized filtering stages to process them as needed. These stages can be more computationally intensive as the volume of alerts being processed decreases with each filtering stage of the pipeline. The expected numbers of alerts per image are indicated by the text near components in Figure 1, as well as by the decreasing width of the red arrows between stages. Most alerts will not meet the criteria necessary to trigger the creation of a combo and will simply pass through these stages.

The distribution system of ANTARES consists of stages that deliver our processed alerts to the astronomical community (these stages are depicted by the red arrow crossing the dashed black outline). Alerts that are labelled are diverted and recorded along with any computed features into the database, while the subset of those that appear different from all known classes represented in the Touchstone library are identified as novel and prioritized for rapid follow-up. All annotations for all alerts are stored in the database and are accessible by end users (illustrated in Figure 1 by the blue catalog labelled "Annotated Coalesced Alerts").

Various elements of the analysis are not provided directly by ANTARES, and the system instead interfaces with LSST or other facilities that provide these features (indicated in green in Figure 1). These include image calibration and subtraction products, updates to catalogs, lists of moving objects, etc. We plan to develop an API that will allow users to "daisy-chain" instances of ANTARES, refining the output from the general broker to produce results specific to their scientific interests. We will create a mechanism for users to further process and interact with the annotated alerts and features stored in our database via a Project Jupyter Hub, to serve many use cases where real-time access to the alert stream is not required.

3.2. Current Status

Many components of ANTARES have been developed: an alert simulator to inject simulated data into the system; relational databases to store external catalogs, ingested data, and processing results; an API for the execution of the different stages and to interact with the database; a load-balancing system for parallel execution; Web frameworks to inspect the results; systems for configuration management and tracking provenance; as well as the front-end interfaces to serve this data to the community. Development of these components required only a relatively small amount of astrophysical data to serve as test cases. At present, we are moving from our initial goal of identifying the rarest of the rare (where we focus on the lowest red box in Figure 1) to the much bigger challenge of developing a general purpose broker. In this paper, we have elected to focus on the core classification stages, as this remains one of the largest open research questions involved in the development of an alert-broker.

4. Data Sources for Developing and Testing ANTARES

In order to develop and test the machine-learning-based classification stages of the ANTARES broker (the yellow bracketed region in Figure 1) and provide a framework for future classification using LSST's data products, we need much larger data sets that provide a representative range of astrophysical sources. To capture the diversity of the variable and transient sky, we drew from three separate data sources described below.

4.1. OGLE Variable Stars

The Optical Gravitational Lensing Experiment (OGLE; Udalski et al. 1992) is a wide-field sky survey originally designed to search for microlensing events. The project monitors over 200 million stars over several years, assembling an enormous database of photometric measurements. The OGLE project classified their sources using several techniques, evolving from human inspection, through simple categorization and template fitting, to machine-learning-based techniques over their more than two decades of operation. The OGLE-III Catalog of Variable Stars (Soszynski et al. 2008) consists of the observations in V and I collected beginning in 2001. We augment the OGLE-III release with new objects from OGLE-IV, available through their FTP server,[24] using a custom parser to translate the files into a standard format and with a standard

[24] ftp://ftp.astrouw.edu.pl/ogle/ogle4/

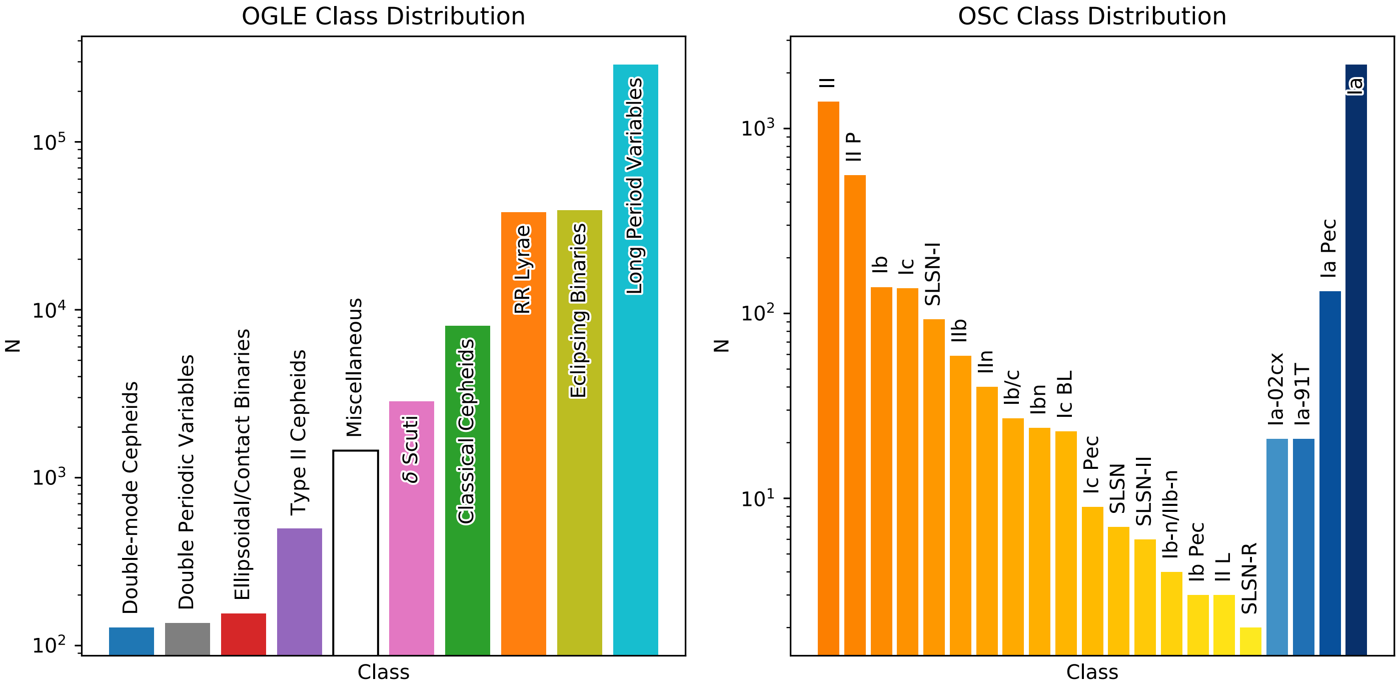

Figure 2. Class distributions for the OGLE (left) and OSC (right) data sets. These distributions clearly highlight the imbalance in the data sets.

set of labels. Adding the new objects allows us a larger sample from classes that are underrepresented in the OGLE-III release.

The classes (with catalog designations listed parenthetically) represented in the OGLE sample include classical Cepheids (“cep”), double periodic variables (“dpv”), δ Cepheids (“dcep”), type II Cepheids (“t2cep”), ellipsoidal/contact binaries (“ell”), eclipsing binaries (“ecl”), δ Scuti variables (“dsct”), RR Lyrae of different types (aggregated as “rrlyr”), and long-period variables or Miras (“lpv”). Miscellaneous other types are aggregated together (“misc”) and are not used in this work, often because there are insufficient observations with $S/N > 5$ to derive features reliably. There is a significant class imbalance in this data set, with over two orders of magnitude more long-period variables than type II Cepheids or double periodic variables. Despite these caveats, OGLE is by far the largest multiclass data set of labelled variable stars with photometry in two passbands (i.e., have at least some color information). The distribution of class labels for the OGLE is shown in the left panel of Figure 2.

4.2. The Open Supernova Catalog

The second data set considered is drawn from the repositories of the Open Supernova Catalog (OSC; Guillochon et al. 2017). The OSC is a public online repository that accepts observations of SNe from all willing contributors. This public repository operates with the goal of containing a complete collection of publicly available data of SNe with both spectra and photometry in visible and near-visible wavelengths, as well as in radio and X-ray. However, the data set is heterogeneous, comprising different classes of SNe, from a variety of teams, selected with different strategies, observed using different sites, telescopes, instruments, passbands, cadences, and image-reduction pipelines, and subject to weather losses, photometric calibration errors, and mislabelling. This heterogeneity is why it has never been used by classifiers until this work. The observational data are stored in the repository as they were originally reported, and include outliers and missing data. Even processing all these varied light curves to produce a consistent feature vector for each object is a nontrivial challenge.

We used all OSC light curves available on the repository as of 2017 January. We subselected objects from 1987 to the present that met a few quality control criteria to exclude objects with an insufficient number of observations. We only used light curves that had at least one claimed type as at least one label is necessary for all supervised machine-learning algorithms. We required at minimum 25 observations across all bands as we found this simple heuristic is enough to ensure sufficient observations in both bands and over a range of phases. For objects where the type is disputed, we take the type claimed by the largest number of unique sources, and if this is insufficient, we assert that the most recently claimed type is correct. The resulting data draw from many different references and a disparate collection of surveys.

In many cases, particularly with the OSC, many of the class labels are subtypes of a parent class, while others are simply ill-defined. For example, “Ia-Pec” in the OSC sample includes underluminous and overluminous SNe Ia while “Ia-02cx” and “Ia-91T” are taken to be underluminous and rapidly declining, and overluminous respectively, and there are no precise criteria for putting an object in one class versus the other. Additionally, in the OSC, many classes are represented by 10 or fewer members—wholly inadequate for training any classifier. We aggregate the many subtypes of SNe into two broad classes—Ia and non-Ia (see Section 6.5). The distribution of class labels for the OSC data set is shown in the right panel of Figure 2.

4.3. SNPhotCC Simulated DES Light Curves

While the real light curves of OSC provide a true assessment of classifier performance, the simulated SNPhotCC data set is the reference standard for SN classifier performance assessment (see Section 2.2). It consists of approximately 18,000 *griz* light curves of Type Ia, Ib/c, and II SNe that were simulated to

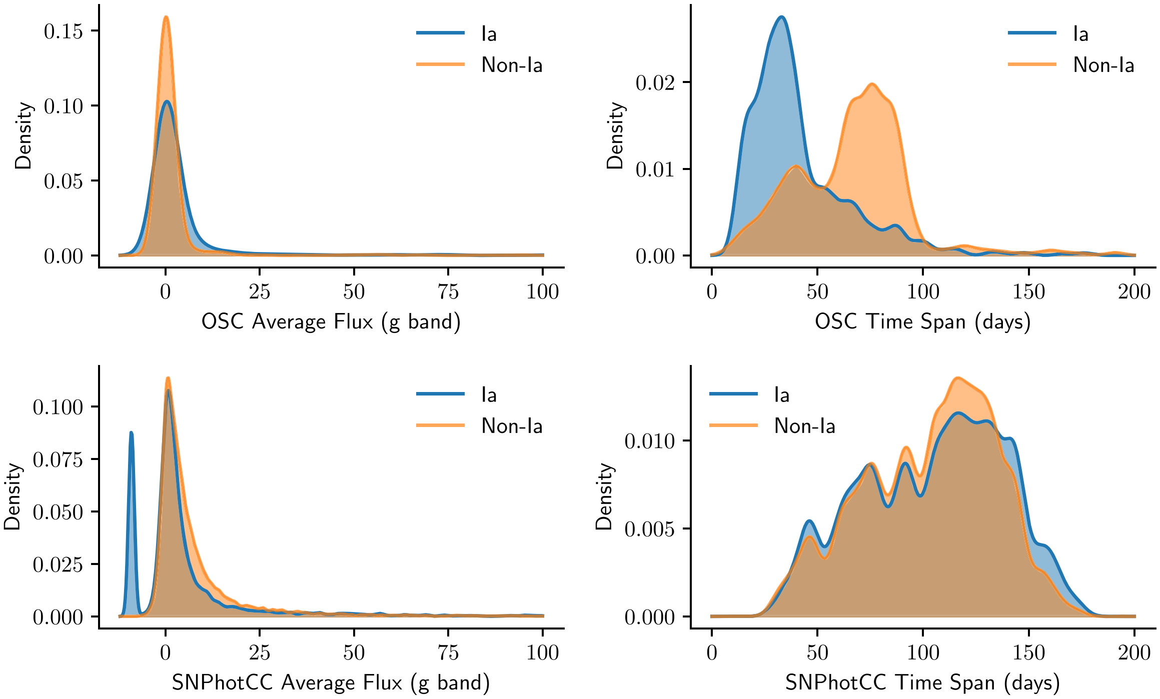

Figure 3. Density plot of the average magnitude (left) and time span (right) of light curves from the OSC (upper row) and SNPhotCC (lower row) data sets.

match the expected properties of DES. Each type is represented in accordance with its expected volumetric rate. For the purposes of the challenge, a “spectroscopically confirmed” subset was provided to give a training set for classification. The training sample is simulated to match the properties of the data set with photometric observations on a 4 meter telescope with a limiting r-band magnitude of 21.5, and spectroscopic follow-up with an 8 meter telescope with a limiting i-band magnitude of 23.5 (Kessler et al. 2010b). This roughly models the performance of the ongoing DES project, based on a library of historical conditions at the Cerro Tololo site assembled during the Equation of State: SupErNovae trace Cosmic Expansion (ESSENCE) survey (Narayan et al. 2016).

The simulated SNe Ia are generated using empirically derived models. Many of the non-Ia SNe were provided by (in increasing order of redshift) the Carnegie Supernova Project (CSP; Folatelli et al. 2010; Stritzinger et al. 2011), the Sloan Digital Sky Survey (SDSS II; Holtzman et al. 2008), and the Supernova Legacy Survey (SNLS; Sullivan et al. 2011; Betoule et al. 2013), and at the time of the challenge, were unpublished. The set of non-Ia light curves undersamples the potential variety that will be seen in future large-scale surveys, with under 50 objects providing the base model for all simulated non-Ia events. However, the SNPhotCC data set still provides a useful step in the interim period before more complete data become available. The Pan-STARRS Medium Deep Survey (MDS) and DES have taken the requisite observations to provide such a sample, and will likely be included in the PLAsTiCC data set.

The SNPhotCC provided two separate challenges, one with host-galaxy redshifts (“+HOSTZ”) and one without (“-NOHOSTZ”). Unsurprisingly, all of the methods in Kessler et al. (2010a) performed better with host-galaxy redshifts. In this work, we have chosen to work on the set *without* host-galaxy redshifts, to reflect the fact that the southern sky that LSST will scan has considerably shallower galaxy catalogs than the northern sky, which has already been imaged by SDSS and PS1.

4.4. Differences between the Data Sets and Preprocessing

The density plots of the average magnitudes and time spans of light curves shown in Figure 3 highlight key differences between these data sources. The SNPhotCC exhibits no strong differences between the Ia and non-Ia SNe for both features. Overall, their time spans are longer than the OSC SNe, as is to be expected from simulating an untargeted survey, following a source for as long as it is visible. For the OSC, though the average magnitudes are similar for Ia and non-Ia light curves, the time spans differ greatly, reflecting the different follow-up strategies used by surveys for these different classes. We explore differences in classifier performance between the OSC and SNPhotCC data set further in Section 7. LSST will produce a much larger and homogeneous collection of SN light curves to replace these disparate data sets. By assessing performance on existing surveys, we can set a lower bound on the classification performance, as well as create a tool that may be used by ongoing precursor surveys to LSST, such as CRTS, the All-Sky Automated Survey for Supernovae (ASAS-SN), and the ZTF.

Most data need relatively simple preprocessing. We remove observations with negative or zero observed flux uncertainty (as these values are unphysical), observations that have missing values for the flux or flux uncertainty (NaNs, as well as dummy values such as −99), and observations represented only by an upper detection limit or lower flux limit, rather than a measurement of the flux.

4.4.1. Passband Mapping

In order to allow for objects from different surveys in the OSC to be comparable to one another, a reference set of filters must be defined. For consistency with the SNPhotCC data, the

Table 1
Surveys Represented in the OSC Light Curves that Require Passband Mapping (see Section 4.4.1)

Survey/Telescope	Largest Compilation
CfA Supernova Group	Hicken et al. (2012)
Carnegie SN Project	Stritzinger et al. (2011)
Lick Observatory SN Search (LOSS)	Ganeshalingam et al. (2010)
NASA Swift Telescope	Brown et al. (2009)
CfA-IR	Friedman et al. (2015)
SDSS SN Search	Holtzman et al. (2008)

Note. The surveys are in decreasing order of precedence for resolution of conflicts when multiple observer-frame passbands from different surveys can be mapped to the same DES passband for any given object. Here, the order roughly reflects the S/N and the cadence of the observations.

griz system used by the DES telescope was adopted as the photometric system. All passbands in the OSC were mapped into the *griz* system using a simple heuristic: if the passband of the original survey overlapped significantly with any of the DES filters, then we assigned the observations to that band. In cases where multiple filters from different low-redshift surveys mapped to the same DES band, we adopted the photometry from the higher ranked survey. The surveys are listed in ranked order in Table 1. We developed an assessment of the quality of the photometry of various low-redshift SN surveys based on their cadence and their typical phase coverage. In order to combine the OSC data with the data from the OGLE survey, which has observations only in V and I, we use just the OSC data mapped into the g and i band for variable–transient categorization and classification. The OGLE variables are diverted after this stage. We use all available *griz* information in the final stage of our pipeline, where we compare SNe against each other to determine the feasibility of selecting a pure sample of SNe Ia.

This is a very simplistic system for passband mapping, and it throws away a large number of high-quality light curves in the process when there is no appropriate overlapping passband, as well as losing information from each object. There are more sophisticated techniques available for passband mapping such as Scolnic et al. (2015), but these require information about the redshift and/or type of the object. The former is not typically available, and the latter is precisely the quantity that we wish to infer. While the SNPhotCC provides reference data with a consistent photometric system, more than 80% of the light curves (both type Ia and non-type Ia) are synthetically generated using the same empirically derived models that are often used later in classification. Thus, it remains necessary to use data sets such as the OSC, even though the disparate nature of the sources makes comparative analysis more challenging. We intend this exercise to be illustrative of broker development, as well as a step toward semi-supervised learning on hitherto unclassified PS1 and iPTF light curves, and is therefore necessary, despite these compromises.

5. Feature Extraction Methodology

Supervised learning algorithms are trained on a library of features from objects with known class labels to derive a desired function. For classification purposes, this function is the class label itself. The derived function can then be applied to the same features derived from data with unknown class labels to predict the value of the function.

Supervised learning algorithms therefore require three successive operations:

1. Projecting the high-dimensional information of each source to a lower dimensional feature space that encapsulates as much information as possible: *encoding*
2. Learning a metric that can be used to quantify the differences between classes in that space using many labelled instances of each class: *training*
3. Applying the metric to new unlabelled instances in order to filter, characterize, or classify them: *prediction*

In this section, we detail the first of these operations—the encoding, or feature extraction. The choices of the features that we attempt to extract are driven by the information available for each source. For this work, we consider feature extraction in three distinct regimes, corresponding to the three different questions we posed in Section 1.1:

1. Soon after the alert is issued, where only limited information is available—perhaps only a couple of observations. The only features derived at this stage are an amplitude, rate of change, or a single color. Rapid real-time prioritization will depend on effective filtering in this regime and will rely on additional contextual information.
2. At intermediate times, when a few tens of observations exist across all bands. This amount of data is sufficient to derive descriptive statistics, such as the kurtosis and skewness, and to attempt to characterize significant timescales, but the sources continue to evolve in time, and the observations have not covered the entire phase curve. Characterization and broad classification in this regime can still support target selection for follow-up observations, but not rapid early-time studies.
3. At later times, when the observations across all bands span the entire phase curve, and there is perhaps little significant additional information to be gained from further observations. Classification in this regime is effectively retrospective. However, prompt publication by the broker system may still have important advantages for many science objectives.

5.1. *The Variability Probability Distribution Function: Thresholding the Variability of the Galactic Stellar Background*

When an alert is initially issued by LSST, it includes previous photometry of the source, if any, in the preceding 100 days. It is expected that forced photometry at the location of the source will be available only within 24 hr. For many transient sources, this photometry will only consist of detection limits. Nevertheless, despite these sparse data, alert-brokers must be able to determine if the source's variability is significant and worth additional follow-up. The presence of an associated galaxy near the location of the source will greatly aid in this determination, but a galaxy is not always guaranteed to be present (e.g., the source might be at high redshift, and a low surface brightness galaxy may not be detectable, or a deep catalog of static sources may not be available at the alert location), or may not be uniquely identified.

To identify and prioritize interesting transients at early phases, a broker must be able to distinguish them against the background of stellar variability. As the time series of the parts

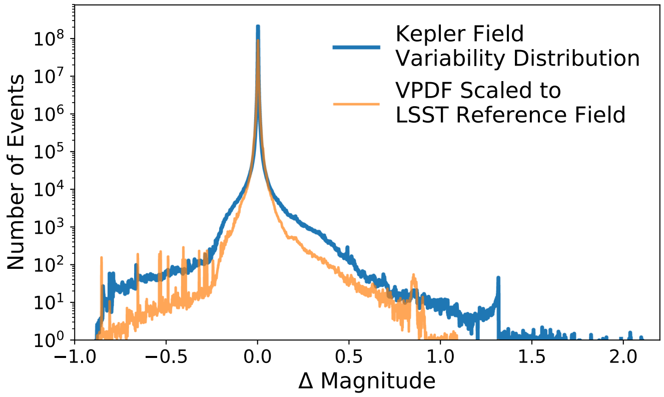

Figure 4. Distribution of variability of stars in the *Kepler* field (blue) obtained from the Quarter 6 time-series data of more than 155,000 stars as a function of change in magnitude from the median. The orange curve shows the resulting distribution after scaling to the Galactic stellar population at the LSST reference field centered on R.A. = 0°, decl. = 0°, based on simulated stellar catalogs from the Besançon Galaxy model.

of the sky scanned by LSST builds up, the bulk of the variable sources for the first few years of LSST operations will be Galactic stars, AGNs, and asteroids without known orbital elements that have been misclassified as stationary difference-image objects (Ridgway et al. 2014, hereafter SR14). Such objects may be of interest to several groups. However, in any search for rare transient events, these relatively mundane sources can be considered contaminants. The alert rates for new sources will plummet as the survey progresses, while the alert rates for new, early-phase transients will remain constant (or improve, as various improvements are made to the survey). We can maximize the early science return from LSST and reduce the number of alerts to process, and thereby the load on brokers, with effective early characterization and filtering. This, in turn, requires a description of the background of stellar variability.

SR14 define the Galactic variability probability distribution function (VPDF), using time-series data from *Kepler* Quarter 6, of over 155,000 stars belonging to different spectral types. The distribution gives the probability of seeing a given variability expressed as the root-mean-square (rms) brightness amplitude for the particular stellar type. However, the *Kepler* field of view covers only a small region on the sky, and the underlying stellar distribution varies with position in our Galaxy. For a given LSST pointing/image covering about 10 deg^2 along a particular line of sight, a distribution of the spectral types of the Galactic stellar population is simulated using the Besançon model[25] (Robin et al. 2003). The variability distribution from the *Kepler* field is then scaled to the corresponding stellar population of the pointing, an instance of which is shown in Figure 4. There are limitations with using the *Kepler* sample, as it does not include supergiants or white dwarf stars, and there are only a few examples of giant stars. However, unlike other studies that follow stars known to exhibit variable behavior (such as OGLE), the *Kepler* sample contains light curves from a large distribution of stars with unbiased selection and is therefore the best currently available data set for this analysis. The final data release from the *Gaia* mission (*Gaia* Collaboration et al. 2016) will include all epoch photometric catalogs, and will hence supersede the *Kepler* sample.

T. Matheson et al. (2018, in preparation) use the change in magnitude, rather than the rms brightness amplitude, to define the VPDF. This parametrization is always useful for a source, even with a single significant observation, as the source required some change in brightness, i.e., a non-zero amplitude in the difference image analysis, to have been triggered as an alert. It can be viewed as a Bayesian prior, formalizing the use of other features that have been used to identify significant variability with sparse data, such as the median absolute deviation (MAD). Rather than a simple threshold for all sources, the VPDF gives the likelihood that the variability of a source at some location on the sky is significantly different from the expected background Galactic stellar variability in the same region. This likelihood can be reduced to a simple binary label (significant/not significant variability).

A variation of the above filtering algorithm is to consider the probability distribution of changes in magnitude, Δm, for a given window between the observations, Δt. This expands the parameter space along the time axis and has been explored for real-time classification of alerts by some groups (e.g., Mahabal et al. 2011). Using light curves of different classes of variable stars and transients obtained with the CRTS survey, they constructed the joint-probability distribution of (Δm, Δt) for each class and used a machine-learning-based algorithm to classify a new alert with an assembled time series. More recently, Mahabal et al. (2017) created a new implementation based on these two parameters, mapping each light curve onto the Δm–Δt space, which in turn was used to construct a 2D feature vector. Such Δm–Δt "images" of light curves belonging to different classes of variable stars were then used to train a convolutional neural network, often used for image recognition and classification in astronomy (e.g., Dieleman et al. 2015; Jacobs et al. 2017), and achieved acceptable results for some classes.

We are also developing an algorithm based on features extracted from the Δm–Δt density distributions of the various classes of varying sources (M. Soraisam et al. 2018, in preparation) to be implemented as a stage within `ANTARES` to help in the timely characterization of interesting/novel events. For example, in this work, among the different kinds of variable stars and transients considered, SNe serve as an example of the cream of the crop. Most of the low-redshift SN light curves in the OSC are not from untargeted searches, but rather they are observed after being discovered by targeted searches. Consequently, the first observation is on average only a week before maximum light, after the source has risen significantly, and even with a conservative estimate for the previous nondetections, these light curves would trivially show high significant variation relative to the Galactic VPDF.

During regular operations, we will use filter stages, such as VPDF thresholding within `ANTARES`, to flag alerts that exhibit significant variability compared to the expected Galactic stellar variability background. These filters use contextual information immediately, without having to train on observations, allowing them to be applied to new surveys that are just coming online. This level of filtering also gives us a throttle; we can raise and lower the threshold dynamically, depending on the region of sky under consideration, or if the alert volume for an image becomes very high, e.g., as a result of poor image subtraction, with many unfiltered artifacts.

Another generalization would be to add the ability to threshold not just on the background stellar variability but also

[25] http://model.obs-besancon.fr/

based on the source's own previous history. If ANTARES has classified a source, we could in principle skip processing of new alerts at the same location until a sufficiently large number of new observations are obtained to merit repeating feature extraction and source characterization. Such an approach implicitly assumes that the current alerts for a source do not add significant new information over the existing history of LSST observations; each source has an "envelope of mundanity" describing an expected range of feature space for new alerts based on previous observations. However, we could generalize VPDF to trigger on alerts that significantly exceed not just the background stellar variability but also this envelope. Such a stage could detect sources that suddenly begin behaving atypically, a la KIC 846285 or Tabby's star. This approach would allow a quantitative treatment of what is presently a subjective assessment that astronomers make for sources that are acting "weird."

5.2. Timescale Characterization

As the source evolves and more observations become available, characterization can become increasingly sophisticated. The bulk of the work of a broker will be with objects in this intermediate regime, as classification here serves the needs of much of the astronomical community, including cone searches and queries against catalogs (e.g., after a gravitational wave trigger is issued), querying for objects similar to or different from some source (e.g., to make a comparison plot), monitoring a previously discovered transient for abnormal behavior (e.g., an SN re-brightening, such as iPTF14hls; Arcavi et al. 2017), and building target lists for follow-up studies (e.g., a project to determine the metallicity of RR Lyrae in the Galactic Halo), among many other use cases.

Where the VPDF and Δm–Δt formalism compare objects based on the timescales probed by the survey, the object itself may have several characteristic timescales. We use the Lomb–Scargle algorithm (Lomb 1976; Scargle 1981) to determine a characteristic timescale for the observations in each passband of each source. While these timescales should agree, and be equal to the fundamental period for periodic sources, the periods determined from different bands may differ due to aliasing. In the case of transients, there is no a priori reason for the timescales computed in different bands to agree.

We use the inverse of the frequency with the maximum normalized Lomb power as the characteristic timescale in each band. We compute the average of the entire power spectrum, as well as the standard deviation, and determine the frequency around the peak where the normalized power drops below one standard deviation above the average. We use the half-width of this range as the uncertainty in the timescale. This is also a useful feature in classification, as the typical cadence will always probe a few cycles for variables with short periods, and these will have a relatively narrow peak in the Fourier power spectrum, *even if the correct period is not determined but an alias is selected from the periodogram*. By contrast, rising transients are akin to windowed impulse functions and have a broad spread, and therefore much larger uncertainties. Additionally, we define the period S/N ratio as the difference between the peak power and the median power, normalized by the standard deviation of the power spectrum, and compute the logarithm of the False Alarm Probability (FAP; Baluev 2008). These four quantities are computed for all OGLE objects in both V and I, as well as the OSC objects in all bands. In both cases, we only use the data that can be mapped to g and i, as described in Section 4.4.1.

We use these features in our machine-learning pipeline (Section 6) in two distinct stages that operate in sequence. In the first, we use all eight timescale characterization features (four each for the g and the i bands) to make an initial variable-SN binary classification. In the second stage, we adopt the features of the band with the smaller timescale uncertainty (in this work, between g and i as those are the only bands available in our combined data set) as the timescale feature vector for the object. We combine these timescale features with more descriptive statistics to train a multiclass variable-SN classifier, which in addition to labeling different variable classes, can be used to identify any SN wrongly classified as variable in the first stage. This structuring enforces a hierarchical taxonomy of classes, with early classification errors detected and fixed by later stages as more data become available.

Gaussian kernel density estimates of the distribution of timescales that were adopted for the second stage for different classes are shown in Figure 5. As can be seen in the figure, intrinsic periodic variables show sharp peaks with low dispersion in their distribution, whereas extrinsic variables and SNe have many different timescales in the sample. The distribution of the shaded regions gives a sense of the relative fractions within each class that exhibit different characteristic timescales, while the offsets between shaded areas between any two classes indicate how easy it is to distinguish them from each other. For example, a simple cut on the timescale at 1 day would be sufficient to distinguish δ Scuti variables from SNe of all types, but it would be difficult to use period alone to distinguish between different types of Cepheid variables. Note that even in this one-dimensional feature space, it is possible to see clear subgroups, e.g., among the RR Lyrae, corresponding to RRab and RRc, even though the label aggregates both subtypes.

While for this work we have elected to use a single number—the timescale with the maximum power in the periodogram—as a feature, a more general technique for timescale characterization would be to simply use the entire periodogram as a feature vector. However, our data set consists of light curves from several different surveys, each of which has a characteristic observing cadence. These different cadences result in peaks in the periodogram that are also characteristic of the surveys. Machine-learning algorithms can use these characteristic peaks in the periodogram to distinguish different surveys from each other, and as all the variables in the data set we have assembled originate from OGLE, this would lead to perfect separation of variables and transients. This separation is not physical, as the algorithm has only picked up on the difference in observing cadences between the different surveys targeting these different classes of objects rather than on the astrophysical differences between variables and SNe.

We use the fasper routine implemented by the FFTW[26] library (Frigo & Johnson 2012) to derive the periodogram, which we wrap in ANTARES with the pyFFTW[27] module. We have found that we can derive a more robust estimate of characteristic timescales for periodic variables in our data set using the power spectrum computed using the multiband Lomb–Scargle algorithm implemented in the VanderPlas & Ivezić (2015)[28] package

[26] http://www.fftw.org/

[27] http://hgomersall.github.io/pyFFTW/

[28] http://www.astroml.org/gatspy/index.html

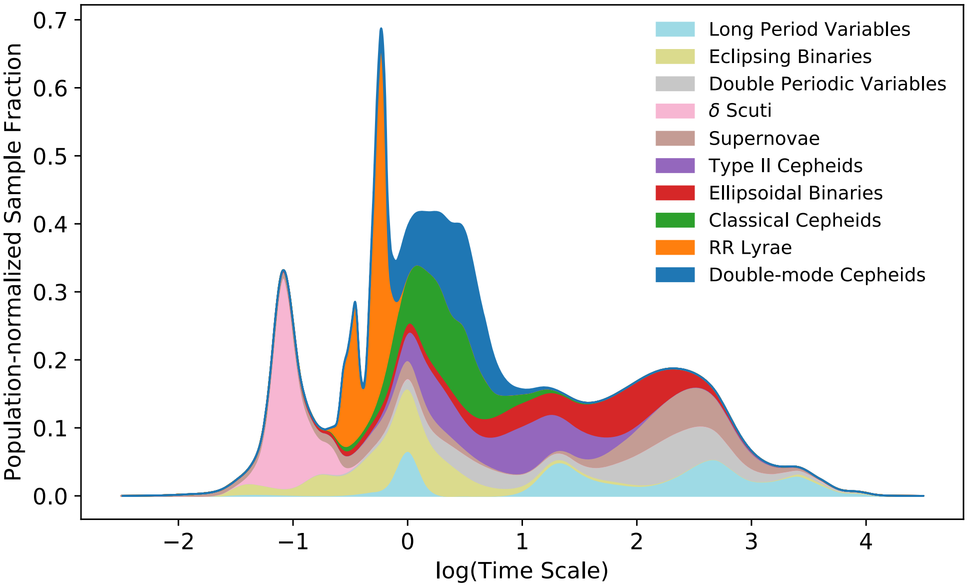

Figure 5. Logarithm of the characteristic timescales (cf., Section 5.2) in days for different classes of objects, plotted as stacked kernel density estimates—a population-normalized cumulative distribution.

Table 2
Description of the 11 Features Extracted from Observations of Each Passband of All Sources

Feature	Description
Autocorrelation Integral, α	The integral of the correlation versus time difference following Mislis et al. (2016)
Entropy	The Shannon entropy assuming a Gaussian CDF following Mislis et al. (2016)
HL Ratio	The ratio of the amplitudes of points higher and lower than the mean
Inter Quartile Range (IQR)	The difference between the 75th and 25th percentile of the magnitude distribution
Kurtosis	Characteristic "peakedness" of the magnitude distribution
Median Absolute Deviation (MAD)	A robust estimator of the standard deviation of the distribution
Shapiro–Wilk Statistic, w	A measure of the magnitude distribution's normality
Standard Deviation/Mean, $\sigma_m/\langle m\rangle$	A measure of the average inverse S/N
Skewness	Characteristic asymmetry of the magnitude distribution
Stetson K	An uncertainty-weighted estimate of the kurtosis following Stetson (1996)
Von-Neumann Ratio, η	A measure of the autocorrelation of the magnitude distribution

Note. Additionally, four passband-independent timescale features are computed (see Section 5.2), producing an $N_{\mathrm{pb}} \times 11 + 4$ dimensional feature vector for every object, where N_{pb} is the number of distinct passbands.

(VanderPlas 2016). However, this improvement is due to the package's implementation of the regularization scheme described in VanderPlas & Ivezić (2015), which is appropriate for variable sources but may not be appropriate for transient sources. Consequently, we only derive one characteristic timescale per band in this work.

5.3. Magnitude Distribution Characterization

As the number of observations grow, we can define more features that can help in classification by incorporating information on the light curve and the flux distribution. The goal of feature extraction is to describe the light curves of the OGLE and OSC sources, but without introducing features that allow a distinction to be made on apparent magnitude (which would introduce bias) or on the observational properties of the survey, such as the telescope and instrument (which would not be informative). Many of these features have already been used very successfully for the classification of variable stars (see references in Section 2.1). We elected to adopt many of the features in R11 as well as in Mislis et al. (2016) and Kim et al. (2014); these are listed in Table 2.

The elements of the feature vector are generally correlated (see Figure 6), while many machine-learning algorithms are designed with the expectation that each element is an independent variable. While feature extraction is intended to lower the dimensionality of the signal, it is possible to reduce the dimensionality even further. We expect correlations both within the feature vector of each band (as many of the features are effectively different measures of the same quantity) and between bands (as the behavior of astrophysical sources is correlated across bands).

We scale the entire feature matrix by removing the mean and decorrelating or "whitening" (Kessy et al. 2017) the data—scaling each column to be uncorrelated and have unit variance. We then use a standard principal component analysis (PCA) to reduce the dimensionality of the feature matrix. We use an $N = 15$ dimensional vector of the PCA features. This choice explains ≈96% of the sample variance.

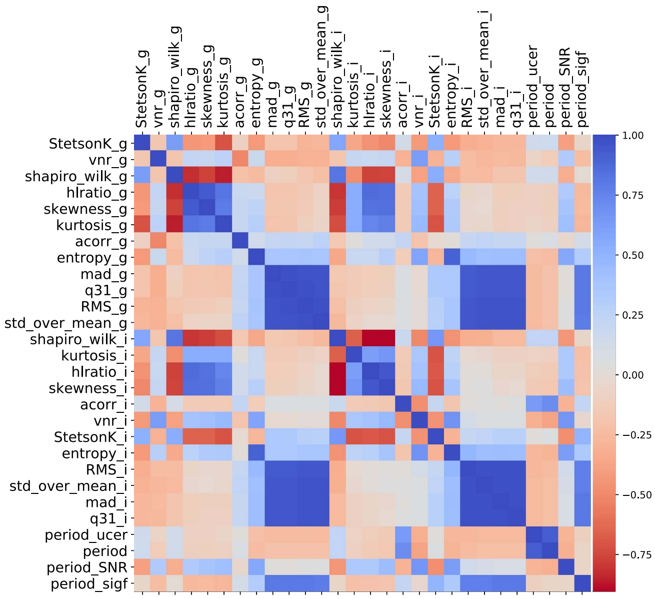

Figure 6. Correlation matrix of the data set. Features that characterize the magnitude distribution extracted from the two bands in this work, *g* and *i*, have many features that are strongly correlated within and between each band, with the color bar at right displaying the coefficient of correlation.

5.3.1. Using t-SNE as a Feature Space Visualization Tool

While the explained variance from the PCA is a measure of how well the low-dimensional feature vectors represent the higher dimensional input, it does not help us determine whether the features themselves are predictive—if they can be used effectively for classification. That question will ultimately be answered by training and validating the machine-learning classifiers; however, even without machine learning, we can examine if the feature vectors are likely to be useful by constructing a t-Distributed Stochastic Neighbor Embedding (t-SNE', van der Maaten & Hinton 2008).[29]

We use the `multicore-tsne` implementation[30] (Ulyanov 2016) of the Barnes–Hut variant of the t-SNE algorithm (van der Maaten 2014), which can produce a 2D or 3D representation of a high-dimensional space, clustering similar points together. The algorithm constructs a k-D tree of all the points, and computes the Euclidean distance between each point and its *k* nearest neighbors using a Student's t-distribution to convert this distance into a probability that the two points are similar.

The algorithm then attempts to find a low-dimensional space that preserves this probability, using a gradient descent algorithm, minimizing the sum of the Kullback–Leibler divergence (a measure of the divergence between two distributions). The number of nearest neighbors is an input to the algorithm known as "perplexity;" however, it is largely insensitive to this choice, and we obtain very similar embeddings for perplexity between 100 and 300, such as that in Figure 7. The algorithm is unsupervised—i.e., it constructs clusters of similar points without any knowledge of the labels. Classes that are well-separated in a t-SNE visualization can generally be distinguished from each other by a machine-learning classifier; however, the converse does not always hold.

There are several caveats to t-SNE visualizations, and neither distances between nor the sizes of the clusters may be informative (Wattenberg et al. 2016).[31] Additionally, the stochastic nature of the algorithm means that different runs with the same data, or different partitions of the input set with a different class balance, can produce different, although qualitatively similar results. We therefore use the algorithm for visualization rather than classification.

We use our PCA feature vector (described in Section 5.3) as the input to the t-SNE as these data are largely insensitive to survey characteristics and the quality of the light curves. Using the light curves directly would require that all objects be interpolated on to a common grid, and the resulting t-SNE would likely be much more sensitive to gaps in the light curve or differences in S/N, which could lead to clustering that does not reflect astrophysical differences between classes. We examined several embeddings derived from our data set, and found that this clustering is not perfect, and some groups such as RR Lyrae appear to be divided into subgroups or clumps. We visually inspected members of the subgroups and find this division to be a reflection of reality, with the t-SNE separating RRab from RRc and RRd subtypes, even with the relative low dimension of the input feature set. Additionally, while it was not possible to distinguish among the different subtypes of Cepheid variables (classical, double-mode, and type II) using only their characteristic timescale, as seen in Figure 5, these groups are distinguishable in the t-SNE plot.

The t-SNE also throws into sharp focus the scale of the imbalanced learning problem. It was necessary to suppress long-period variables in the visualization as they outnumber the next largest class by an order of magnitude, and the structure was dominated by clusters of Miras. By default, many machine-learning algorithms will optimize "accuracy"—the overall fraction of the predicted labels that are correct—at the expense of "recall" or "sensitivity" to smaller classes. Simply, in a very imbalanced data set with, e.g., 997 members of a red class, and only three members of a black class, classifiers are very likely to simply "put it all on red," even if the most interesting events are the rare ones. This has implications for several problems, from fraud detection in financial data to identifying electromagnetic counterparts for gravitational wave sources.

The t-SNE is best employed as a diagnostic tool to identify potential classification challenges with the data set and the chosen feature representation, prior to machine learning. This can help avoid three common cascaded problems often encountered in applied machine-learning work: the adoption of naive classification algorithms, the use of inappropriate metrics, and the fine tuning of the algorithms to optimize those metrics.

5.4. Light-curve Characterization

Once sufficient observations have been obtained to cover the source's phase curve fully, complex feature extraction can be performed, even if computationally intensive, as the object is unlikely to require any reprocessing. Classification of transients in this regime is retrospective, as they will have faded to below the level of the sky. Nevertheless, classification at these times is extremely important and can serve a variety of research programs. In this section, we consider a project with one such

[29] https://lvdmaaten.github.io/tsne/

[30] https://github.com/DmitryUlyanov/Multicore-TSNE

[31] https://distill.pub/2016/misread-tsne/

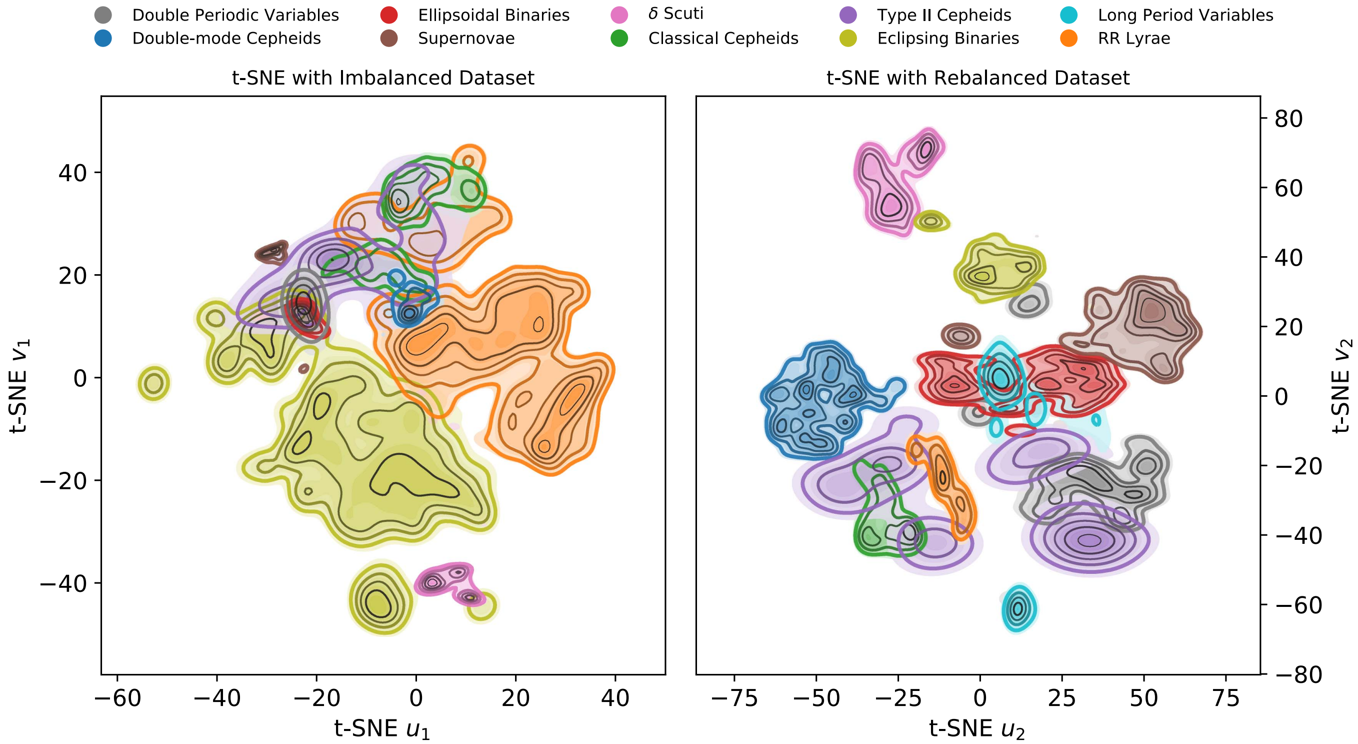

Figure 7. A t-distributed Stochastic Neighbor Embedding (t-SNE) "Petri dish" of the feature matrix after scaling and PCA with whitening on the imbalanced (left) and rebalanced (right) data set. The t-SNE embedding attempts to preserve points that are similar to each other in the full feature space, clustering them together in a lower dimensional space that is easier to visualize. The t-SNE embedding is determined without using any class labels, and points are colored by class after they are embedded in the low dimensional space. We represent the $\sim 10^2$–10^5 points in each class using a bi-variate kernel density estimate (KDE). Neither the two t-SNE axes nor the sizes of the clusters are physically interpretable. With the class imbalance, the t-SNE has performed relatively poorly, dispersing the largest classes (eclipsing binaries and RR Lyrae) and separating them from each other, but without distinguishing between the minority classes. We account for the drastic class imbalance in the real data sets using a combination of techniques in Section 6. When applied to a balanced data set, the t-SNE separation improves dramatically, with most of the classes clearly distinguished.

requirement—the need for an extremely high-purity sample of SNe Ia for cosmological studies, extracted by determining the subclass for the SN output of the previous stages, e.g., by the LSST Dark Energy Survey Collaboration (DESC). For this level of classification, we follow the approach of L16 and construct a much more complex feature vector to describe the events, using Gaussian process regression followed by wavelet decomposition to model the events.

Observations from real astrophysical surveys are unevenly spaced, have gaps due to weather losses, and have heteroskedastic errors with faint objects having much lower S/Ns than bright objects. Even with perfect data, objects in the distant universe are redshifted and undergo time dilation, making a direct comparison with low-redshift objects nontrivial. A precise comparison requires knowledge of the "K-correction" (originally described in Oke & Sandage 1968), but this cannot be computed without knowledge of the underlying spectral energy distribution (SED) of the object, and implicitly, its astrophysical class, which is the same quantity that we wish to infer from the data. Furthermore, many feature extraction and machine-learning algorithms impose additional requirements on the data, such as evenly spaced observations and the absence of missing data.

To make use of these algorithms, we use various methods to generate smooth, evenly sampled representations from the noisy, sparse observations. Because alert-brokers such as ANTARES will not have a priori information about the class of the object under consideration, these methods cannot rely on templates of various astrophysical classes or on simple parametric representations of light curves, as none are sufficiently general to describe all possible variable and transient phenomena. Additionally, these methods must be robust enough to work despite the limitations of observations and must be computationally efficient to work at scale with the LSST data rate.

5.4.1. Gaussian Process Regression

A Gaussian process (MacKay 2003) models every point in some continuous input space (time in the case of light curves) with a normally distributed random variable, which defines the distribution for how far the point may lie from the mean. Any finite collection of Gaussian random variables follows a multivariate normal distribution. Conditioning this model on the observations—the regression—solves for the mean of this multivariate normal distribution as a function of the continuous input variable.

Additionally, the regression problem is simplified by imposing a parametric function relating two points in the input space, t_i and t_j, to each other—a covariance model. Often, this "kernel" function is expressed solely in term of the separation of the two points, $|t_i - t_j|$; in this case, the covariance model is described as "stationary." This parametrization can be quite flexible as any linear combination of kernels remains a valid kernel. Consequently, the Gaussian process framework is very adaptable and is used to describe a wide array of data from different fields. We used the george[32] Python module (Ambikasaran et al. 2015) to perform the Gaussian process

[32] http://george.readthedocs.io/en/latest/

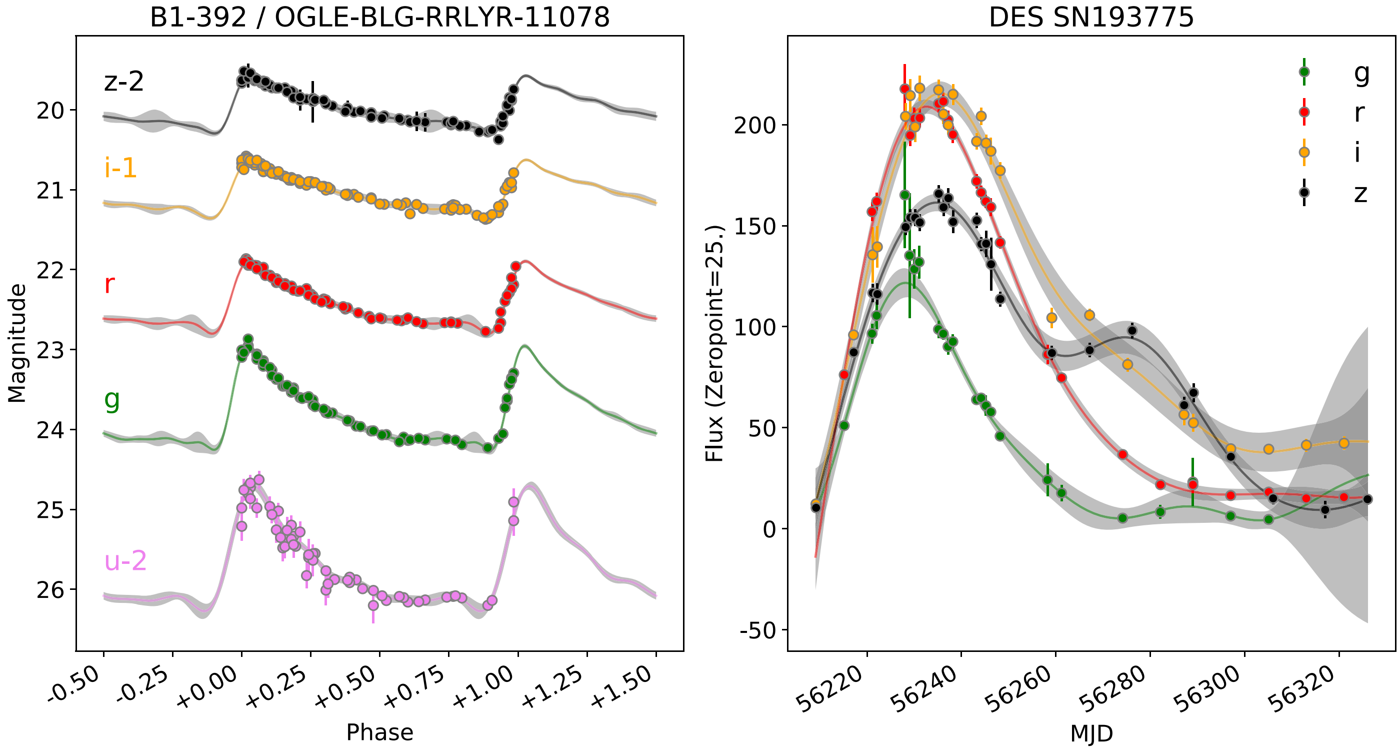

Figure 8. Gaussian processes are flexible enough to produce a smooth representation of multiband light curves of both periodic and transient objects reported in flux or in magnitudes using appropriate kernels. This is illustrated with model fits for an RR Lyrae (left) from a survey of the Galactic Bulge with DECam (courtesy of P.I. A. Saha) and a simulated SNPhotCC SN (right). The RR Lyrae, B1-392, is chosen as it appears in Saha & Vivas (2017) and is also present in the OGLE data set (OGLE-BLG-RRLYR-11078) used in this work, while the SNPhotCC light curve has a redshift of 0.4089, near the median redshift of the DES and Pan-STARRS SN Ia samples.

regression for each SN light curve, adopting the smooth Matern 3/2 kernel. The output is insensitive to the choice of the kernel parametrization, with Matern 5/2 and squared exponential kernels performing very comparably, and the differences between the outputs are comparable with the uncertainties on the data.

While Gaussian processes are a very general technique and can be used for sophisticated modeling of observations while accounting for the measurement uncertainties, we use it as a generalized method of interpolating the light-curve observations onto a common grid. Gaussian processes are computationally intensive ($\mathcal{O}(N^3)$ for a set of N observations); the limited number of observations in a typical light curve and the simplifications we can make by modeling 1D time series data with stationary kernels serve to keep the computational cost extremely low. Example Gaussian process regression models are shown in Figure 8 for an SNPhotCC Ia as well as a periodic variable star.

5.4.2. *Wavelet Decomposition*

Wavelet transformations are a common technique used to express a square integrable function as an orthonormal series of wavelets—functions that begin at zero, increase, and then fall back to zero. The technique can be considered as a harmonic analysis, with complex signals expressed as the sum of a series of simple pulses. The transformation preserves the overall shape, but not the time extension. This allows signals with the same shape but different characteristic timescales to be compared to each other using only the coefficients of the orthonormal series (frequently called the "detail coefficients"). There are several different families of wavelets to construct the orthonormal series, each of which has different properties. The choice of which family of wavelets to use is typically made to allow an approximate reconstruction of the original signal with only a few terms of the full series, i.e., with only a few detail coefficients. Consequently, wavelet transformations are frequently employed for lossy compression of the signal.

Wavelet algorithms have been tested extensively on the SNPhotCC data set by Varughese et al. (2015) and L16. This is the first work to validate wavelet-based methods on real observations. We use the `PyWavelets`[33] package within our pipeline, and we explore two wavelet-decomposition techniques to select the resulting feature set to be used in the machine-learning classifier: (1) the discrete wavelet transformation with BAGIDIS used in Varughese et al. (2015) and (2) the stationary wavelet transformation used in L16.

The BAGIDIS (Basis Giving Distances) methodology (Timmermans & von Sachs 2010) uses a basis pursuit algorithm based on the methods of Fryzlewicz (2007). The basis pursuit method decomposes the smooth input light curve onto an unbalanced Haar wavelet basis. The associated basis coefficients are ordered according to the strength of their effect on the shape of the signal. This method allows a small number of initial coefficients to encode the bulk of the information, as illustrated in Figure 9.

The second wavelet-decomposition method, the Stationary Wavelet Transform (SWT), is widely used for edge detection and noise reduction of images. For a stationary wavelet transformation x, with x_n coefficients, a new filter x_n^m is obtained by inserting $2^{(m-1)}$ zeros or "holes" between each xth

[33] https://pywavelets.readthedocs.io/en/latest/

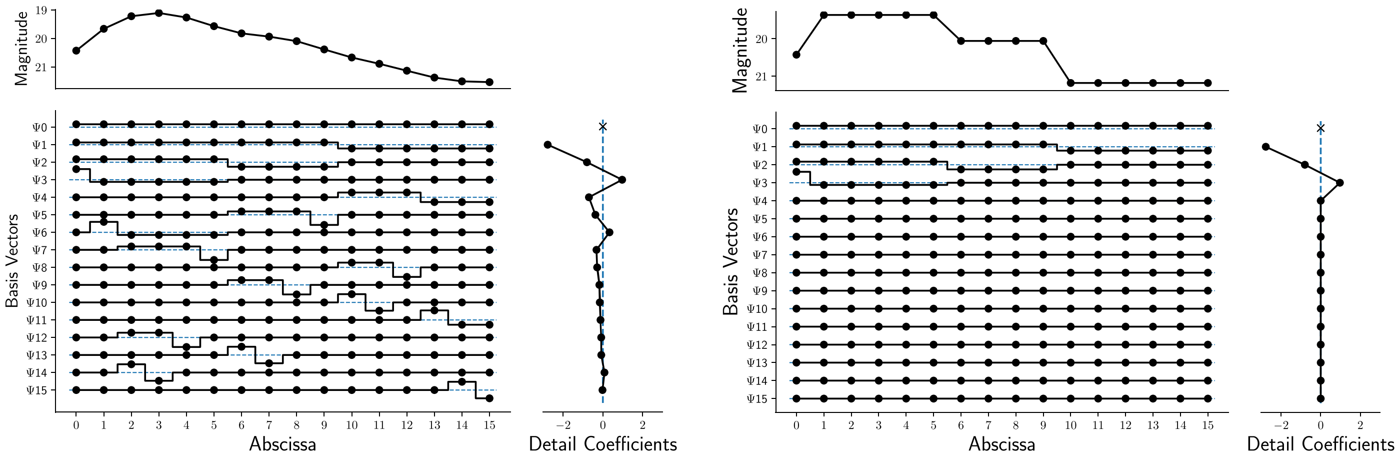

Figure 9. BAGIDIS decomposition (left) and reconstruction (right) of a Type Ia light curve. The unbalanced Haar basis vectors are shown on the left side of each plot, and the breakpoints correspond to where a given basis vector changes sign. The corresponding detail coefficients are shown on the right. The reconstructed BAGIDIS curve is a coarse approximation, taking only the first four coefficients and setting the rest to zero. Note that the first component captures the average value of the light curve and is not used for classification of SNe as this would introduce a cosmological bias.

coefficient, effectively applying a sequence of low-pass filters operating on different scales from the input. The advantage of the SWT method is that it can be used with different families of wavelets and possesses translational invariance (the input abscissa can be shifted by a common offset). We elected to extract wavelet features using only the Daubechies and symlet wavelet families. Symlets were used in L16, and only differ slightly (less asymmetric) from the more common Daubechies family. The presence of hundreds of correlated wavelet features would simply introduce variance into any machine-learning classifier. Consequently, we use PCA for dimensionality reduction of the SWT coefficient feature space as described in Section 6.5.1.

6. Machine-learning Pipeline

6.1. Stages: The Functional Unit of Alert-broker Pipelines

As described in Section 3, the ANTARES pipeline as a whole is comprised of a handful of discrete stages, each of which specifies a set of actions the broker must perform on the sources in the alert stream. Our architecture (see Figure 1) defines different sequential levels of processing. The stages within each level can be run in parallel, with the output being coalesced, allowing each alert to be annotated by multiple processes and providing each subsequent level with more information. Sources that do not meet filter criteria for further processing or do not have sufficient information for a stage are diverted. Once an annotated source reaches the bottom of the ANTARES architecture, it is stored in our locus-aggregated alert database, made publicly available, as well as broadcast to external alert-brokers. Additionally, we will filter out the most rare alerts, broadcasting them separately to facilitate coordinated follow-up studies across the electromagnetic spectrum.

In Section 1.1, we stated three questions to motivate the development of stages for the pipeline in this work: how effective is machine learning at (a) early-time categorization of variables and transients, (b) classification into broadly separable astrophysical classes without full phase curve information, and (c) late-time retrospective classification aimed at producing a high-purity sample of objects? Each of these questions led to the construction, in Section 5, of a stage to encode the information contained in the light curve as a low-dimensional feature vector. Having computed the matrix of feature vectors for the data set, we then define filters to *select* alerts and train machine-learning algorithms to *categorize* and *classify* them—examples of core machine-learning stages that will comprise the ANTARES pipeline. These stages parallel our motivating questions of machine-based selection, categorization, and classification.

We plan to implement iterated semi-supervised learning by using this pipeline to label new data sets and then using those sets to retrain our classifiers. As we do so, the stages we use will undoubtedly evolve significantly. This iterative process of machine learning, classifier retraining, and stage modification will continue throughout the entire lifetime of the LSST project. Consequently, brokers will need to adopt version control, not only for their code base but also to track the provenance of the library of data sets, the feature matrices, the different splits used for training and testing, and binary representations of the filtering and classification stages: the "Touchstone" illustrated in Figure 1. We structure the different stages hierarchically, to reflect how they are used to process sources with different amounts of phase coverage, hence with different amounts of information encoded in the feature vector.

6.2. Putting It All Together: Constructing an Alert-broker Pipeline

When an alert is received by the pipeline developed for this work, the first processing stage consists of a filter using the most basic features computed in Section 5.1. This stage is filtering as opposed to classification, and is constructed from the best available unbiased catalog of stellar variability available—at present, the *Kepler* sample. Consequently, it involves no machine learning, and we do not consider it further in this section.

As additional observations of the target are acquired, we can perform more advanced characterization of the time-series, as described in Sections 5.2 and 5.3, and attempt classification. We expect that stages that operate in this intermediate regime, where some observations have been acquired after the initial alert but the full phase curve of the event has not been probed, will serve the bulk of the astronomical community's needs.

To begin with, we attempt to determine whether the object is a variable or a transient. The only transient type considered

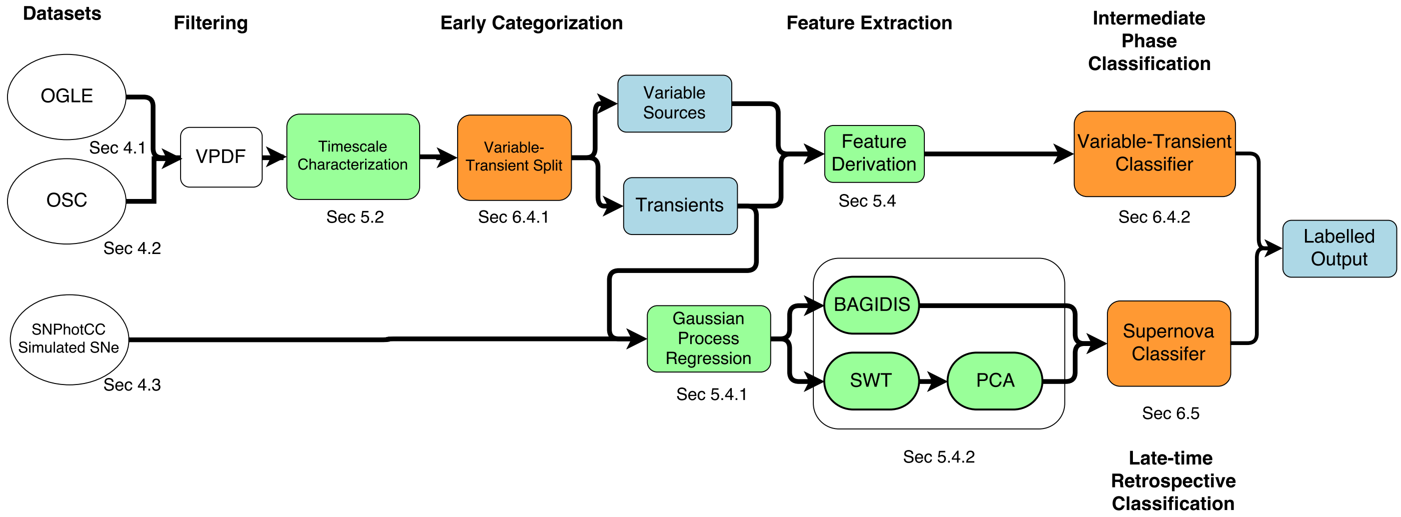

Figure 10. Schematic of the ANTARES machine-learning pipeline, corresponding to the yellow bracketed region in Figure 1. The pipeline stages are designed to examine three different use cases for an alert-broker, corresponding to the different amount of data in the alert packets for each source, as described in Section 1.1. The input data sets are described in Section 4, the feature extraction in Section 5, and the various stages in Section 6. The specific subsections in this work corresponding to each element of the pipeline are indicated below the stage. Stages that extract feature information are indicated in green, categorization or classification stages are indicated in orange, and outputs are colored in blue.

here are SNe, though LSST will produce alerts from other astrophysical classes of transients. This is not a significant limitation for distinguishing between variable and transient sources, and the choice to use only SNe in this work reflects the labelled data sets that are readily available for training. As we use this work for semi-supervised learning with Pan-STARRS, we will be able to construct a much more homogeneous training set. We do not expect stages like this to become substantially more complex, as they are ultimately doing binary classification.

The next stage after characterizing the alert and determining if the object is a recurring variable or transient is to attempt to determine its class. This stage is intrinsically more complex than variable–transient separation as it involves multiclass labelling, a problem that is exacerbated by the extreme population imbalances in our combined OSC and OGLE samples. Even with a more homogeneous data sample drawn from a single survey, this imbalance will persist as different classes of objects have very different astrophysical rates, and therefore any classifier attempting to tackle this problem must adopt strategies to deal with unbalanced, and likely nonrepresentative, training sets.

Finally, once enough of the time evolution of the source has been probed, we can perform complex feature derivation, such as the quantities extracted by the processes described in Section 5.4. While this processing is typically CPU intensive, it need not be repeated for transient objects and is unlikely to need repeating for variable sources until a sufficiently long time baseline is covered by the observations, which would allow us to look for long-term variability. In this regime, we look at a specific use case of scientific interest to many groups—the extraction of a high-purity photometric sample of SNe Ia for cosmological investigations.

The pipeline and stages developed for this work are represented graphically in Figure 10. Each of the stages of our machine-learning pipeline requires the training and validation of a classification algorithm. We use a random forest algorithm for all of the machine-learning tasks in this work. While there are several alternative machine-learning algorithms we could employ, random forests have several advantages, described in the following section, that make them particularly suitable for this study.

6.3. Decision Tree Learning and Random Forests

Decision tree learning allows for a mapping from input features to output classes by means of a series of selection rules. An individual tree is trained by generating a selection rule based upon whichever feature and threshold give the maximum information gain, where information gain can be determined by several different metrics. The two most common choices are the Gini-impurity,

$$I_G(p) = \sum_{i\neq k} p_i p_k, \tag{1}$$

or the entropy,

$$I_E(p) = \sum_{i=1}^{J} p_i \log_2 p_i, \tag{2}$$

where p_i is the percentage of the ith class in the sample of J classes present at the child node of each split. The percentages are normalized to sum to unity. The information gain is defined as the difference between the metric computed for the parent node and the weighted mean of the metric for the child nodes (i.e., after the selection rule is imposed). The generation of selection rules proceeds recursively until objects in the training set in a given area of a tree are all of a single class, or until a given threshold of tree depth.

An individual decision tree classifier that is grown "deep" is typically overfit, i.e., has a very high variance. The opposite is true for a decision tree classifier that is grown "shallow." It will have low variance but high bias, i.e., is underfit, and does not capture the relationships between the feature vectors and target outputs. Using a single decision tree for classification generally yields poor performance, so decision trees are rarely used on their own for machine learning (James 2013). Instead, powerful ensemble methods have been developed that rely upon the averaging of errors between ensembles of trees. A popularly used method is random forests (Breiman 1999).

A random forest classifier is a learning method that trains an ensemble of decision tree classifiers and takes the mode of the classification results as the output. Random forests have been used to great effect in a number of fields, and in particular for photometric SN classification (Lochner et al. 2016), as well as in other areas of astronomy (Dubath et al. 2011; Carrasco et al. 2015). L16 and others have demonstrated that random forests show similar performance to other classification algorithms in the context of astronomical time-series data sets. Therefore, we employ random forests for all three learning tasks—early variable/transient categorization, intermediate variable and transient classification, and late-time SN Ia/non-Ia separation—that we consider in this work.

The use of an ensemble of decision trees in the random forest has the effect of greatly decreasing the variance of the classifier, but increasing the bias. The algorithm samples the instances in the training set repeatedly with replacement (randomly in the case of a random forest). If there are N features for each instance, a threshold of n can be specified such that $n \ll N$. Only n variables randomly selected from the N features are then used when growing the individual decision tree. Each tree is grown as far as possible until all objects in the end branches of a tree are of a single class. When seeking to classify a new object, the input features are processed by all of the trees in the forest. Each tree outputs a classification based upon its selection rules and "votes" for the class it determined. The mode of the class selections determines the output classification of the forest. By aggregating ensembles of decision trees, random forests avoid the bias–variance trade-off that is inevitable with a single decision tree.

Random forests also provide several useful metrics "for free": the out-of-bag (OOB) decision function and relative feature importance. For each random subset of the data selected in the first step of training, there is a portion of the data that was not used in the building of the decision tree (the OOB data). For any given object, there will be approximately one-third of the total number of trees that never used the object in the tree generation (Breiman 1999). The decision function is then the classification result from running each of the OOB objects through the two-thirds of the trees that did not use them for training. An error estimate is generated by comparing the decision function to the labelled classifications. Random forests also provide a natural robust estimate of the feature importance. At each successive split made on a given feature, m, when training a single decision tree, the algorithm computes the decrease in the weighted impurity. For a forest of decision trees, the weighted impurity decrease for each feature can be averaged across all trees, and the features can be ranked by this average.

Random forests effectively memorize the data used for training. For a thorough analysis of generalization error, it is essential that either independent data be used for testing or the OOB error estimate can be taken. The OOB error estimate has been shown to be a biased estimate of the generalization error, and an external testing set is still important to validate results Bylander (2002). We use the `RandomForestClassifier` implemented by the `scikit-learn` Python package[34] (Pedregosa et al. 2011) throughout our pipeline. We describe the development and training of each of the stages of our machine-learning pipeline in the following sections.

6.4. Classification of Variables and Transients

6.4.1. Variable–Transient Separation

For variable versus transient separation, we assemble a vector of the features computed in Section 5.2 for each of the two passbands (g and i) for each object in the combined OSC and OGLE sample. The advantage of constructing the vector from only timescale features is that such information is frequently reported in external catalogs of variable sources such as *Gaia* and Pan-STARRS. This will allow us to populate the feature vector, even in the absence of many LSST observations, using surveys that observe in a different set of bandpasses.

All objects in the OSC sample were considered transient, irrespective of subtype, whereas all objects in the OGLE sample were considered to be recurring variables. For both the OSC and the OGLE data, only the g and i passbands were taken after passband mapping. If they were not present, the light curve was discarded. As Kim et al. (2014) note, several period-related features, such as the period S/N ratio, the period itself, and associated uncertainty, allow periodic and non-periodic objects to be distinguished. Requiring timescale information from two bands increases our sensitivity to this split, as periodic variables often exhibit similar behavior in different passbands, whereas transient sources may not. The OGLE objects generally have an observation period that extends far longer than that of the OSC SNe, so we avoid using features that operate as a proxy for the duration of observation. This is a limitation that arises from the heterogeneity of our sample. We expect to be able to derive more complex features for this stage when operating on homogeneous data sets.

To classify the OGLE and OSC light curves as recurring variables or transients, we used a random forest classifier with balanced class weights. This rebalancing helps account for the ~8 to 1 ratio of variable to transient objects in the sample. The rebalancing adopted here is simple: the random forest applies a weight to the minority class inversely proportional to its percentage of the training data. These weights are used in the development of the decision tree in two places—in the weighting of the Gini impurity coefficient and in the final vote tallying for an object; the decision becomes a weighted majority vote in accordance with the balanced class weights as opposed to a simple majority. There are many alternative approaches to rebalancing the training sample, and we will examine some in other stages of our pipeline. However, for this learning task, we can use cross-validation to show that the simple weighting approach suffices.

To evaluate the consistency of the classifier, we calculate the evaluation scores by taking the average over a fivefold cross-validation. The data were split up into five different training and testing sets, or "folds." Each fold's training set was used exactly once, with the remaining data used for testing in each iteration. The classifier was run using 200 decision trees, though the classification performance was not highly dependent on the number of trees chosen, as we expect for a simple binary decision task. For the subsequent stages in the pipeline where we consider increasingly complex problems, we adopt increasingly sophisticated approaches to tuning the classifier. We report the results of this classification stage in Section 7.

[34] http://scikit-learn.org/stable/

6.4.2. Variable and Transient Classification

While variable–transient separation is of utility for many follow-up studies, a labelled feed of alerts with a high confidence of belonging to a particular astrophysical class is very desirable and serves a large range of scientific interests. As the full time evolution of the sources has not been probed, the input features must be chosen carefully to be robust to outlying photometry and stabilize as more data are added. If this condition is not satisfied, it is likely that the predicted classification will not be stable, i.e., it will change frequently as more observations are added. We construct the feature vector for each source following Section 5.3.

For the binary classification problem of variable–transient separation, class labels are aggregated into either variable or transient for the training, and consequently, the class imbalance of the combined OGLE and OSC data set is much reduced. However, as we are now attempting to classify variables and transients into their subtypes, we cannot aggregate labels. This fundamentally multiclass problem requires that we contend with the extreme class imbalance in our data sample. We account for the class imbalance using a combination of techniques: undersampling the majority class, aggregating minority classes that are very similar into super classes, and synthetic minority oversampling.

6.4.2.1. Undersampling and Aggregation to Reduce Class Imbalance

With $\mathcal{O}(10^6)$ members, the long-period variables in the OGLE data set outnumber every other class in the combined sample of OGLE and OSC objects. Therefore, we undersample this class by using 10% of the total samples for training and testing, reserving the rest for validation. This makes the number of available long-period variables comparable to the next biggest classes in the OGLE sample—RR Lyrae and eclipsing binaries.

Additionally, we can aggregate some classes that share many similarities together in the training sample. For this work, we elected to combine eclipsing binaries and ellipsoidal/contact binaries together under the Algols label, and we combined classical Cepheids, double-mode Cepheids, and type II Cepheids together under the "Cepheid Variables" label.

This aggregation is reasonable given the design goals of this stage, which we expect will process $\sim$10 observations from LSST for each source. This is unlikely to be sufficient to distinguish between the aggregated subtypes; high-confidence classification into such subtypes will require a long baseline of observations to constrain variability on extended timescales.

While these two methods help address some of the class imbalance, they are insufficient by themselves, as many of the classes of variables still outnumber the transients by two orders of magnitude. The class distribution of the full data set after undersampling and aggregation is listed in Table 3.

Table 3
Sample Sizes of the Classes in the Combined OSC and OGLE Data Set after Feature Extraction from Light Curves with a Sufficient Number of Observations, and after Undersampling of Long-period Variables and Aggregation

Class	Number of Objects
Algols	39,452
RR Lyrae	38,243
Long-period Variables	28,904
Cepheid Variables	8672
δ Scuti	2844
Supernovae	1048
Double-period Variables	136

Note. Despite these techniques, the sample remains extremely imbalanced with classes of interest, such as supernovae, being underrepresented relative to variables by two orders of magnitude. We use SVM-SMOTE (Section 6.4.2.2) to generate synthetic samples from minority classes and improve the classifier's sensitivity to class boundaries.

6.4.2.2. Dealing with Extreme Class Imbalance: Synthetic Minority Oversampling

Random forests include the ability to account for class imbalance using re-weighting or random sampling with replacement. Randomized oversampling duplicates members of the minority class in the training set, thereby preventing classifiers from being dominated by the majority class. However, this technique is naive when the class imbalance is drastic, as in the case of the combined OGLE and OSC data set, with some classes having two to three orders of magnitude more members than others. Each member of the minority class in the training set is duplicated several times, and we found unsurprisingly that this leads machine-learning classifiers to overfit the duplicated samples at the expense of precision and recall when validated with test samples. Therefore, we employ the Synthetic Minority Oversampling Technique (SMOTE; Bowyer et al. 2011) to generate new samples of the minority class rather than simply to duplicate existing members.

SMOTE is fundamentally a multidimensional interpolation scheme. The standard implementation of the algorithm first constructs a k-D tree of the training set and then determines which classes are in the minority and require oversampling. When resampling to create a new synthetic data point, the algorithm selects one of the k-nearest neighbors in the feature space of a selected minority class member and constructs the vector between the selected point and its neighbor. The algorithm draws a random number between 0 and 1, and multiplies the vector by this number to generate a new point. The basic implementation of SMOTE can be affected by outliers in the minority class, causing the algorithm to extrapolate new points in regions of the feature space where the minority class is not otherwise represented. We therefore employ the more sophisticated SVM-SMOTE (Nguyen et al. 2011) variant of the algorithm. This variant constructs a Support Vector Machine (SVM) to determine boundaries between classes, and interpolates new members near the border lines, to increase the machine-learning classifier's sensitivity to class separation. We utilize the SVM-SMOTE implementation from the `imbalanced-learn`[35] (Lemaître et al. 2017) package in our pipeline.

We emphasize that it is critical that SMOTE and its variants be applied only to the training set and not to the full data set prior to the train/test split—the latter would only result in overfitting of the interpolated samples rather than producing any improvement by accounting for the imbalance between the classes. There are even more sophisticated techniques that combine oversampling methods such as SMOTE with undersampling techniques like Edited Nearest Neighbors (ENN) to ensure the resampled training sets are free of noise. Batista et al. (2004) compare several methods for rebalancing, and we

[35] http://contrib.scikit-learn.org/imbalanced-learn/stable/index.html

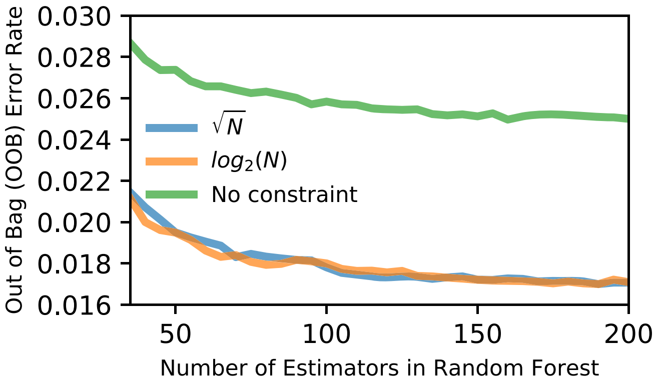

Figure 11. Tuning a random forest by examining the out-of-bag (OOB) error rate vs. number of decision trees (estimators). The curves correspond to different constraints on the maximum number of features that each tree can consider from our $N = 15$ dimensional feature vector.

expect many of these techniques to become more prevalent in the domain of astrophysics as groups are forced to contend with the imbalanced samples that will be produced by wide-field synoptic surveys.

However, even minority oversampling techniques such as SVM-SMOTE may be insufficient to address classes that are completely underrepresented and that have a high dispersion in feature space. We have deliberately included the double-period variables with only 136 members in the sample to illustrate this. The class consists of objects that show more than one fundamental period but do not appear to belong to other astrophysical classes. The class itself then is ill-defined, with members sharing properties with other classes and consequently having a large dispersion in feature space. This makes the construction of class boundaries using SVM sensitive to the specific objects included in each of the k-fold training samples. The naive use of resampling techniques in this scenario may lead to biased classifiers that capture spurious relationships generated by interpolation and the target outputs. Astronomers employing such imbalanced learning techniques must make an informed choice as to how to rebalance the training sample, rather than employing these sophisticated algorithms as black boxes.

6.4.2.3. Training and Cross-validation

We use ninefold cross-validation, splitting the data set of $N = 15$ dimensional feature vectors (see Section 5.3) into nine different training and testing samples. We reserve 60% of the data for testing and use the complement for training. We apply SVM-SMOTE to each of the ninefold samples to generate a new balanced training set for machine learning with the random forest algorithm. We tuned the algorithm by examining the OOB error rate for random forests with a different numbers of trees (see Figure 11) and selected an ensemble size of a 100 decision trees; increasing the number further only results in a marginal decrease in the OOB error and an increase in the accuracy.

We also examined the OOB error rate versus the maximum number of allowed features for classification, testing the behavior if we imposed no constraint, or limited the maximum number of features to at most $\log_2(N)$ or $\sqrt{N}$. We found little difference between the two methods of restricting the number of features. However, both of these methods consistently outperformed no constraint whatsoever, at all ensemble sizes. Closer examination of the outputs from the classifier suggests that a lack of constraint allows the decision trees in the ensemble to adopt selection rules based on features that produce very little change in the entropy, whereas restricting the maximum number constrains the trees to select features that maximize the Kullback–Leibler divergence, i.e., the information gain. This behavior is exactly analogous to the optimization step of the t-SNE visualization (Section 5.3.1). We report on the results of the classifier in Section 7 using a forest with 100 trees, trained to optimize the information gain, and with the maximum number of features limited to $\log_2(N)$.

6.5. Supernova Classification

Acquiring enough observations to cover the full time evolution of the sources necessarily means that classification for transient objects is retrospective. However, this regime is still of interest for several groups engaged in population studies. One of the biggest challenges of these studies will be discriminating between the class of interest and “impostors”—objects that have very similar light curves, but different underlying astrophysics. This is the key distinction to the stages described in Section 6.4 that are designed to separate sources into broadly distinct astrophysical classes using features derived from only a section of the light curve. Brokers like ANTARES must extract the maximum amount of information available in the light curve and build complex classifiers that are capable of discriminating between the class of interest to meet the requirement for a high-purity photometrically selected sample.

The specific use case we consider in this section is the need for a stage capable of producing a photometric sample of SNe Ia for a cosmological study to constrain any evolution of the equation of state of the dark energy. Several groups have considered this problem in detail since SNPhotCC became available, offering a benchmark data set and a rich literature against which we can compare results. We adopt a variant of the wavelet-based classification methods employed in L16 to extract information from the SN light curves.

Nonparametric representations of the time evolution have a distinct advantage over template-based methods for alert-brokers: the same features can be effectively extracted from most of the light curves, whereas template fitting or parametric representations are typically tuned to at most a few classes of astrophysical objects. We construct the feature vector by modeling each light curve with a Gaussian process (Section 5.4.1) and applying wavelet transformations (Section 5.4.2) to the interpolated output. While this processing is considerably more computationally intensive than deriving the feature vector for variable–transient classification (Section 6.4), because this stage is run on the output of the previous stage, it only has to consider the fraction of alerts that have a high confidence of not being a recurring variable. This choice reflects a simple design philosophy that we have adopted for ANTARES: “do the least with the most, and the most with the least”—i.e., we apply expensive analyses only when the percentage of relevant alerts is low.

As we are largely comparing objects with multiband photometry from the OSC and SNPhotCC against each other, rather than against OGLE, we use all of the available observer frame bands that can be mapped to the *griz* as described in Section 4.4.1. Portions of the data were removed from the classification process due to selection cuts. For the SNPhotCC,

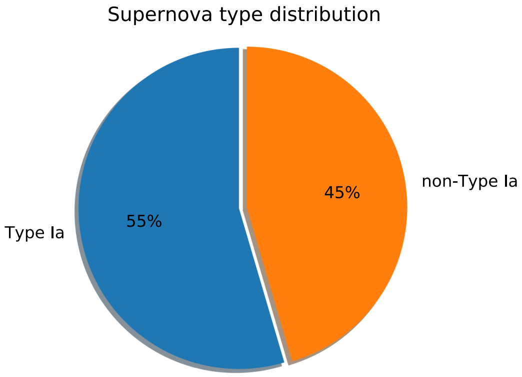

Figure 12. Distribution of SNe in the OSC after aggregation into binary Ia/non-Ia labels. This aggregation is necessary as many SNe subtypes are not distinct photometrically. The resulting data set is not strongly imbalanced.

if the objects did not have coverage in all four of the *griz* filters, the light curve was discarded. Out of the 17,133 light curves in the data set, 279 were eliminated by this selection cut. For the OSC, the criterion was relaxed, and only objects with observations in the *g*, *r*, and *i* filters were used for classification. Of the 3259 light curves, 2035 were eliminated by this selection cut. Using the same selection requirements as the SNPhotCC for the OSC would have resulted in a wholly unacceptable loss of 3118 light curves, leaving only 141 light curves for analysis. While we could relax this further to only require a single color, and thereby include more objects, this is not reflective of the data that LSST will produce, or the existing Pan-STARRS and the upcoming simulated PLAsTiCC data sets to which we wish to adapt this pipeline, defeating the goals of this work.

We reduced this stage to a binary classification system, using SNe Ia (including subtypes) as the positive class and non-Ia as the negative class. The non-Ia SNe included both Ib/c and II when classifying both the SNPhotCC and OSC data. This aggregation into binary labels is unfortunate but necessary given that the labels in the OSC data set (see Figure 2) are determined by spectroscopic indicators, and many of the subtypes are not substantially distinct photometrically (e.g., Ia-91T, Ia, Ia-91bg, Ia-02cx). Taming the SN zoo with aggregation, rather than with the more complex SMOTE approach, has the bonus of producing a relatively balanced data set (Figure 12), allowing us to employ random sampling with replacement.

6.5.1. Training and Cross-validation with Hyperparameter Optimization

All machine-learning algorithms have several hyperparameters that require optimization. For this work, many of the defaults for the random forest were considered to be appropriate for initial examination, with only the number of estimators taken as a potential variable. This is suitable for most of the classifiers, which only utilize a small number of features. However, our wavelet feature extraction stages produce a very large number of feature coefficients ($\mathcal{O}(10^2)$ for the SWT).

While we employ dimensionality reduction, either in the form of PCA or the BAGIDIS decomposition to reduce the number of coefficients, the wavelet transformation still results in an extremely high-dimensional feature space. Optimizing classifier performance (and therefore the purity of the extracted SN Ia sample) requires that we investigate how many and what features from this extremely high-dimensional space are useful. Therefore, we optimize the number of wavelet coefficients k and the maximum number of decision trees allowed for classification by the random forest, N, for this stage of the pipeline.

Table 4
The Optimized Hyperparameter of the Number of Wavelet Coefficients, k, used in Classification of the SNPhotCC (top) and OSC (Bottom) Data Sets

Wavelet Scheme	k	σ
(a) SNPhotCC		
Daubechies	112	9.6
Symlets	109	9.7
BAGIDIS	8	0.48
(b) Open Supernova Catalog		
Daubechies	66	16
Symlets	66	13.6
BAGIDIS	9	0.6

Note. These numbers are averages over five iterations to assess model stability, and the standard deviations of each value are on the right.

To assess model stability for the SN classification, we used fivefold nested cross-validation. Nested cross-validation allows for the optimization of hyperparameters combined with evaluation on a separate hold-out set. This ensures that the optimization step does not bias the classifier and allows for the entirety of the data to be used for training and testing once all iterations are completed.

The classifier optimization step is nested within the hyperparameter optimization. We ran a hyperparameter search for each cross-validation fold of the SN classifier. This optimization routine maximized the discrimination between the correctly and incorrectly predicted inputs from the classifier by varying the number of input components returned by the dimensionality reduction methods, k, listed above. The results of this randomized search are listed in Table 4.

The hyperparameter for the number of wavelet coefficients used in classification, k, was stable over the five iterations for all wavelet types. As expected by their design, the BAGIDIS decomposition coefficients required far fewer components for peak classification performance than the Daubechies or symlets coefficients. The SNPhotCC in general required a larger number of principal components when maximizing performance.

Neither the OSC nor the SNPhotCC depended strongly on the number of decision trees used in the forest, N, as long as the number was reasonably high ($N \geqslant 300$). This is in keeping with other works and our finding in Figure 11 that suggest that as long as the number of trees in a given random forest is above a certain threshold, the performance increases only slightly with more trees (Breiman 1999). We utilized a forest with 600 trees for analysis of all the different wavelet methods, for both the OSC and SNPhotCC data.

7. Machine-learning Classifier Performance

After training each stage of our pipeline, we evaluated its performance with cross-validation on test sets. We use several standard statistical quantities to assess the stages, and we briefly describe these evaluation metrics below.

7.1. Evaluation Metrics

We use three metrics for the evaluation of the classifier performance: (1) accuracy, (2) the receiver operating characteristic curve, and (3) the normalized confusion matrix. All three metrics have been used previously in the astronomical literature on machine learning (e.g., R11 and L16). We briefly describe the properties of these metrics below.

Most evaluation metrics are defined in terms of binary classification, with one class being considered "positive" and the other being "negative." The basic quantities used to describe the input sample are the number of positive and negative cases in the sample, P and N. The quantities used to describe the classified sample are the number of correctly classified (T) objects—the "True Positives," TP, and "True Negatives," TN. Their complement is the number of incorrectly classified (F) objects—the "False Positives," FP and "False Negatives," FN. These quantities can also be used in a multiclass scenario by taking a "One-versus-rest" approach, where one class of the sample with J classes is treated as positive, and all the others are treated as negatives.

7.1.1. Accuracy

The simplest metric that is used is the accuracy with which a trained classifier predicts the labels of a test set—the fraction of correctly predicted labels:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{P + N}. \tag{3}$$

Accuracy is a poor metric of evaluation if the classes are not evenly distributed, as is the case for the SNPhotCC data set (30% Type Ia). In such cases, several other metrics can be used to evaluate classifier performance while accounting for class imbalance, and these provide more useful measures of classifier performance in astrophysical contexts. We evaluate the receiver operating characteristic (ROC) curve and the area under the curve (AUC) in this work.

7.1.2. Receiver Operating Characteristic Curves

Receiver operating characteristic (ROC) curves visualize the trade-off between sample purity and sample completeness—i.e., the true positive rate (TPR) and the false positive rate (FPR) as a function of the threshold used for classification. The true positive rate, TPR, is the ratio of correctly classified positives to the total number of positives in the data set:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{P}, \tag{4}$$

and similarly, the false positive rate, FPR, is defined as

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} = \frac{\text{FP}}{N}. \tag{5}$$

The output classification probabilities from the classifier are on the unit interval. If the threshold for a positive classification is set to 0.8, the classifier will not classify the object as a positive until the probability exceeds 80%. By varying this threshold parameter, we can better assess the performance of the machine-learning algorithm. The ROC curve varies the threshold continuously, evaluating the TPR and FPR for each value of the threshold. This additionally provides a metric to determine what threshold is appropriate for a study. Studies that understand the impact of false positives, such as cosmological studies using photometric samples of SN Ia, can determine a threshold that provides the desired sample purity. We use the ROC curve to evaluate the performance of our wavelet-based SN Ia classifier for the OSC and SNPhotCC data sets in Section 7.2.3.

The goal for classification is to maximize the TPR while simultaneously minimizing the FPR. A useful single-number metric that can be extracted from an ROC curve is the fractional area under the ROC curve (AUC). If the AUC is equal to 0.5, then the TPR equals the FPR for all thresholds, indicating random classification, i.e., an uninformative "guess." An AUC of 1 indicates that the TPR is maximized at all thresholds and the FPR is minimized, representing perfect classification.

7.1.3. The Confusion Matrix

While the ROC curve is a useful metric to study classifier performance in a "One-versus-rest" context, it is often useful to know what the false positives are—i.e., which astrophysical sources are the sources of contamination for the class of interest. The normalized confusion matrix evaluates the fraction of each input class as a function of each output class:

$$c_{ij} = \frac{N_{ij}}{P_i}, \tag{6}$$

where N_{ij} is the number of elements of class i labelled as class j. Where a visualization such as the t-SNE (Section 5.3.1) can provide a *qualitative* assessment of what the sources of contamination will be prior to machine learning, the confusion matrix provides the quantitative assessment after the classifier has been developed. Both approaches have merit: the first aids in the design of the feature encoding and the classifier, while the second provides a metric that can be used to compare different classifiers against each other. We use the confusion matrix to evaluate the performance of our variable and transient classification for the combined OGLE and OSC data set in Section 7.2.2.

7.2. Quantified Classifier Performance

In the following sections, we measure the various evaluation metrics for each stage of our pipeline and compare to the literature wherever possible. While we discuss the performance of each stage separately, brokers will structure several machine-learning algorithms into a pipeline, and the stages cannot be considered independently of each other. Simulated data sets such as the upcoming PLAsTiCC will be essential to evaluate broker performance.

7.2.1. The Variable–Transient Split Using Timescale Characterization

As discussed in Sections 6.4.2.1 and 6.4.2.2, real astrophysical data sets have complex selection effects that result in imbalanced class representation. Even in volume-limited surveys, the fundamental differences in event rates can create an almost order of magnitude imbalance, e.g., between recurring variables and transients. The ubiquitous class imbalance makes the simple accuracy score a poor metric for evaluation. We therefore construct the ROC curve for this stage and evaluate the AUC. The periodic versus nonperiodic classification performed well, with a consistent AUC of 0.99 when run over fivefold cross-validation. Only 8 per 40,000 objects are misclassified.

LSST can expect this level of performance as early as a few months after operation. Most variables will be detectable in multiple passbands, and the survey will rapidly establish a baseline of variability for all sources, whereas transients will typically only have nondetections before explosion. Effectively then, this stage distinguishes between variables and transients using the output of a multiband Fourier transform, as the former are continuous signals while the latter are wave packets. In addition, we can expect external catalog information from surveys such as *Gaia*, Pan-STARRS, and SkyMapper to all help distinguish between variable and transient sources.

The binary output assigned by this stage does not give us an indication of which classes of astrophysical variables are being erroneously labelled as transients. However, we can get a sense for this from the next stage in the pipeline, discussed below, which attempts a broad classification of variables and transients using subclasses of major groups of variables.

7.2.2. The Classification of the OGLE and OSC Sample Using Timescale and Magnitude Features

For nonbinary classification problems, the ROC is not a curve but a hyper-surface of every class against every other class, and is not easily interpretable. ROC curves can still be constructed by adopting a "One-versus-Rest" scheme, where one classifier is trained per class in the sample. However, this approach addresses a different question than originally posed, providing not multiclass classification but a vector of noncomparable scores assessing the likelihood of a source being a member of each class, i.e., multilabel categorization. We therefore report the full normalized confusion matrix for the variable–transient classification stage, shown in Figure 13.

The overall accuracy of the classifier is 0.96. Despite the imbalanced data set, all but one class in the set has an accuracy above 0.90, with most above 0.95. This indicates that the classifier is not overfitting the majority class, and the resampling approach we adopted in training (see Section 6.4.2.2) has been effective. The notable exception is the class of double-periodic variables. This category is ill-defined and contains several different astrophysical classes, all of which exhibit at least two strong periods. Additionally, there are only 136 total members of this class, compared to $\mathcal{O}(10^3 - 10^6)$ members in the other classes. The heterogeneity of the instances of this class, and their drastic underrepresentation, conspire to make any resampling scheme ineffective.

Nevertheless, such drastically underrepresented classes can be very interesting scientifically, e.g., the electromagnetic counterparts of gravitational wave sources such as GW 170817 (Coulter et al. 2017; Abbott et al. 2017). Alert-brokers will need to develop different strategies to identify these extremely rare events in the data. It may be possible to use simulations to populate classes that are very rare, use techniques such as isolation forests to identify large outliers in the feature space, or develop bespoke filtering stages that include contextual information for these rare sources. But the overall performance of this stage augurs well for LSST, which will have more than the two passbands used for this study, and will be much more homogeneous and better calibrated. Multiband time evolution with a cadence comparable to the characteristic timescales for many different kinds of sources will allow us to employ features that are better at discriminating between the different astrophysical classes and their subtypes.

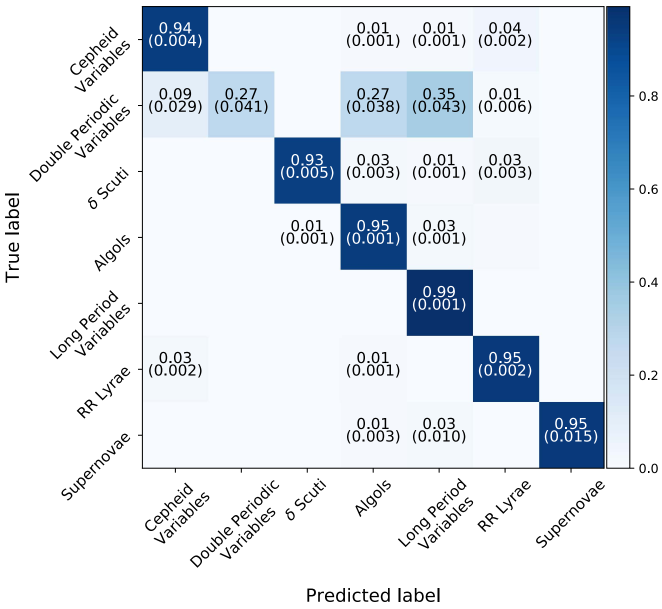

Figure 13. Normalized confusion matrix (c_{ij}) of the variable and transient classifier applied to the ninefold cross-validated test sets. We report the mean fraction of each of the true classes i classified as a member of class j. The standard deviation of the confusion matrix elements is reported in parentheses. Elements without a numerical score have $<1\%$ of the total class population. The only label on which the classifier performs poorly is the ill-defined, underrepresented class of double-periodic variables.

7.2.3. The Classification of Supernovae Using Wavelet Transformations of Light Curves

SN classification is the most complex stage of the pipeline. We are attempting to discriminate between the most similar light curves of the data set to provide a high-purity feed to a cosmological experiment that is susceptible to bias arising from sample contamination. To accomplish this goal, we are using a large and difficult-to-compute feature vector together with the most carefully optimized machine-learning stage of our pipeline. Furthermore, as this stage is applied retrospectively, no additional observations can be obtained to improve the accuracy of the classifier. Figure 14 shows ROC curves for the OSC and SNPhotCC data sets for all wavelet classes and levels tested.

7.2.3.1. Results for the SNPhotCC

The AUC scores from the SNPhotCC show promising results, with values of 0.97–0.98 consistently for both the Daubechies and symlets features, and 0.98 for the BAGIDIS features. The SNPhotCC data set has been used for testing classification schemes by many teams over the past seven years, and as a result there are many sources of comparison. The most directly comparable are those from Lochner et al. (2016), as they used the AUC as their performance metric, as well as a similar classification scheme. Using Boosted Decision Trees, they achieved an AUC of 0.98 with symlets. An AUC of 0.98 reflects excellent classification performance on the part of our broker.

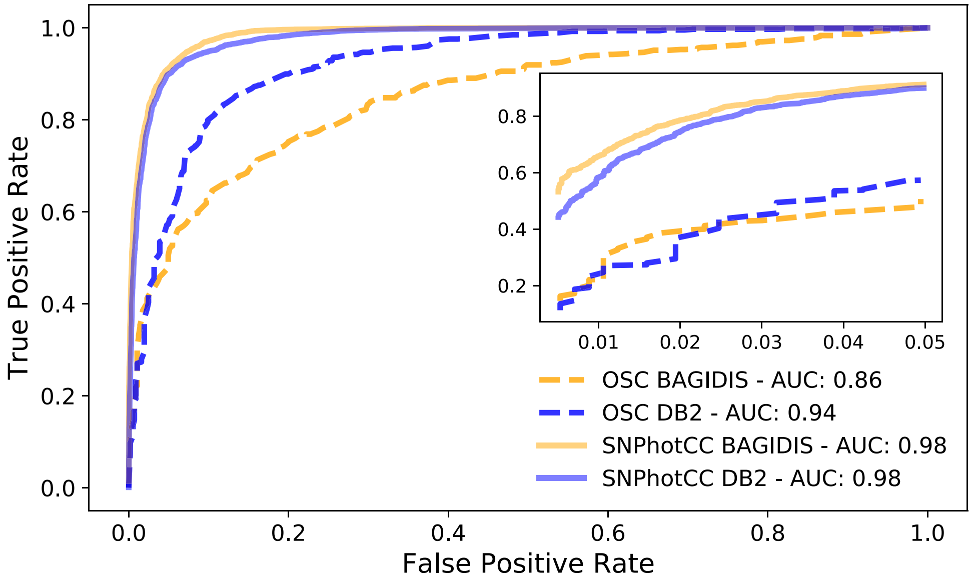

Figure 14. ROC curves for the SNPhotCC and OSC data sets. The classifier performance drops on the OSC relative to the SNPhotCC as indicated by the AUC score in the legend. This is illustrated in the inset, where the ROC curves are shown over a very narrow range in the FPR, necessary for a high-purity sample of SNe Ia.

7.2.3.2. Results for the OSC

This is the only work to date that has attempted to classify the heterogeneous and diverse set of all SNe publicly available through the OSC. There are no other figures of merit from other works against which to compare these results. Consequently, we compare the AUCs for the different wavelet-decomposition methods against each other.

The BAGIDIS features perform significantly worse than both the Daubechies and the symlets, with an average AUC of 0.85, and particularly poorly compared to the SNPhotCC result. This is likely because the simulated SNPhotCC data are all generated from the same set of underlying models with the same survey characteristics and are much more homogeneous than the OSC data. The heterogeneity of the OSC data introduces noise into the BAGIDIS features, consequently decreasing classifier performance. For the OSC, there is no significant difference between the symlets and Daubechies wavelet results, both reporting high AUCs of 0.94.

8. Discussion and Concluding Remarks

The ANTARES project described in this paper is a fully automated implementation of a machine-learning pipeline, forming the core of an alert-broker.

This work serves two purposes. The first is to provide illustrative examples of the development and functioning of an alert-broker: what issues must be considered, what challenges arise, what techniques exist to address to mitigate them, what open-source packages exist, and what level of performance is possible at present.

The second purpose of this work is to develop a pipeline to process existing and upcoming data sets (such as Pan-STARRS and ZTF) through the ANTARES pipeline in order to implement iterated semi-supervised learning. In an ideal world, we would already have a readily available representative labelled training set with which to develop an alert-broker for LSST and could simply wait for the survey to begin observations. Unfortunately, no such training set exists, and broker development is an open-ended research question. We developed this pipeline recognizing that it will change continuously as we prepare for LSST, with the addition of more astrophysical classes, feature characterization, classification stages, and the incorporation of contextual information.

8.1. Real Data Sets as Tests of Broker Performance

We have examined the effectiveness of machine-learning algorithms as stages of an alert-broker for LSST. We developed three different stages, each addressing a different question from what a generic alert-broker will have to answer—variable–transient separation, variable and transient classification, and high-purity sample selection.

To train and test these stages, we assembled a very large labelled data set of real variable and transient objects, including observations from a wide variety of surveys in at least two passbands. The heterogeneity and class imbalance of this data set required that we impose additional constraints in order to ensure that our machine-learning pipeline was making decisions based on astrophysical properties rather than on characteristics of the parent surveys. These constraints are evident in the design of our feature extraction and our choice to exclude contextual information.

Even when processing real alerts, contextual information cannot be guaranteed; the southern sky has less extant coverage, and LSST will discover sources that are more faint than any in existing surveys. Nor will such contextual information always add useful information. Further, alert-brokers will need to test their performance on large simulated data sets of photometric time series (e.g., PLAsTiCC) that lack such information. Simulations that do choose to include contextual information will likely only reflect our present ignorance about many of the correlations between astrophysical sources and their environment. Indeed, LSST itself will perform many of the population studies that will shed light on these relationships.

Despite these limitations, working with real data sets has several advantages. The extreme class imbalance of our data set presents a much more realistic challenge than more homogeneous training sets (real or simulated), which only represent a small subset of the astrophysical sources that LSST can expect to discover. Additionally, real data have several pathologies (outlying data, gaps due to bad weather, calibration errors) that are often not reflected in simulations.

The heterogeneity of our data set forced us to adopt more advanced feature-extraction schemes and to carefully consider how we trained our machine-learning pipeline in order to avoid bias. This data set can be used to benchmark the algorithms, exposing previously unconsidered failure modes and setting an effective lower bound for machine-learning performance. Each of the stages we designed for this work has benefited from the use of a real labelled data set, especially because we can address specific use cases for an alert-broker such as ANTARES, as outlined in Section 1.1. We expect this pipeline to be even more effective on homogeneous data sets. However, the real benefit of improved data sets is that we improve the stages themselves further. Below, we discuss how each of the stages in this pipeline might evolve as we enter the era of LSST.

8.2. Performance of the Variable–Transient Categorization

With only two bands, the variable versus transient classifier achieves an AUC of 0.99. This is partly due to the inclusion of the timescale uncertainty and false-alarm probability along with the characteristic timescale as features (as these features are sensitive to the shape of the signal), as well as the use of timescale information from two independent passbands, which are strongly correlated in the case of recurring variables.

With the addition of contextual information from surveys such as *Gaia* and filtering stages such as the VPDF, we will be able to improve variable–transient separation further, allowing us to perform categorization with fewer observations. After a year of LSST survey operations, we will have a long baseline for recurrent variables, effectively guaranteeing nearly perfect categorization with only a few observations of a new alert source. This will enable follow-up studies that probe the early-time evolution of several classes of astrophysical transients, giving us a window into the physics of their progenitor systems. We can further improve this capacity by including the periodogram vector itself as a feature. This will allow source classification, rather than just categorization. Together with contextual information, such a stage will be useful to rapidly identify new transients in the alert stream that are likely members of a particular astrophysical class, enabling prioritized fast-turnaround follow-up.

We can also examine the use of neural networks and deep-learning techniques to identify relevant features from image data, such as postage-stamp cutouts included with the alerts, rather than relying solely on time-series data. This will allow the extraction of the maximum amount of information from each LSST alert packet.

8.3. Performance of the Variable and Transient Classifier

Broad classification of the alert stream is the core function of any alert-broker, and indeed can be considered synonymous with brokering itself. To accomplish this task, we extracted informative astrophysical features from our heterogeneous data set, and we developed effective techniques to address the extreme class imbalance. This allowed us to train a random forest that could successfully distinguish several broad astrophysical classes with an average accuracy of ∼96%.

However, the limitations of the data set meant that neither the classes represented nor the features derived were complete. Adding other classes (e.g., AGN) would require adding other surveys as well as other features. These features could be used to distinguish between the surveys, leading to the development of biased classifiers that cannot be adapted to new surveys.

Further development of this stage will benefit the most from a new homogeneous data set, such as light curves from the Pan-STARRS Medium Deep Survey. However, this data set is not labelled, and we will need to use this pipeline to begin to classify the PS1 light curves with the goal of retraining our existing classifiers and developing new stages.

Another interesting avenue for development is joint human–machine classification, where a subset of objects are provided to the public using citizen science portals such as Zooniverse.[36] Human-vetted objects can be used to validate the output of the brokers, as well as improve classification performance on edge cases, where visual inspection may identify features that have not been considered previously.

8.4. Performance of the SN Ia Versus Non-SN Ia Classifier

For the SN stage, we explored a nonparametric approach to the problem of photometric SN classification. The peak performance for the OSC was 0.94 AUC, based on the approach developed by L16, and 0.98 AUC for the SNPhotCC. These numbers represent high-quality classification, and the AUC of 0.98 is at the same level as L16's performance when using wavelet features. It outperforms or is very competitive with other SN classification algorithms that have been benchmarked with the SNPhotCC data set.

The BAGIDIS encoding of the light curve is much smaller (almost a factor of two) than the SWT coefficients, yet both methods perform equally well on the SNPhotCC data. Classification is less effective on the real OSC light curves than on the simulated SNPhotCC, with the performance of BAGIDIS dropping significantly. The Unbalanced-Haar transform underlying BAGIDIS is more sensitive to sharp changes because of the non-differentiability of the Haar wavelet. The relative lack of stability (and thus more noisy sharp changes) for the OSC Gaussian process fits as compared to that of the SNPhotCC implies that the Haar wavelet coefficients are a less reliable way to capture information from the OSC light curves. This likely reflects the heterogeneity of the OSC data set, which contains photometry from several different telescopes and instruments, calibrated with varying degrees of photometric precision, spanning three decades. This outperformance of both wavelet methods on the SNPhotCC data set relative to the OSC data set bodes well for analysis of future LSST light curves. The data stream from LSST will have consistent photometry, and ultimately be more similar to the SNPhotCC data set than to the heterogeneous OSC. The AUC value of 0.98 for both wavelet schemes is very competitive and highlights the success of the nonparametric approach to light-curve classification.

The biggest improvements to this stage will be the addition of more independent feature extraction methods and more astrophysical classes in the training. All classification with a large number of observations is retrospective, and there are few additional observations that can be obtained that will add useful new information. Consequently, when considering the full time evolution and contextual information for sources, brokers must

[36] https://www.zooniverse.org/

be able to extract extremely high-purity samples for any well-defined astrophysical class. Brokers that can surmount this high bar will be able to produce "living" catalogs of the variable and transient sky, extracting the maximum gain from the alert stream.

9. Future Work

The fundamental challenge for ANTARES and all alert-brokers is that no present survey can generate even a limited number of alerts with similar properties as LSST. The extant labelled data sets suitable for supervised learning algorithms are small and drawn from a mixture of different surveys. Simulated data sets, while useful, often do not accurately reflect the pathologies of real data. Many correlations that are present in real data, such as those between transients and their host environment, or variables and their location in the galaxy, are simply not captured by simulations.

While unlabelled data exist from sources such as PS1 Medium Deep Survey data and DES, much of the data to classify them effectively, such as publicly available detection tables, do not exist. The most promising approach then is semi-supervised learning, wherein a smaller training set is used to classify a fraction of a larger unlabelled set, which is then in turn used to retrain the classifier. We plan to perform a retrospective classification of light curves from the Pan-STARRS PS1 medium deep survey, which we will treat as a semi-supervised learning problem. We also welcome the contribution of models or data from the astronomical community, and we are keen to develop new stages that meet the needs of the many groups involved in time-domain science.

One of the most pressing developments for ANTARES will be adding a "None of the above" label and consistent criteria for a source to meet to be flagged as such. Identifying and potentially clustering outliers that do not belong to any previously known astrophysical class are the core focus of ANTARES. The stages that result from these studies will be tested extensively by the upcoming PLAsTiCC data set.

Significant work will also be devoted to the front-end interface for ANTARES, which we will begin to test publicly. We will begin running on various live alert streams from surveys, including the ASAS-SN and upcoming ZTF public survey alerts. These studies in particular are essential for developing and battle-testing ANTARES before the commencement of the LSST survey.

G.N. is supported by the Lasker Fellowship at the Space Telescope Science Institute. T.Z. began working on ANTARES under the NOAO Summer REU program (2015, adviser: GN). T.Z. elected to continue working with G.N. on his senior honors thesis on supernova light curve classification at Macalester College (earned 2017 April). Material from T.Z.'s thesis is presented in this work. M.L. acknowledges support from the SKA SA, the NRF, and AIMS. ANTARES is generously supported by NSF INSPIRE grant (CISE AST-1344204, PI: R. Snodgrass) as well as funding to develop the NOAO Community Science Data Center (CSDC).

This work makes extensive use of the OGLE catalog and the Open Supernova Catalog, and we are grateful to both projects for maintaining open-access portals to data. We are very grateful to James Guillochon for his help with ingesting the OSC data into ANTARES. This work makes extensive use of open-source software and high-performance computing resources, and would not be possible without the diligent efforts of the system administrators of the ANTARES cluster, Adam Scott and Pete Wargo, as well as the IT staff at NOAO, STScI, and the University of Arizona. The authors thank Amali Vaz for her careful reading and useful suggestions. We are grateful to the anonymous referee for detailed feedback, which has helped refine this work. This work made extensive use of several open-source packages listed in the software section.

Software: numpy, scipy, matplotlib, astropy, pandas, pyMySQL, H5py, pyFFTW, gatspy, george, PyWavelets, imbalanced-learn, scikit-learn, SLURM, MPI4py, the conda package and environment management systems.

ORCID iDs

Gautham Narayan https://orcid.org/0000-0001-6022-0484
Michelle Lochner https://orcid.org/0000-0003-2221-8281
Thomas Matheson https://orcid.org/0000-0001-6685-0479
Abhijit Saha https://orcid.org/0000-0002-6839-4881
Tim Axelrod https://orcid.org/0000-0002-5722-7199
Robert S. Maier https://orcid.org/0000-0002-1259-1341
Stephen T. Ridgway https://orcid.org/0000-0003-2557-7132

References

Abbott, B. P., Abbott, R., Abbotto, T. D., et al. 2017, ApJL, 848, L12
Abolfathi, B., Aguado, D. S., Aguilar, G., et al. 2017, arXiv:1707.09322
Ambikasaran, S., Foreman-Mackey, D., Greengard, L., Hogg, D. W., & O'Neil, M. 2015, ITPAM, 38, arXiv:1403.6015
Arcavi, I., Howell, D. A., Kasen, D., et al. 2017, Natur, 551, 210
Baluev, R. V. 2008, MNRAS, 385, 1279
Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. 2004, SIGKDD Explor. Newsl., 6, 20
Betoule, M., Marriner, J., Regnault, N., et al. 2013, A&A, 552, A124
Bloom, J. S., Richards, J. W., Nugent, P. E., et al. 2012, PASP, 124, 1175
Bowyer, K. W., Chawla, N. V., Hall, L. O., & Kegelmeyer, W. P. 2011, CoRR, abs/1106.1813
Breiman, L. 1999, UC Berkeley TR5, 67
Brink, H., Richards, J. W., Poznanski, D., et al. 2013, MNRAS, 435, 1047
Brown, P. J., Holland, S. T., Immler, S., et al. 2009, AJ, 137, 4517
Bylander, T. 2002, Machine Learning, 48, 287
Cabrera-Vives, G., Reyes, I., Förster, F., Estévez, P. A., & Maureira, J.-C. 2017, ApJ, 836, 97
Carrasco, D., Barrientos, L. F., Pichara, K., et al. 2015, A&A, 584, A44
Coulter, D. A., Foley, R. J., Kilpatrick, C. D., et al. 2017, Sci, 358, 1556
Dark Energy Survey Collaboration, Abbott, T., Abdalla, F. B., et al. 2016, MNRAS, 460, 1270
Debosscher, J., Sarro, L. M., Aerts, C., et al. 2007, A&A, 475, 1159
Debosscher, J., Sarro, L. M., López, M., et al. 2009, A&A, 506, 519
Dieleman, S., Willett, K. W., & Dambre, J. 2015, MNRAS, 450, 1441
Djorgovski, S. G., Mahabal, A. A., Donalek, C., et al. 2014, arXiv:1407.3502
du Buisson, L., Sivanandam, N., Bassett, B. A., & Smith, M. 2015, MNRAS, 454, 2026
Dubath, P., Rimoldini, L., Süveges, M., et al. 2011, MNRAS, 414, 2602
Eyer, L., & Blake, C. 2005, MNRAS, 358, 30
Folatelli, G., Phillips, M. M., Burns, C. R., et al. 2010, AJ, 139, 120
Foley, R. J., & Mandel, K. 2013, ApJ, 778, 167
Friedman, A. S., Wood-Vasey, W. M., Marion, G. H., et al. 2015, ApJS, 220, 9
Frigo, M., & Johnson, S. G. 2012, FFTW: Fastest Fourier Transform in the West, Astrophysics Source Code Library, ascl:1201.015
Fryzlewicz, P. 2007, J. Am. Stat. Assoc., 102, 1318
Gaia Collaboration, Prusti, T., de Bruijne, J. H. J., et al. 2016, A&A, 595, A1
Ganeshalingam, M., Li, W., Filippenko, A. V., et al. 2010, ApJS, 190, 418
Goldstein, D. A., D'Andrea, C. B., Fischer, J. A., et al. 2015, AJ, 150, 82
Graham, M. J., Drake, A. J., Djorgovski, S. G., et al. 2013a, MNRAS, 434, 3423
Graham, M. J., Drake, A. J., Djorgovski, S. G., Mahabal, A. A., & Donalek, C. 2013b, MNRAS, 434, 2629
Gregory, P. C., & Loredo, T. J. 1992, ApJ, 398, 146
Guillochon, J., Parrent, J., Kelley, L. Z., & Margutti, R. 2017, ApJ, 835, 64

Guy, J., Sullivan, M., Conley, A., et al. 2010, A&A, 523, A7
Hicken, M., Challis, P., Kirshner, R. P., et al. 2012, ApJS, 200, 12
Holtzman, J. A., Marriner, J., Kessler, R., et al. 2008, AJ, 136, 2306
Ivezić, v., Tyson, J. A., Acosta, E., et al. 2008, arXiv:0805.2366v4
Jacobs, C., Glazebrook, K., Collett, T., More, A., & McCarthy, C. 2017, MNRAS, 471, 167
James, G. 2013, in An Introduction to Statistical Learning: With Applications in R (New York: Springer), 320
Jones, D. O., Scolnic, D. M., Riess, A. G., et al. 2017, ApJ, 843, 6
Karpenka, N. V., Feroz, F., & Hobson, M. P. 2013, MNRAS, 429, 1278
Kessler, R., Bassett, B., Belov, P., et al. 2010a, PASP, 122, 1415
Kessler, R., Conley, A., Jha, S., & Kuhlmann, S. 2010b, arXiv:1001.5210
Kessy, A., Lewin, A., & Strimmer, K. 2017, The American Statistician, 0, 0
Khazov, D., Yaron, O., Gal-Yam, A., et al. 2016, ApJ, 818, 3
Kim, D.-W., Protopapas, P., Bailer-Jones, C. A. L., et al. 2014, A&A, 566, A43
Klencki, J., Wyrzykowski, Ł., Kostrzewa-Rutkowska, Z., & Udalski, A. 2016, AcA, 66, 15
Lafler, J., & Kinman, T. D. 1965, ApJS, 11, 216
Law, N. M., Kulkarni, S. R., Dekany, R. G., et al. 2009, PASP, 121, 1395
Lemaître, G., Nogueira, F., & Aridas, C. K. 2017, Journal of Machine Learning Research, 18, 1
Lochner, M., McEwen, J. D., Peiris, H. V., Lahav, O., & Winter, M. K. 2016, ApJS, 225, 31
Lomb, N. R. 1976, Ap&SS, 39, 447
MacKay, D. J. C. 2003, Information Theory, Inference and Learning Algorithms (Cambridge: Cambridge Univ. Press)
Mahabal, A., Sheth, K., Gieseke, F., et al. 2017, CoRR, arXiv:1709.06257
Mahabal, A. A., Djorgovski, S. G., Drake, A. J., et al. 2011, BASI, 39, 387
Masci, F. J., Hoffman, D. I., Grillmair, C. J., & Cutri, R. M. 2014, AJ, 148, 21
Masci, F. J., Laher, R. R., Rebbapragada, U. D., et al. 2017, PASP, 129, 014002
Mislis, D., Bachelet, E., Alsubai, K. A., Bramich, D. M., & Parley, N. 2016, MNRAS, 455, 626
Morii, M., Ikeda, S., Tominaga, N., et al. 2016, PASJ, 68, 104
Narayan, G., Rest, A., Tucker, B. E., et al. 2016, ApJS, 224, 3
Nguyen, H. M., Cooper, E. W., & Kamei, K. 2011, Int.J. Knowl. Eng. Soft Data Paradigm., 3, 4
Ofek, E. O., Laher, R., Law, N., et al. 2012, PASP, 124, 62
Oke, J. B., & Sandage, A. 1968, ApJ, 154, 21
Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, Journal of Machine Learning Research, 12, 2825, http://dl.acm.org/citation.cfm?id=1953048.2078195
Poznanski, D., Maoz, D., & Gal-Yam, A. 2007, AJ, 134, 1285
Rau, A., Kulkarni, S. R., Law, N. M., et al. 2009, PASP, 121, 1334
Richards, J. W., Starr, D. L., Brink, H., et al. 2012, ApJ, 744, 192
Richards, J. W., Starr, D. L., Butler, N. R., et al. 2011, ApJ, 733, 10
Ridgway, S. T., Matheson, T., Mighell, K. J., Olsen, K. A., & Howell, S. B. 2014, ApJ, 796, 53
Robin, A. C., Reylé, C., Derrière, S., & Picaud, S. 2003, A&A, 409, 523
Saha, A., Matheson, T., Snodgrass, R., et al. 2014, Proc. SPIE, 9149, 914908
Saha, A., & Vivas, A. K. 2017, AJ, 154, 231
Saha, A., Wang, Z., Matheson, T., et al. 2016, Proc. SPIE, 9910, 99100F
Sarro, L. M., Debosscher, J., López, M., & Aerts, C. 2009, A&A, 494, 739
Scargle, J. D. 1981, ApJS, 45, 1
Schwarzenberg-Czerny, A. 1996, ApJL, 460, L107
Scolnic, D., Casertano, S., Riess, A., et al. 2015, ApJ, 815, 117
Seaman, R., Williams, R., Allan, A., et al. 2011, arXiv:1110.0523
Smith, R. M., Dekany, R. G., Bebek, C., et al. 2014, Proc. SPIE, 9147, 914779
Soszynski, I., Poleski, R., Udalski, A., et al. 2008, AcA, 58, 163
Staley, T. D., & Fender, R. 2016, arXiv:1606.03735
Stellingwerf, R. F. 1978, ApJ, 224, 953
Stetson, P. B. 1996, PASP, 108, 851
Stritzinger, M. D., Phillips, M. M., Boldt, L. N., et al. 2011, AJ, 142, 156
Sullivan, M., Guy, J., Conley, A., et al. 2011, ApJ, 737, 102
Timmermans, C., & von Sachs, R. 2010, https://dial.uclouvain.be/pr/boreal/object/boreal:91090
Udalski, A., Szymanski, M., Kaluzny, J., Kubiak, M., & Mateo, M. 1992, AcA, 42, 253
Ulyanov, D. 2016, Multicore-TSNE, https://github.com/DmitryUlyanov/Multicore-TSNE, GitHub
van der Maaten, L. 2014, J. Mach. Learn. Res., 15, 3221
van der Maaten, L., & Hinton, G. 2008, J. Mach. Learn. Res., 9, 2579
VanderPlas, J. 2016, Gatspy: General Tools for Astronomical Time Series in Python, Astrophysics Source Code Library, ascl:1610.007
VanderPlas, J. T., & Ivezić, Ž. 2015, ApJ, 812, 18
Varughese, M. M., von Sachs, R., Stephanou, M., & Bassett, B. A. 2015, MNRAS, 453, 2848
Wattenberg, M., Viégas, F., & Johnson, I. 2016, Distill, 00002
Williams, R. D., Djorgovski, S. G., Drake, A. J., Graham, M. J., & Mahabal, A. 2009, in ASP Conf. Ser. 411, Astronomical Data Analysis Software and Systems XVIII, ed. D. A. Bohlender, D. Durand, & P. Dowler (San Francisco, CA: ASP), 115
Wright, D. E., Smartt, S. J., Smith, K. W., et al. 2015, MNRAS, 449, 451