

Machine Learning on Difference Image Analysis: A comparison of methods for transient detection

B. Sánchez^{a,f,*}, M. J. Domínguez R.^{a,f}, M. Lares^{a,f}, M. Beroiz^{d,e,1}, J. B. Cabral^{1a,b}, S. Gurovich^a, C. Quiñones^f, R. Artola^f, C. Colazo^a, M. Schneider^a, C. Girardini^a, M. Tornatore^a, J. L. Nilo Castellón^{g,h}, D. García Lambas^{a,f}, M. C. Díaz^{c,1}

^a Instituto De Astronomía Teórica y Experimental (IATE-CONICET), Laprida 854, X5000BGR, Córdoba, Argentina.

^b Facultad de Ciencias Exactas, Ingeniería y Agrimensura, UNR, Pellegrini 250 - S2000BTP, Rosario, Argentina.

^c Center for Gravitational Wave Astronomy, University of Texas Rio Grande Valley, Brownsville, TX, USA.

^d University of Texas Rio Grande Valley (UTRGV), One West University Blvd. Brownsville, Texas 78520, USA.

^e University of Texas at San Antonio (UTSA), 1 UTSA Circle, San Antonio, TX 78249, USA.

^f Observatorio Astronómico de Córdoba, Universidad Nacional de Córdoba (OAC-UNC), Laprida 854, X5000BGR, Córdoba, Argentina.

^g Departamento de Física y Astronomía, Facultad de Ciencias, Universidad de La Serena. Av. Juan Cisternas 1200, La Serena, Chile.

^h Instituto de Investigación Multidisciplinario en Ciencia y Tecnología, Universidad de La Serena. Benavente 980, La Serena, Chile.

arXiv:1812.10518v2 [astro-ph.IM] 8 Aug 2019

Abstract

We present a comparison of several Difference Image Analysis (DIA) techniques, in combination with Machine Learning (ML) algorithms, applied to the identification of optical transients associated to gravitational wave events. Each technique is assessed based on the scoring metrics of Precision, Recall, and their harmonic mean $F1$, measured on the DIA results as standalone techniques, and also in the results after the application of ML algorithms, on transient source injections over simulated and real data. This simulations cover a wide range of instrumental configurations, as well as a variety of scenarios of observation conditions, by exploring a multi dimensional set of relevant parameters, allowing us to extract general conclusions related to the identification of transient astrophysical events.

The newest subtraction techniques, and particularly the methodology published in Zackay et al. (2016) are implemented on an Open Source Python package, named *properimage*, suitable for many other astronomical image analyses. This together, with the ML libraries we describe, provides an effective transient detection software pipeline. Here we study the effects of the different ML techniques, and the relative feature importances for classification of transient candidates, and propose an optimal combined strategy. This constitutes the basic elements of pipelines that could be applied in searches of electromagnetic counterparts to GW sources.

Keywords: methods: data analysis, techniques: image processing

1. Motivations: Synoptic era scenario

Synoptic sky surveys are promoters of a new era of observational astronomy, where data volume is becoming a major challenge, and discoveries are happening at a rate never experienced before. Several collaborations, involved in observational astrophysics projects, are pushing towards a *data-driven* science paradigm, and transforming astronomy. The Large Synoptic Survey Telescope (LSST, Ivezić and for the LSST Collaboration, 2008; LSST Science Collaboration et al., 2009) is going to bring this phenomenon to a higher level, where raw data disk-space consumption is going to be in the PetaByte-scales by the end of the project. To face this transformation, astronomers have been involved in information technology development for several decades, bringing to existence organizations such as IVOA¹, an international alliance committed to organize and make available a living archive of historical astrophysical data.

In this context, a new era of observational astronomy is arising since the first direct detection of Gravitational Waves (GW, Abbott and Collaboration, 2016). This historical discovery places a huge responsibility on synoptic telescopes: the search for the electromagnetic (EM) counterparts of these GW events. The Transient Optical Robotic Observatory of the South (TOROS²), is a project aimed at identifying those GW sources. During *Advanced LIGO science run O1* TOROS participated in the search for an optical counterpart to the first GW detection (Díaz and Collaboration, 2016), in an effort to determine the origin of its progenitor. The theoretical scenario was developed in several articles, such as Kasen et al. (2013); Barnes et al. (2016), where models predict that a GW event like GW150914 involves a merger between compact objects. This model has three possible cases, featuring binary combinations of Black Hole and Neutron Stars components. In case a Neutron Star is one of these, the merger will produce an EM emission (or *Kilonova*) that will last a couple of days, and will be visible at optical and near-infrared wavelengths. The search for such an

*Corresponding author

Email address: bruno@oac.unc.edu.ar (B. Sánchez)

¹<http://ivoa.net/>

²<https://toros.utrgv.edu>

elusive signature is a major challenge, and in many ways can be described as a race against time. This objective was reached recently, when the event GW170817 was identified by several collaborations as the first observed Kilonova, see for example Abbott et al. (2017a); Díaz et al. (2017) and references therein.

One difficulty involved is the need for a comparison method between images obtained on different epochs, since a fast detection of small variations in brightness, over a large region of the sky is critical to identify the signature of a Kilonova event. Another issue is the requirement of a wide field of view, since the instrument dedicated to the search would need to cover several hundred square degrees per night, and also reaching deep magnitude limits. If we combine these two simple conditions of wide sky coverage and temporal resolution we have that the image comparison method will need to deal with variable Point Spread Functions (PSF) across several square degrees and, at the same time, be able to detect the magnitude variations with high fidelity. One of the approaches to this task is to compare the detected sources, and their brightnesses, on each epoch, and to select as possible transient candidates any mis-match. Though this methodology can find difficulties when applied on crowded stellar fields, or when the transient event is buried in a galaxy, and its flux is entangled with the luminosity profile of the host, (this was particularly the case for the event of GW170817). In the former case the angular cross match of sources can have a computational cost which would introduce an undesired time overhead for a transient discovery survey with high cadence. In the latter case in order to measure the flux and position of the transient source a correct modeling of the host galaxy luminosity profile must be applied beforehand.

There are several works that tackle this problem by using an image subtraction methodology, such as Alard and Lupton (1998); Bramich (2008). This image differentiation approach avoids the discussed issues of catalog cross-matching. A different approach has been taken by Zackay et al. (2016) in a series of three papers that derive an improved treatment of astronomical images from an statistical point of view. This is more general than previous works, and translates several astronomical common methods such as source detection into statistical language. The authors of this method claim that it also reduces the necessity of a posterior Machine Learning (ML) analysis. The astronomical community has been increasing the implementation of ML, seizing its capability for solving data processing issues based on handcrafted examples (Fortson et al., 2012), specially in the transient detection area (Djorgovski et al., 2010; Law et al., 2009; Rau et al., 2009).

In this work we implement three difference image analysis techniques: Alard & Lupton’s (from now on A), Bramich’s (B), and Zackay’s methodology in two separated ways (Z and S) to simulated and real data. Afterwards we train ML algorithms to identify interesting targets buried in a sea of *bogus* detections, with as extreme ratios as 1% or less. The aim of this article is to develop and establish the best possible combination of difference imaging and machine learning techniques based on a comprehensive metric. Novel promising methods such as those based on *Convolutional Neural Networks* (Sedaghat and Mahabal, 2017) will be compared in the future, when TOROS

collaboration had produced enough data to train such complex models. In this small survey context classical Machine Learning algorithms should to be suitable enough.

In the following section we introduce the difference image and Machine Learning techniques to be studied, in section 3 we present the simulated and real data sets that we will be used in the analysis. In section 4 we discuss the results of difference image analysis techniques previous to Machine Learning algorithm implementation. After that we perform the feature selection, and analyze different ML algorithms, estimate their performance for the classification of *real/bogus* detected on difference imaging results. Finally in section 5 we present concluding remarks and future prospects over this work. The software developed and datasets here used are open source, and can be found on TOROS public repositories.³

2. Methods

2.1. Difference Image Analysis

Difference Image Analysis (DIA) is a technique which directly compares two images of the same position in the sky, taken at different epochs. It is usual that one of these is a co-addition of many previously taken images, and has very high signal to noise ratio, known as *reference frame* (R). The other image would be the recently acquired *new image* (N). Both images are assumed to be astrometrically aligned and registered, and so a special kind of subtraction is performed to deliver, after object detection, the transient candidates. The detection procedure in the difference images is in essence a classification problem (source or background), that gives as a result detections of transient sources (TS), detection of artifacts (Ar) and missed transients (MT).

2.1.1. Linearized kernel models

Image differencing goes back to Phillips and Davis (1995) where the Eq. 1 linking the new and the reference images at position (x, y) , is proposed by means of a deconvolution *kernel* ($Ker(u, v)$), bound directly to the change in shape of the PSF between both images. A direct solution would be like Eq. 2 by solving in Fourier space for the kernel (here \widehat{A} denotes the Fourier transform of A).

$$R(x, y) \otimes Ker(u, v) = N(x, y) \quad (1)$$

$$\widehat{Ker} = \frac{\widehat{N}}{\widehat{R}} \quad (2)$$

This can become numerically unstable in the case that the PSF of N is narrower than the PSF of the R image, and also in the presence of high frequency noise on R , making essential to the success of this methodology the good quality of the Reference images. The linearized kernel model was extensively developed in Alard and Lupton (1998), where a decomposition in base functions $B_i(x, y)$ for the kernel is proposed:

³<https://github.com/toros-astro>

$Ker = \sum_i k_i B_i(x, y)$, where k_i are the coefficients. Given the assumption that every pixel in the images are drawn from a Gaussian distribution $\mathcal{N}((R \otimes Ker)(x, y), \sigma(x, y))$ -where \mathcal{N} denotes a normal distribution function-, we can estimate a Maximum Likelihood using the following cost function Q , equivalent to a chi-square test:

$$Q = \int_{(x,y)} [N(x, y) - (Ref(x, y) \otimes Ker(u, v))]^2 / \sigma(x, y)^2 \quad (3)$$

The first proposed kernel was a sum of Gaussian functions modulated by low order polynomials ($p_u(x)$ and $p_v(y)$), like the Eq. 4.

$$Ker(x, y) = \sum_n a_n \mathcal{N}(\mu = 0, \sigma_u, \sigma_v) p_u(x) p_v(y) \quad (4)$$

This model is not versatile enough for complex-structured PSFs, and the image subtractions performed with this methodology may present artifacts. Bramich (2008) proposed a more flexible model modification, as it treats each pixel from the PSF as an independent value. This is basically to use *delta* type basis functions (Eq. 5), where each one is modulated by a coefficient which represents the pixel value located in position (u_i, v_i) .

$$Ker(u, v) = \sum_i k_i \delta(u - u_i, v - v_i) \quad (5)$$

The determination of k_i coefficients is performed during minimization of the function Q , that is, during the likelihood maximisation. This techniques have been applied before on various astronomical analyses such as variable star search, and exoplanet search, for example in Oelkers et al. (2013, 2015) just to name a few. In Bramich et al. (2016) the author explores further options for kernel modelling, adding several selection criteria to optimize the subtraction, although this is not being tested in this work.

2.1.2. Zackay formal image treatments

The proposed image model by Zackay et al. comes from a different statistical point of view, as they choose to represent the pixels of the images as distributions, and attempt to perform hypothesis testing on their values.

In the case of the reference image R the model is as below:

$$R(x, y) = F_R T \otimes P_R + \epsilon_R \quad (6)$$

with T being the *true* image, i.e. taken with a perfect infinite telescope, and no atmosphere influence; $F_R \in \mathbb{R}$ is the transparency, which encloses the atmosphere and instrumental absorption, and would play the role of a flux zero point; the P_R is the PSF, which should be normalized to have unit sum; and ϵ_R is the background noise with variance $V(\epsilon_R) = \sigma_R^2$, assumed to be normally distributed.

This model is suitable for statistical hypothesis testing of the existence of a new source, since it allows the definition of simple null and alternative hypotheses for the new image:

$$\mathcal{H}_0 : N = F_N T \otimes P_N + \epsilon_N \quad (7)$$

$$\mathcal{H}_1(q, \alpha) : N = F_N (T + \alpha \delta_q) \otimes P_N + \epsilon_N. \quad (8)$$

The lack of evidence for the null hypothesis \mathcal{H}_0 favors the existence of a point source at position q with flux α in the new image, which is affected by PSF P_N , transparency F_N and noise ϵ_N . According to the Neyman-Pearson lemma (Neyman and Pearson, 1933a), the most powerful (following the definition given in (Neyman and Pearson, 1933b)) statistic is the likelihood ratio test:

$$\mathcal{L}(\alpha, q) = \frac{\mathcal{P}(N, R | \mathcal{H}_0)}{\mathcal{P}(N, R | \mathcal{H}_1(\alpha, q))}, \quad (9)$$

which can be calculated without prior information on T . It can be proven that maximising equation 9 is the same as maximising the statistic S of equation 10

$$S := \frac{\log(\mathcal{L}(\alpha, q))}{\alpha} \quad (10)$$

and after intermediate calculations available in the appendix A of (Zackay et al., 2016) the expression for the Fourier transform of this statistic is obtained in terms of Fourier transform of known quantities:

$$\widehat{S} = \frac{F_N F_R^2 \overline{\widehat{P}_N} |\widehat{P}_R|^2 \widehat{N} - F_R F_N^2 \overline{\widehat{P}_R} |\widehat{P}_N|^2 \widehat{R}}{\sigma_R^2 F_N^2 |\widehat{P}_N|^2 + \sigma_N^2 F_R^2 |\widehat{P}_R|^2}, \quad (11)$$

By following definitions presented in (Zackay and Ofek, 2017b) and (Zackay and Ofek, 2017a) it is possible to prove that S is the cross match convolution of the real difference image D and its corresponding PSF P_D of equation 13.

$$\widehat{S} = F_D \widehat{D} \overline{\widehat{P}_D} \quad (12)$$

By algebraic manipulations the expression of each one looks like

$$\widehat{D} = \frac{F_R \widehat{P}_R \widehat{N} - F_N \widehat{P}_N \widehat{R}}{\sqrt{\sigma_N^2 F_R^2 |\widehat{P}_R|^2 + \sigma_R^2 F_N^2 |\widehat{P}_N|^2}} \quad (13)$$

$$\widehat{P}_D = \frac{F_R F_N \widehat{P}_R \widehat{P}_N}{F_D \sqrt{\sigma_N^2 F_R^2 |\widehat{P}_R|^2 + \sigma_R^2 F_N^2 |\widehat{P}_N|^2}} \quad (14)$$

with the flux based zero point relative to the difference

$$F_D = \frac{F_N F_R}{\sqrt{\sigma_N^2 F_R^2 + \sigma_R^2 F_N^2}}. \quad (15)$$

For source detection the authors claim that the best option is to determine the locations where S presents peaks outside 5σ (the robust σ). This is the same as a *p-value* test cut. We implemented this as a separated technique, being the true source detection method presented in the already cited work. Among other features, this S image statistic has correlated background noise, and thus is not suitable for every astronomical information extraction. To recover this Zackay et al. derives the formulation of an S_{corr} image in Eq. 98 from Appendix C, which is not affected by the noise level in the vicinity of bright sources and other additional noise components.

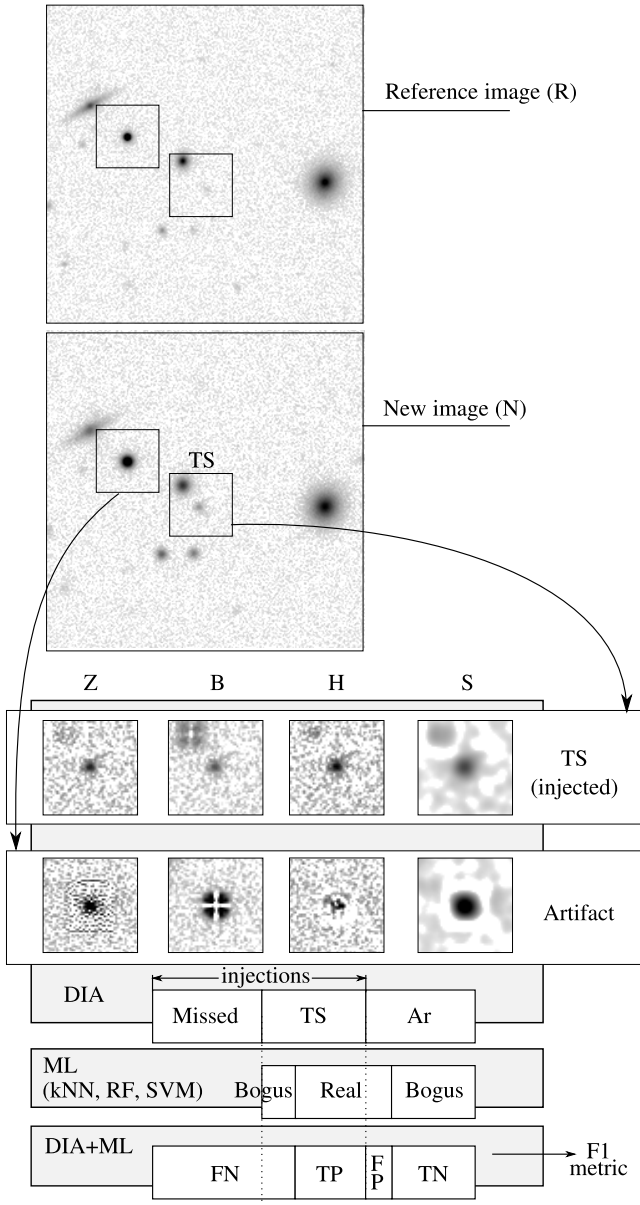


Figure 1: Diagram showing the combined DIA and ML analysis process developed in this work. Reference (R) and new (N) simulated images are processed using DIA by applying Alard and Lupton (H), Zackay et al. (Z), Bramich (B) methods, and S_{corr} (S) image. We show for comparison, an example of a real object, detected from an injected transient source (TS) and an example of a bogus object arising from an artifact (Ar) in the image difference. In the bottom of this figure, we show three bars, splitted into several blocks, each one representing different classification results at different stages of our multi-stage classification process. The first bar represents the results of source detection after the DIA algorithms have performed the image subtraction. The first two blocks of this bar represent the injected sources, which are splitted into Missed and TS detected, and the left block is the resulting set of artifacts Ar. The ML algorithms attempt to learn the classification of TS and Ar subsets into Real and Bogus for each DIA technique. The results of ML are displayed in the second bar, where the sum of the blocks TS and Ar are splitted into Real and Bogus blocks of data. The combination of the DIA+ML results is represented in the last horizontal bar, showing the final quantities of False Negatives (FN), True Positives (TP), False Positives (FP) and the True Negatives (TN). As a final figure of merit we calculate the F1 for each combination of DIA+ML algorithms.

2.2. Elements of Machine Learning

DIA techniques provide the means to detect transient and variable sources on images. However, as several defects arise and are detected as bogus, it is necessary to classify them in order to identify the real sources. Machine Learning (ML) takes advantage of the power of massive amounts of data to generate suitable models to assess the bogus real classification problem. In the classification of transient objects there are several implementations by big collaborations whose capability of collecting data got to overwhelm their human classification capacity. This situation forced them to innovate and apply several machine learning techniques to data selection, in order to make manageable their volumes of raw data, and focusing their attention on the most promising candidates. For a summary of the methodologies implemented on recent years see, e.g., Bloom et al. (2012).

2.2.1. Machine Learning Algorithms

Machine learning (ML) algorithms rely on the use of data to generalize the relations between intervening variables in order to make predictions. There are several classes of algorithms that belong to this area. They differ on how they generalize the examples, and more specifically, on the way they represent the data as models. A relevant review on learning with algorithms can be found in Domingos (2012), where a discussion on aspects that concern to any ML algorithm is carried out. The pertinent jargon adopted in this work can be summarized as:

- objective class: (also target class) the output variables that we want to predict. In binary classification problems they are commonly referred as “positive” and “negative” classes.
- instance: a data example. Can be a training (“labeled”) instance, or a new observation.
- features: the predictors, in short the measurable and/or computable quantities that represent each instance.
- score: a value used to quantify the performance of an algorithm to retrieving the objective class given a training-test dataset.
- confusion matrix: a double entry table where it is possible to visualize the classification results in terms of instances correctly and incorrectly classified. This table originates the score metric values used in classification problems.

For more details or references see, e.g. Mitchell (1997); Hastie et al. (2001). Training datasets are key on supervised ML algorithms, which learn model representations focused in inferring the objective class according to the describing features. A confidence score is used to select the best model representation. In this context, the training data must be labeled, using any previous classification available, such as human training. This validation process is objective at the time of understanding the results, and the meta-parameters of the model already trained provide second-order information on the data itself. In this work we employ supervised classification algorithms from

the standard library *Scikit-Learn* (Pedregosa et al., 2011) –used version is 0.20.1–, which is one of the most popular libraries for ML, written in the convenient programming language Python. In what follows we summarize the ML algorithms used in this work.

- K-Nearest Neighbors (Hastie et al., 2001): a really simple algorithm, basically classifies an instance given the most popular kind in its vicinity on feature space, using as a parameter K the number of close instances to take into account. Although this method has the advantage of fast training, a drawback is that in a high number of dimensions, euclidean distance can be quite computationally expensive and at the same time inaccurate in terms of the feature space real metric. And on top of this you need to store the training data, in order to classify new instances.
- Support Vector Machines: a quite sophisticated algorithm. In a few words, this technique tries to characterize an hyperplane in the feature space that separates different classes. A major quality of this algorithm is that the procedure to find this hyperplane only depends on inner products of feature vectors (in the linear algebra sense), and so, non-linear transformation kernels can be used to make this classifier able to work in a wider range of problems.
- Random Forest (RF) from (Breiman, 2001): this is a so called *meta algorithm* since is basically a combination of simpler ML methods. In this case RF is a collection of Decision Trees that use a randomly chosen subset of features to train.

A decision tree is a rather simple concept, basically is a chain of `if--else` that separates the data taking into account only one feature at the time. There exists several variations of this algorithm depending on the statistic that is used to separate the data, such as information gain, entropy maximisation, etc. The Random Forest brings an ensemble of trees, that cast a vote, and the majority decision is taken.

2.2.2. Feature selection process

Each feature provides different amounts of information about the objective class, being more informative features the most important ones. To estimate this relative importance we can use a wide range of techniques that involve from data visualization, to Principal Component Analysis (PCA, Pearson (1901)). Since some features might provide redundant information, it is convenient to obtain the maximum amount of information with the minimum set of features since it reduces dimensionality and pruning this non-informative variables can lead to an additional computational speed up. In some cases this can be achieved by a transformation of the feature space, by creating new computable features that reduce the dimensionality of the problem (e.g. PCA). Sometimes this is not necessary since it is possible to discard redundant features and just use the ones that have better performance.

In order to maximize the performance of the used ML algorithms we introduce convenient feature selection strategies for

each case. The first strategy is to analyze importance for each feature individually, this is called univariate analysis. The simplest technique is just to filter features with low variance, since a constant quantity would hold no information regarding the target class. This general approach was applied to every tested algorithm. Another univariate technique is to calculate the mutual information between each feature and the target class (Cover and Thomas, 2012). Selecting features which maximize this value would in principle select the features with higher predictive capability on the target class. The mutual information technique was used for the KNN algorithm only.

For the decision tree family of algorithms we introduce the Random Forest derived feature importance calculation (Strobl et al., 2007, 2008). The analysis we used consists of training the model using every feature and in the testing stage carrying a random permutation of the values of each feature, erasing any correlation with the target class. The decrease in performance of the trained algorithm would quantify the importance of the permuted feature, without need of re-training the model. In order to determine the significance of this decrease we include in the training and testing set a control feature with random values, and compare the importances in relation to it. Any feature with an importance less than the random feature would be discarded. This technique is biased in the case of correlated variables, but this could be avoided by pruning them before performing the selection.

Lastly, in the case of SVM we employed a methodology known as Recursive Feature Elimination (Guyon et al., 2002), which works by using an external weighting algorithm, which is evaluated in random subsets of features, and recursively pruning those with low weight. The external weighting algorithm can be any linear model capable of delivering a coefficient for each feature, which makes this technique suitable for linear Support Vector Machines.

2.3. Evaluation of DIA+ML algorithms performance

The focus of this work is centered at the task of recovering transient sources from telescope images by combining the Difference Image Analysis and the Machine Learning methods in order to maximize recovery completeness and minimize its contamination. To test the training stage of a ML algorithm, a labeled testing dataset is used to generate predictions, and the performance can be quantified identically to a hypothesis test, by constructing the *confusion matrix*, as follows: (see Fig. 1)

- True positives (*TP*) are the injected transient sources correctly detected and classified as *real* instances.
- False positives (*FP*) are the artifacts in the image differences and the misclassified instances of *bogus* objects.
- True Negatives (*TN*) are the correctly classified *bogus*
- False Negatives (*FN*) are the lost instances due to misclassification or missed by the DIA.

Notice that the components of the confusion matrix can be computed for any detection and classification problem, in particular, either for the results of the DIA or of the DIA+ML, changing the previous definitions accordingly.

Using the values of the confusion matrix we can compute more sophisticated scores, useful to quantify performance metrics for different optimization strategies.

- Precision measures how many of the classified as positive instances were actually positive. It can be calculated like this: $TP/(TP + FP)$.
- Recall (R) or True Positive Rate (TPR) characterizes how many of the positive examples were actually retrieved. This is $TP/(TP + FN)$.
- False Positive Rate (FPR) is the probability of a false detection, this is $FP/(FP + TN)$.
- False Negative Rate (FNR) is defined as $1 - R$, and is the rate of lost positive examples.

Precision and recall are useful to check the algorithms performance in this unbalanced class context, where the bogus or artifact objects rate depends on the difference imaging method applied. A more informative value is the $F1$ score, derived from the precision and recall metrics, which correctly weights the cost of the errors of losing transients as well as detecting artifacts. $F1$ -measure is the harmonic mean of the P and R metrics, both equally weighted, and can be used as a final figure of merit. It is computed using $2PR/(P + R)$, which is the same as $2TP/(2TP + FN + FP)$ and is also a number from 0 to 1. This metric is less sensitive to unbalanced classification scenarios, because it takes an intermediate value between P and R , but staying closer to the lower value, penalizing the discrepancy of Precision and Recall. At the same time is a metric which does not requires the value of the TN amount, a quantity we cannot derive from any technique, due to the nature of the problem.

To assess the performance of a trained algorithm, usually new data is used. After the training stage, it is possible to detect cases of *over* or *under* training, by using labeled examples which were not processed yet. For this a standard technique named cross-validation, in which the same training set is divided into training and testing subsets, is preferred. This allows to record the mis-classifications and build a confusion matrix. Therefore, we use *stratified* k -fold cross validation (Witten et al., 2016), which splits the training set in k subsets: $k-1$ pieces serving for training and the remaining just for validation purpose. The results of the validation of this k classification algorithms trained are the k confusion matrices, one for each fold of test data. The several metrics explained above are then calculated, yielding a confident performance evaluation of the algorithm.

3. Simulated and real datasets

In order to test and compare the different combinations of DIA and ML techniques for transient detection, we explored a

range of different observing conditions using a purpose made data-set generating simulated images with transients injections. However these simulations are not completely realistic, so that we also test the combined techniques using observations triggered by GW ALIGO alerts. In both cases the injection of transients allows to assess suitable rate estimates to test the performance of the detection methods.

3.1. Generating a simulated image dataset for ML training

We simulated images using ASTROMATIC Software⁴, particularly STUFF and SKYMAKER (Bertin, 2009), which together can produce realistic images of stars and galaxies, for any given telescope hardware configuration, and including image artifacts such as saturation, spikes, secondary mirror spider shadows, etc. Since it is open source software, it is possible to reproduce the results of the image simulation by introducing the same configurations. We simulated the data in several steps:

1. First we used STUFF to produce a catalog of real objects in the field, including galaxies and stars, containing their positions and real photometric properties, as well as shape parameters.
2. Then this catalog is used to make a fits image using SKYMAKER. This is taken as the "reference image" (R).
3. Next some stellar sources (transient) are added to the catalog previously created, at random positions, and with random magnitudes drawn from a fixed Luminosity Function (LF) distribution.
4. The final outcome is a "new image" (N) with the transients sources included.
5. The last stage of the simulation is to perform the DIA subtraction between N and R , and perform the source detection on the resulting difference image.

These steps are repeated once for each point of the explored parameter space, having then, one R and one N image for each of them. The simulation parameter values cover a relevant range of possible observational configurations, taking into account three aspects, namely, the telescope & site characteristics, the sky stellar background, and the relative location and brightness of the transient with respect to their host galaxy. The simulations expand eight parameters, and each one has associated two images, one corresponds to the reference image and the other is the new image. The values of the parameters used in the images simulations are described in Table 1. Regarding the observational configurations, we considered five parameters, namely, the diameters of the telescope primary and secondary mirrors, the seeing FWHM for R and also N , the plate scale, and the exposure time. The values of telescope apertures are selected so that they represent the available instruments by our collaboration. The seeing of the R images took values of 0.8, 1, and 1.3 arc-seconds, following empirical determination of TOROS future site characteristic values (Fig. 3 in Renzi et al. (2009)). The seeing of the N images took values of 1., 1.9, 2.5, motivated by typical and bad observing conditions.

⁴<https://www.astromatic.net/about>

The plate scale and exposure times values are chosen according to the available CCD cameras. Regarding the telescope and site characteristics, we include as particular cases, the Estación Astrofísica de Bosque Alegre (EABA), the TOROS pilot instrument (TORITOS) and the projected 0.6-m telescope for the TOROS site. The number and contrast of the stellar sources are described by the stellar density parameter. The range of stellar densities (given by the STARCOUNT_ZP parameter) represents fields of different environments going from typical densities of an mid galactic latitude, and up to densities of less than 5 degrees from the MW disk center at $l \sim 60$ deg of longitude, with a limiting magnitude of $i \sim 19$. The luminosity distribution of these sources are governed by a power law, with an exponent which took the values of 0.1, 0.5, 0.9 dex per mag (STARCOUNT_SLOPE parameter in the SKYMAKER software) Also, we allow the variation of the background surface brightness, using values of 20 and 21 for reference images, typically taken mostly on dark nights, and 20, 19, and even 18 for new images, taken in different conditions. Regarding the host galaxies of the injected transients, we sampled the relative brightness and the angular distance from the host center from uniform distribution in the ranges $[-4, 1]$ magnitudes and $[0, 5]$ half light radius, respectively. This allows to explore different transient/host relative configurations, including low relative luminosities and position ranging from the center up to the outer stellar halo. For a given observational configuration (or set of parameters), we define the magnitude range where reliable photometry can be obtained, based on the photometric calibration obtained applying RANSAC (Fischler and Bolles, 1981) robust linear regression on standard sources. The RANSAC method prunes spurious sources, and obtains an estimation of slope and zero point values, not sensitive to outliers, and at the same time also provides a filter mask identifying this outliers.

The explored parameter space may include configurations which are not probable. For instance the combination of a 1.54 mirror, with 300 seconds exposure time, in a night with a bright background light (i.e. moon light), a large plate scale and a broad seeing. A number of this corner cases are present in the explored configuration space, and in some of this cases simulated images appear completely saturated, and in others the photometric quality is extremely low. This corner cases have been discarded, leaving a total of 26205 groups of N , R and DIA differenced images. The results shown in the next sections include all sensible points in the explored parameter space, which comprises an heterogeneous combination of image qualities. Nevertheless, the independent photometric calibration of each image informs us the range of validity of flux determination on each configuration making us able to fairly compare results among the whole simulated dataset.

In section 4, we discuss the general trends that result from our analysis. Although our parameter space do not cover all possible configurations for telescope optics and site, we provide the codes that allow to simulated any other observational configuration.

A total of 3272784 transients were injected, placed on top of an extended object (as expected in the case of Kilonovae) with random angular position and distance relative to the host

galaxies below 5 half light radius. The simulated galaxies have different morphological types, and also have different redshift values, random orientation and ellipticity and their luminosities are chosen according to a Schechter luminosity function. The R magnitudes of the transient objects are disposed with a random offset from the host galaxy, drawn from a uniform distribution, between values -4 and $+1$ magnitudes.

In the Fig. 1 we present a scheme of the results of the difference image subtraction and the following ML classification of real and bogus transient sources. Besides the reference and new simulated images we show the result of the subtractions performed for this pair of images.

The subtractions were carried out by three different implementations of the techniques introduced above, namely: Zackay, Alard & Lupton, and Bramich algorithms, and we show stamps of bogus and real objects for visual comparison. We also present the resulting S_{corr} image computed as in Eq. 98 from Appendix C of Zackay et al. (2016). As can be seen in the stamps, the properties of the subtractions vary according to the applied methodology. It is worth noticing the shape and the appearance of the same objects after the subtraction has been performed. In the case of the transient source injected we see that every technique presents a point source of almost equal size, except for the S image, which shows an enlarged light distribution. This is due to the nature of the S image, which is a convolution of the Z image with its own PSF. In the second row, we show artifacts originated in the same bright point source. The reason this artifact is consistent in every technique is them failing to correctly match the photometric properties of both R and N images. In every case we find different structures and these arise because of the intrinsic differences among methods. The artifact in the Z image has a boxy shape due to the PSF determination methodology, in the case of the B image the kernel matches the center of the stellar sources but fails to adequately account for the flux in the wings. Similarly in the case of H we find that the source is still visible, though it lacks a clear structure. In the S case we see an excess of intensity, with a smooth profile, though surrounded by negative pixels, signs of a flux mismatch in the Z image. For the implementation of the Alard and Lupton method we adopted the publicly available HOTPANTS⁵ software by Becker (2015) (version 5.1.11). The Bramich implementation is a Python code by the authors, available at <https://github.com/toros-astro/ois> (version 0.1.14), as well as Zackay et al. implementation, which is also a Python code by the authors available at <https://github.com/toros-astro/ProperImage>. Both implementations are built upon standard scientific libraries such as NumPy (version used here is 1.15.14), SciPy Jones et al. (2001–) (version 1.1.0) and Astropy (Astropy Collaboration et al., 2013; Price-Whelan et al., 2018) (version 3.0.4), and run on versions of Python 2.7 as well as 3.6. Also ois and Properimage are fully documented and tested, and they are Open Source, free to the community to use. Many examples and details of the implementation can be found in the documentation

⁵<http://www.astro.washington.edu/users/becker/v2.0/hotpants.html>

PARAMETER	UNITS	VALUES	TOROS instruments		
			EABA	TOROS	TORITOS
aperture of the telescope	[m]	[0.4, 0.6, 1.54]	1.54	0.6	0.4
reference seeing FWHM	[arcsec]	[0.8, 1, 1.3]	1.3	0.8	0.8
new image seeing FWHM	[arcsec]	[1.3, 1.9, 2.5]	2.5	1.0	1.0
plate scale	[arcsec/pix]	[0.3, 0.7, 1.4]			
exposure times	[sec]	[60, 120, 300]			
stellar density	[stars per sq deg]	[4e3, 8e3, 32e3, 64e3, 128e3, 256e3]			
stellar luminosity distribution exponent	[dexp per mag]	[0.1, 0.5, 0.9]			
background brightness (R)	[mag per arcsec ²]	[20, 21, 22]			
background brightness (N)	[mag per arcsec ²]	[18, 19, 20]			
relative brightness from host	r-band magnitudes	sampled from Unif(-4,1)			
angular distance from host	half light radius	sampled from Unif(0, 5)			

Table 1: Parameter space for simulated images to be explored for transient detection.

at <http://optimal-image-subtraction.readthedocs.io> and <http://properimage.readthedocs.io>. This implementation applies a set of pre-processing stages to the images, in order to correctly treat the background, bad pixel masking and interpolation, and PSF determination. It is worth noticing that the Zackay implementation works faster when the exposure times of the reference and new images are equal. In this case there is almost no need for a zero point calibration, which saves computational time. Alard-Lupton implementation is written in C programming language, also it employes a simpler Gaussian PSF assumption, making this method faster than the others by a factor of almost 4X.

3.2. Injection of transient objects on observed images

The simulated images previously used are practical for developing the DIA techniques and generating a training dataset for the bogus/real classification problem. Also this approach allows to explore the dependence of the algorithm performance on different observing settings and transient properties. Nevertheless, this approach is limited by the simplifying hypothesis. In order to take into account the flaws that arise in the observing process and subsequent analysis, we present in this subsection the process of injecting transients sources into real observational images. We used images obtained by the TOROS collaboration as part of the follow up of the triggers during Advanced LIGO science run O2. The images were obtained for the gravitational wave event GW170104, using the Estación Astrofísica Bosque Alegre (EABA) 1.54-m Newtonian telescope. The instrument was set to white light image acquisition, and the CCD used was a Apogee Alta U16. Since the observations were performed hours after the gravitational wave event trigger was received, we had no previous references of the selected targets. The "reference a posteriori" methodology was implemented, using images taken over the following months. The objects were selected by cross-correlating and filtering the skymap provided by LSC GCN:20364 and the galaxy catalog from (White et al., 2011), according to the methodology described in the first TOROS follow up paper (Díaz and Collaboration, 2016). The set of observed galaxies comprise NGC1341, NGC1567, NGC1808, ESO0555-022, ESO3564-

014, PGC073926, PGC147285, at two epochs separated by eleven months. The images were reduced using a standard image processing, and then co-added for each epoch using the *SWarp* (Bertin, 2010) public software. The procedure for the subtraction analysis is consistent with the one performed on the simulated dataset. We perform the subtractions using each image twice, once as a reference, and once as a new image. Given the BH-BH merger nature of the GW event (Abbott et al., 2017b) it was not expected any EM emission incoming from this source, and in fact no kilonovae were detected Sanchez et al. (2018). Therefore we decided to inject transients for the intended analysis. In order to simulate a transient object with consistent PSF, we replicated 15 of the actual stars in each frame, inside the true dynamic range of the images. The total number of realizations was 176 using the observed galaxies, yielding a total of 2640 transient injections. The subtraction requires the images to be registered, to that end we developed the python package *astroalign*⁶. This package was inspired by *astrometry.net*⁷ but it does not rely on a prefixed star catalog, instead it aligns two images comparing the asterisms drawn by the brightest stars in the field. These procedures require a complete pipeline for processing and data management as the CORRAL framework (Cabral et al., 2017). This is an open source python package which merges a database connection interface with a Model-View-Controller paradigm, making building complex experimental designs simpler and straightforward, and letting the framework figure out the multi-processing itself. This allows us to write specific processing steps, like those involved in DIA and ML combined analysis, according to an intuitive data handling model. Therefore, we built a similar processing pipeline, using the same sequence of steps, for both the real and the simulated datasets. The completeness and contamination of the transient detection by the DIA pipelines can be increased and reduced, respectively, by the application of the ML algorithms previously introduced. These metrics, among others, were used to rank the transient detection agents, thus allowing to chose the most suitable approach for applications on TOROS images.

⁶<https://github.com/toros-astro/astroalign>

⁷<http://astrometry.net>

4. Comparing transient detection agents

The DIA methods can deliver either direct subtractions (those which return an astronomical image without any convolution process), or convolved subtractions, as the S statistic proposed by Zackay et al.. For the direct difference image methods, the transient candidate identifications were performed using SE_{TRACTOR}⁸ (version 2.19.5), with the same configuration parameters.

The S image, in turn, is a cross-correlation of the difference image with its own PSF, and so the detection of transient candidates is different from the other DIA methodologies. By calculating the S_{corr} statistic from the S image and performing a robust determination of its mean and standard deviation (μ , σ), we

can define the significance (α) of the detection of a candidate in a given pixel as follows:

$$\alpha = (S_{corr} - \mu) / \sigma$$

In Fig.2 it can be seen the distribution of significances for the artifacts and transient sources as well as both cuts of α for 3.5 and 5 in vertical lines, for both datasets. The chosen threshold of 3.5 is relaxed with respect to the originally proposed value of 5 by Zackay et al., attempting to detect dim transient sources. Although this increases the number of artifacts, the following ML analysis is expected to label them as bogus. Since SE_{TRACTOR} performs a pre-convolution on every image it scans, we cannot use it on S_{corr} images. Instead we obtained candidates on the S_{corr} images using SEP⁹, an open source package capable of running a source extraction without kernel pre-convolution. This software provides photometric measurements similar to those delivered by SE_{TRACTOR}. The threshold used with this source detection technique is again a limit on 3.5 over the background.

Therefore we were able to analyze the candidate sources, its photometric properties and parameters measured by SE_{TRACTOR} and SEP for every DIA technique, and use this results as features for the ML stage. We end having five DIA transient detection results: detections over Zackay D images (Z), over Bramich (B) and over Alard-Lupton differences (A), the three of them provided by SE_{TRACTOR}; detections over S_{corr} using SEP (S_{SEP}), and detections using a simple pixel threshold ($S_{3.5\sigma}$). In what follows (Subsec. 4.1), we explore the main differences between samples of artifacts, real transient sources and missed sources for each DIA method. Since the comparison is based on the very same images, we can directly relate the occurrence of both real transient sources and missed objects among methods, not being this possible in the case of artifacts, since those can be random subtraction errors misidentified as transient candidates. The labeled candidates for Z, B, A and S_{SEP} DIA results, and their photometric quantities measured are the inputs of the feature selection process, for training and testing of the ML models aimed at doing the bogus-real classification (Subsec. 4.3).

4.1. Performance of the DIA methods

The fraction of occurrences of the different classes of objects is our first piece of information regarding the subtraction methods performance, prior to any Machine Learning technique application. In Table 2 we show the number of transient sources (TS), missed detections (missed) and artifact sources (Ar), along with the corresponding fractions, TPR, FNR and FPR, respectively. We also report the F1 statistic after the DIA implementation that will be later compared to the same statistic after the ML application.

By definition the values of FNR and TNR always add up to one, and FPR can be any number, since the normalization is over the total number of injected sources. Regarding the rates of recovered transient sources (TPR), we read that there is variability on the results for different techniques. There is a baseline of 50% for every technique, and a top value of 93%, finding intermediate values in the simulated as well in the real dataset. The number of missed objects is larger in the case of the simulation, since we injected transients in a larger number of simulated images, covering a wide range of experimental configuration (as indicated in Table 1).

For the simulated dataset we can read that Bramich finds less transients, and at the same time produces less artifacts. For the real dataset it finds more transients than any other technique, and produces a relatively low amount of artifacts. The technique which finds more transient sources is S_{SEP} , with a relatively low number of artifacts for the simulated dataset. In the case of the real dataset the scenario is the opposite. The method which generates more artifacts in the simulated case is Alard-Lupton, yielding more than twice the amount of false detections than Zackay technique. In case of the real dataset this happens for the $S_{3.5\sigma}$ which has a FPR of 200, and Zackay generates less artifacts than any other DIA method. This behaviour can be explained with the tendency of $S_{3.5\sigma}$ of finding local maximae in the edges of images, or near bright sources, an issue we would like to address in the future.

In the simulated dataset we find that the $F1$ statistic is systematically higher than the real dataset results. This is mostly due to extremely high FPR we measure in the latter. It is clear that we have more sources of confusion in the real images, this could explained by the presence of instrumental defects on the CCD camera used, poor flat field calibration, and correlated noise coming from the stacking procedure. These effects are not straightforward to include in the simulations therefore we can think of them as a representation of an optimistic case scenario, in comparison to the observations.

4.2. Analyses of the DIA results

In order to compare the photometric properties of the transient candidates for every DIA method, we performed a photometric calibration with the flux of the simulated astronomical sources, by using a robust linear regression as already detailed in Sec. 3. The measured magnitudes of the transients recovered in the simulation generated images shows agreement among the different DIA methods. We show in the Fig. 3 the mean and the standard error of the difference between the injected and measured magnitudes of the transients recovered in the simulated

⁸<https://www.astromatic.net/software/sextactor>

⁹<https://github.com/kbarbary/sep>

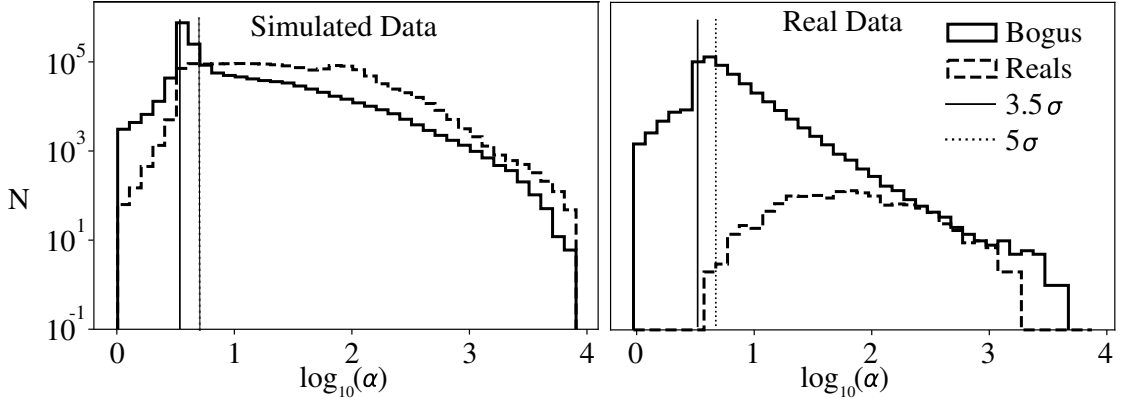


Figure 2: Distribution of α values for the artifacts and transient sources for S_{corr} detection technique. Notice the logarithmic scales in both axes. In vertical lines we include the position of thresholds for $\alpha = 3.5\sigma$ and $\alpha = 5\sigma$. Left: simulated dataset. Right: real dataset.

Simulated dataset							
	TS	Missed	Ar	TPR	FNR	FPR	F1
Z	1,933,065	1,339,719	3,170,089	0.59	0.41	0.97	0.46
B	1,596,713	1,676,071	1,979,948	0.49	0.51	0.60	0.47
A	1,971,291	1,301,493	5,537,472	0.60	0.40	1.70	0.37
S_{SEP}	2,180,390	1,092,394	2,456,876	0.67	0.33	0.75	0.55
$S_{3.5\sigma}$	2,092,625	1,180,159	2,700,107	0.64	0.36	0.83	0.52

Real dataset							
	TS	Missed	Ar	TPR	FNR	FPR	F1
Z	2296	344	25914	0.87	0.13	9.8	0.15
B	2468	172	47731	0.93	0.07	18.1	0.09
A	2179	461	110,025	0.83	0.17	41.7	0.04
S_{SEP}	2099	541	128,820	0.80	0.20	48.8	0.03
$S_{3.5\sigma}$	2043	597	528,927	0.77	0.23	200.4	0.008

Table 2: Number of detections of transient sources (TS), missed detections (missed) and artifact sources (Ar), and the corresponding rates (TPR, FNR and FPR, respectively) for the implemented DIA methods on the simulated and real datasets, computed before ML analysis. We include the S_{Corr} candidates thresholded and extracted with SEP .

images, as a function of the transient r -magnitude. We also show in this Figure the results for transients injected on EABA observations, and the subset of simulated images that are closest to the observing configuration of the EABA telescope and site. The difference between the EABA observations and the corresponding simulation arises because of several factors. Most importantly is that the real dataset images are stacks of images, and this largely enhances its dynamic range, allowing to observe bright sources without saturation, and at the same time, sources at magnitudes fainter than 17 with an good signal to noise ratio. In the simulations, the aperture photometry might not be able to ideally capture the true flux value of the bright saturated or enlarged sources. Prior to reaching the saturation limit, the linearity of the CCD response is lost, progressively deviating flux measurements from true values. The limiting magnitude difference between simulations and real data are due to the actual performance of the EABA instrument, and the atmospheric conditions during the nights of O2 follow up obser-

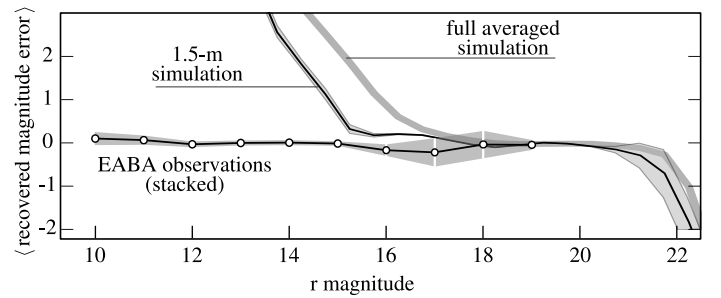


Figure 3: Magnitude difference between injected and recovered sources as a function of the magnitude of the injected transients, for the full averaged simulation (thick light gray), the stacked EABA images (circles) and a simulation with an equivalent mirror size and exposure time (solid black). Error bars are $1 \times \sigma$ wide.

vations. Taking this into account, it is worth noticing that the simulated and observed images are consistent within a range of approximately 5 magnitudes.

We also studied several statistics for each class of object, trying to gain insight into their properties. The Cumulative Luminosity Function for each class in the simulated dataset is displayed on the top row of the Fig. 4, where it can be seen that for the transient sources the methods are roughly equivalent. We find that for the magnitude range $r \geq 15$ the methods behave similarly, but Zackay and S-Corr have more transient sources detected than the rest of the techniques. In this faint end of the luminosity function the main reason for losing objects is the detection limits. In the bright end $r \leq 15$ however, there is not a clear pattern, and besides the true missed objects, the already discussed errors in the magnitude measurements for bright objects due to saturation could increase the discrepancy with the simulated magnitude values.

In the case of the artifacts, both in the simulated and observed dataset, we find similar behaviour in the DIA techniques, although it is worth to notice that the accumulated number of objects is in disagree with the reported figures in Tab. 2, this is due to the fact that many artifact sources present flux measurements without astrophysical meaning. The values of magnitudes calculated for a portion of the artifacts fall outside the

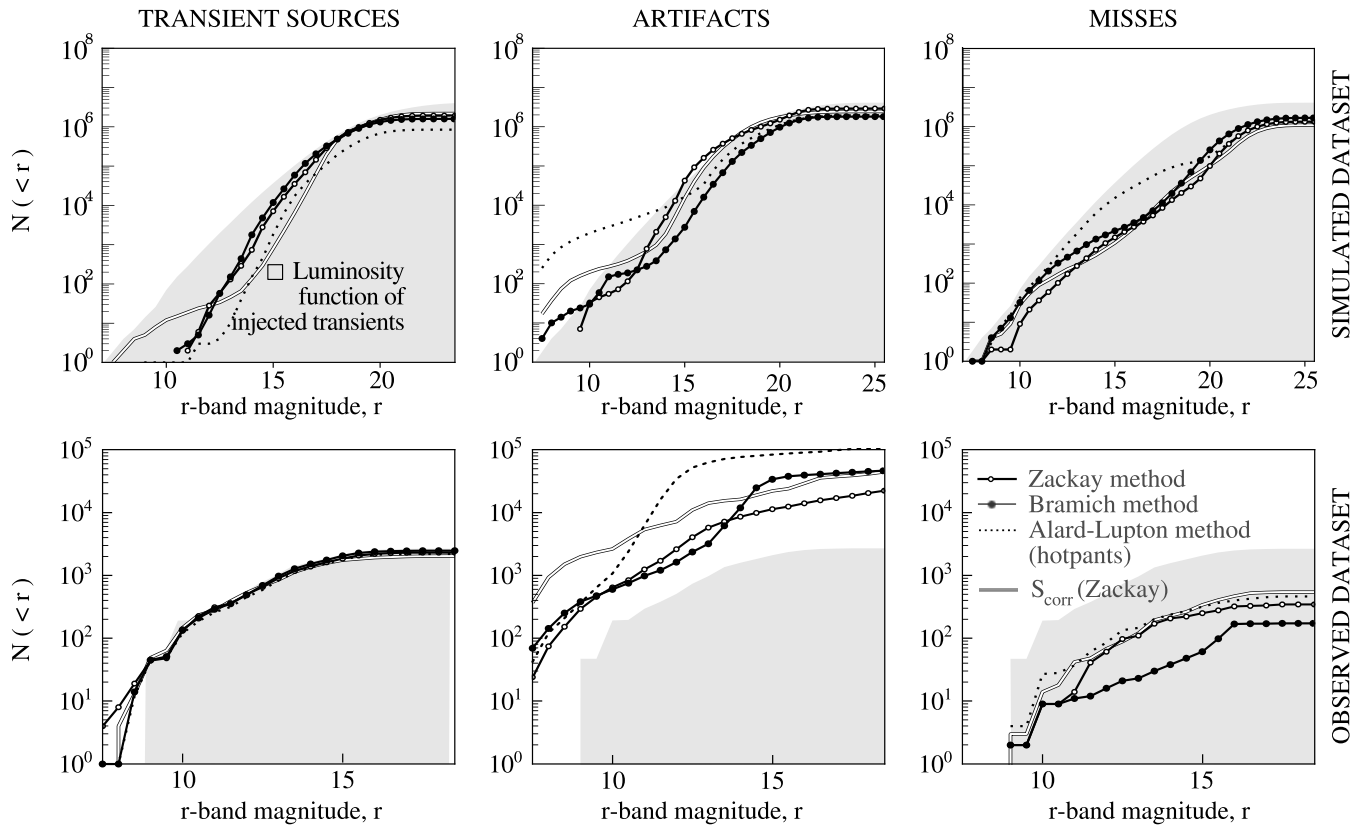


Figure 4: Cumulative Luminosity functions for Transient Sources (TS) (left column), Artifacts (Ar) (center column) and Misses (right column) object classes. In the first row we have the simulated data and in the second row the real dataset. The cumulative luminosity function of the injected sources is displayed in every panel as the shaded area. The line code for DIA techniques is: Zackay's is always in lines with white dots, where Bramich's is in lines with filled dots, Alard-Lupton is in dotted lines, as well as S_{corr} statistics is in black solid line.

range presented in the Fig. 4.

The right panel of Fig. 4 top row, corresponding to the missed objects in the simulations, shows that all methods fail to recover an important number of sources fainter than 21 magnitudes, and also objects brighter than 12 magnitudes. Since this missed, and transient sources sets of objects are the complement of each other we find similar explanations for the bi modality of lost sources and the deficits of injected recovered objects. Particularly S-Corr is the method which lose less injected sources, followed by Alard-Lupton, although the latter losses objects in the whole range presented, in contrast with the other methods. As a comparison with the real dataset, we find that there is not such gap in between the faint and bright end of the missed objects luminosity function, but instead a smooth ever increasing distribution is observed. In the real dataset, we have a different picture, with every method losing objects, almost equally, except for the Bramich method, whose distribution seems to be fainter, and shows accumulated numbers below every other technique. We also observe a slight increase in the number of objects lost for the S-Corr compared to the rest of the techniques, although is should be recalled that there is a big gap between the sample sizes of simulated and real datasets.

In certain way a higher contamination of bogus may be acceptable since Machine Learning algorithms could separate them from the true interesting candidates, and on the other side, a high FNR is quite undesirable since those candidates are “lost forever”. This compromise should be constrained in advance, and shouldn’t be taken for granted. Every algorithm could be optimized by using iterations or other supplementary techniques, moving the scenario to a more favorable ratios scenario.

4.3. Machine Learning Results

As explored above, we have photometric properties for the detected candidates in the several DIA images. On top of this we also have shape properties, as well as high order statistical moments on their light distributions provided by SExtractor - an in S_{corr} case SEP-. This labeled dataset together with the mentioned features, constitutes the input instances for training and testing ML algorithms in the task of classifying the bogus and real objects.

In order to asses the different scenarios we simulated, our machine learning experiments were conducted in several steps:

- We grouped the dataset in terms of three simulation configuration values: the mirror diameter of the telescope, the exposure time, and the seeing of the new image. This gave us 27 subsets of data, where we conducted identical and independent experiments.
- Each experiment was carried out firstly by splitting the dataset into a training and *final testing* set, with a 20%-80% proportion respectively, due to the enormous amount of data available for training.
- The training subset is used to perform feature preprocess and selection, and to perform a k-fold cross validation performance measure for three ML algorithms: k-Nearest

Neighbors, Random Forests, and linear Support Vector Machines.

- We calculated the confusion matrix for the ML classification (Bogus–Real), and combined it with the DIA performance metrics, in order to rank the DIA+ML methodologies using an overall figure of merit. This is done by deriving a confusion matrix from the injected sources, through the DIA (Missed, TS, Artifacts), to its final DIA+ML classification results (FN, TP, FP, and TN), such as illustrated in Fig. 1.

The used Machine Learning algorithms as previously stated were:

- k-Nearest Neighbors, using 7 neighbors, with uniform weights and euclidean distances (using scaled feature values).
- Random Forest, with 800 trees, with up to 7 features per tree, stopping the tree growing if less than 20 examples per leaf, using a Gini impurity criterion.
- Support Vector Machines, using L2 norm penalization, with a tolerance parameter of 10^{-5} , solving the dual optimization problem, and weighting the classes if unbalanced for their frequency.

All the configurations for the ML algorithms correspond to SciKit Learn version 0.20.1.

4.3.1. The feature selection process

We performed a preprocessing of the features, by scaling them to zero mean, and unit variance. Afterwards we applied univariate analysis by using a variance threshold cut of 0.1, pruning constant features. Following this simple treatment we used three different feature selection strategies, adapted to each ML algorithm tested .

To calculate the importances for kNN algorithm, we used the mutual information of the features and the target class, selecting the percentile 30 as a threshold cut.

For RandomForest we applied a feature selection process following Bloom et al. (2012), using a RandomForest training stage, and picking those features that were the most informative in the majority of the individual trees. To avoid bias in the selection we pruned the correlated features, and afterwards we followed the methodology described in associated Python package *rffpimpin* a 10-fold cross validation experiment. To determine the unimportant features we added to the training dataset a uniformly distributed random variable. Using this procedure we can set the zero value of the scale as the importance of this Random feature. We calculated the mean and standard deviation of the 10 values obtained in the 10-fold experiment, tossing away those features consistent with the values obtained for the Random one.

For Support Vector Machines we applied a Recursive Feature Elimination on a 6-fold experiment, using an elimination step of 1, and choosing *F1* as the scoring metric to maximize.

4.3.2. Evaluation of DIA+ML algorithms

In Fig. 5 we show a heatmap of values of $F1$ statistic (scaled by a factor of 10^3) for the results of the 12 DIA+ML combinations (4 DIA techniques and 3 ML algorithms). The map is obtained by grouping several possible instrumental configurations, spanning the dimensions of the FWHM for the new image (N_{FWHM}) measured in arcseconds, the exposure time (t_{exp}) measured in seconds, and the diameter of the primary mirror (D) in cm, and performing the ML *train-test* experiments on each group. This covers a wide range of instruments, from small to middle size telescopes, and from good to poor observing conditions. There exists though, several possible configurations which are not covered by our analysis, designed mostly for TOROS collaboration available instruments. Nevertheless the analysis is valid for numerous transient search science collaborations with instruments falling within the range of our simulation parameters, such as (piofthesky, black gem, assassn, los alerces, catalina sky survey, etc). It is also worth noticing that for each of the 27 instrumental configurations analyzed we are mixing the combination of values of the rest of the simulation parameters (listed in Tab. 1), and in consequence including several dissimilar transient detection scenarios into the same ML experiment.

This result shows an expected dependence in the simulation parameters, clearly favouring longer exposure times and smaller seeing FWHM for the new image, there is also weaker but present dependence on mirror size. The strongest dependence of the $F1$ value holds with the DIA technique applied. It is clear that independently of the ML used Alard & Lupton DIA method has better performance in terms of $F1$ statistic in the simulations, followed by Zackay's techniques (including S_{corr}). There is no major difference among ML algorithms for a given DIA technique, but a small advantage seems to be obtained when using RandomForest.

We also include the values of $F1$ for the real dataset in the top of the map, which can be compared to the highlighted equivalent simulation. In the observed dataset we find that generally Bramich is better ranked, and the best DIA+ML technique is its combination with kNN. Notice that the overall $F1$ for this DIA method is much better in this observed dataset than in the simulations. However the range of values for the ML+DIA combination are comparable, despite the difference in the $F1$ values of DIA only.

In order to better compare the performances reported in Fig. 5 we have marginalized over the groups of instrumental configurations, showing the quartiles of the distribution of $F1$ values in the Fig. 6. In order to compare these values before and after ML classification we also include as horizontal lines the quartiles of the distribution of the $F1$ values obtained in each group after the application of the DIA methods only. It is clear from this figure that Alard&Lupton technique is better ranked than any other DIA method, and at the same it experiences the largest boost in performance after the application of any ML algorithm. We also show in dots the results for the observed dataset and in horizontal dotted line we also include the $F1$ value for the corresponding DIA methodology (see Tab. 2). In the real dataset

we find that the combinations which make use of kNN and RandomForest algorithms are significantly better ranked than SVM combinations. In general Bramich DIA technique results in better performance as measured by $F1$ statistic, comparable to the best values obtained in the simulations. However we notice that this result is linked mostly to the improvement that ML algorithm provides, which is larger than in the simulated dataset. It is noticeable also, that Alard&Lupton results are consistent between simulation and real dataset within the uncertainties, being this also consistent with the Bramich results for the real dataset case. This will be further explored when the TOROS collaboration instrument data acquisition period begins, and larger collections of images are available.

As a global result we report that the best combination of DIA+ML for the simulated dataset is Alard & Lupton implementation Hotpants, and RandomForest. For the real dataset in turn we find that the best performance is with Bramich DIA combined with kNN machine learning algorithm. For these two cases we present the selected features and their normalized relative importances in Fig. 7. The included features in this figure sum up 90% of the total importance calculated for the whole set of selected features in each DIA+ML case. We can see that in the simulation the 5 most important features in the case of the best ranked DIA+ML for the simulated dataset, that is A&Lupton, are in order ELONGATION, FLAGS, FWHM_IMAGE, MAG_AUTO and CLASS_STAR. In case of the real dataset, for Bramich+kNN the 5 most important features are in order B_IMAGE, CXX_IMAGE, CYY_IMAGE, SN, FLUXERR_ISO. The details on the calculation and meaning of each feature are included in Appendix A. The disagreement among the maps of importance might be explained considering the differences in the DIA+ML algorithm, and its feature selection procedure. Still, we can detect that the most important features represent as expected, shape and brightness parameters.

In order to gain some insights on the simulated dataset, we explored some of the dependences of the metrics of performance with the parameters of the injections. In the Fig. 8 we show the values of Recall R and $F1$ metric scores, as two dimensional maps, for two pairs of the simulation parameters, for the A&Lupton+RandomForest technique. The first one, in left panel, shows the dependence of R with the values of relative position and brightness to the host galaxy. This maps clearly shows a strong dependence of the Recall of the final ML+DIA on the relative brightness, and at the same time shows no dependence on the distance to the host galaxy center. The second map, in right panel, shows the $F1$ values as a function of the parameters of the stellar luminosity function of the star field on the images: STARCOUNT_SLOPE and STARCOUNT_ZP. As explained in Sec. 3.1 the stellar luminosity function is a power law, and the exponent is the value of STARCOUNT_SLOPE, the STARCOUNT_ZP parameter is the total density of stars in the field (see Tab. 1). This map indicates that in a dense environment the $F1$ is lower, indicating the number of stars in the image as source of confusion for the DIA+ML technique. At the same time a lower slope, which traduces into in more bright stars at a fixed total density, pushes the scoring to lower values. It is clear then that dense stellar fields, like the ones in the galactic plane,

		EABA:													
		746	704	668	840	813	680	775	719	621	585	506	450	F1	
$N_{FWHM}=1.3$	60	400	629	627	629	587	593	593	634	636	638	731	742	738	
		600	637	643	643	599	605	606	648	660	660	734	743	737	
		1540	655	658	658	614	623	624	679	681	681	739	746	741	
	120	400	715	738	734	640	646	645	818	843	832	732	737	733	
		600	734	745	742	653	668	665	835	844	832	737	758	750	
		1540	738	756	752	667	683	681	817	848	838	742	754	748	
	300	400	754	796	791	683	705	703	760	805	791	774	787	782	
		600	764	802	798	703	719	718	770	804	794	782	792	787	
		1540	781	802	798	716	720	721	777	818	805	789	795	790	
$N_{FWHM}=1.9$	60	400	564	577	577	511	527	528	583	594	595	688	684	683	
		600	577	589	590	526	539	541	606	611	612	690	694	691	
		1540	589	594	595	534	557	558	616	626	627	699	704	701	
	120	400	663	695	691	592	605	604	814	839	827	686	702	697	
		600	671	704	699	599	617	617	807	831	821	692	706	701	
		1540	683	712	707	614	637	636	797	845	833	696	715	709	
	300	400	713	745	738	622	632	636	764	809	799	720	737	733	
		600	706	740	735	629	655	657	771	812	801	728	736	733	
		1540	716	755	749	650	678	678	772	823	810	734	745	739	
$N_{FWHM}=2.5$	60	400	501	512	513	442	459	462	532	541	542	636	639	638	
		600	509	528	528	450	462	466	555	558	559	645	647	645	
		1540	523	534	534	463	472	476	566	576	576	654	663	660	
	120	400	619	648	644	540	561	561	793	820	808	656	664	660	
		600	628	656	651	539	574	572	784	812	799	667	665	661	
		1540	639	663	657	552	580	580	780	828	819	673	673	668	
	300	400	649	684	678	555	600	603	756	804	797	680	696	693	
		600	650	683	679	574	616	615	743	802	791	688	690	687	
		1540	652	690	686	580	611	611	742	793	784	691	694	691	
$t_{exp}[s]$	D [cm]	kNN	RF	SVM	kNN	RF	SVM	kNN	RF	SVM	kNN	RF	SVM		
		Zackay			Bramich			A&Lupton			S _{corr}				

Figure 5: A heatmap of $F1 \times 10^3$ score performance values for each of the 27 instrumental configurations (as rows) where the DIA+ML algorithms have been applied (as columns). In the top row the scores for the observed dataset are included in the same grayscale, and a separated row is highlighted for comparison, corresponding to the simulations with values of (N_{FWHM} , $t_{exp} = 60s$, $D = 1.54$).

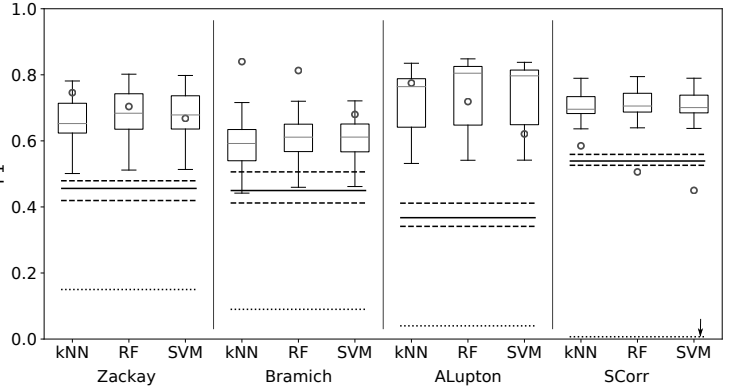


Figure 6: The distribution of F1 values presented in Fig. 5, displayed as boxes and whiskers. Inside the box we have the median, the box edges displays the 25 and 75 percentile, as well as the whiskers show the minimum and maximum values of the sample. In horizontal solid lines we show the median of the F1 score prior to ML, and as dashed the quartiles of these values. In circles we have the values of F1 for the real dataset. In dotted lines we also show the F1 value for the real data corresponding to Tab. 2.

are places where a decrease in the performance of the DIA+ML is expected. These class of studies are possible to evaluate by using a multi parameter simulation, like the one carried out.

5. Conclusions

We developed open source tools for image subtraction following the DIA techniques from Bramich (2008) and Zackay et al. (2016). We also made use of HOTPANTS implementation of Alard-Lupton algorithm. The Python packages developed, together with HOTPANTS, were mounted on top of a CORRAL data processing pipeline, in order to systematically perform image subtractions with transient candidate injections. The images used for this were drawn from EM counterpart search, carried out by TOROS during O2 LIGO-Virgo-Scientific collaboration science observing run in 2017. This observations were performed during January and November, using the “reference a posteriori” methodology previously applied by TOROS collaboration. Additionally we developed comprehensive simulations of images, exploring a multi dimensional parameter space of instrumental configuration as well as observing conditions, as detailed in Tab. 1, generating millions of transient injected on top of extended sources over several thousands of images. The nature of the injected transients does not include moving objects, or stellar variability, focusing the analysis on Kilonova/Supernova detection scenarios.

The mentioned pipelines measured the ratios of recovery of injected transients, as well the source contamination and loss for each DIA algorithm. We also compared their photometric results, including the S_{corr} image photometry. In order to separate the true transient candidates from the spurious artifact detections we applied Machine Learning algorithms, trained with data generated by our pipeline. We carried out a feature selection stage, and completed a cross validation train-test experiment, in order to calculate score metrics such as Precision and

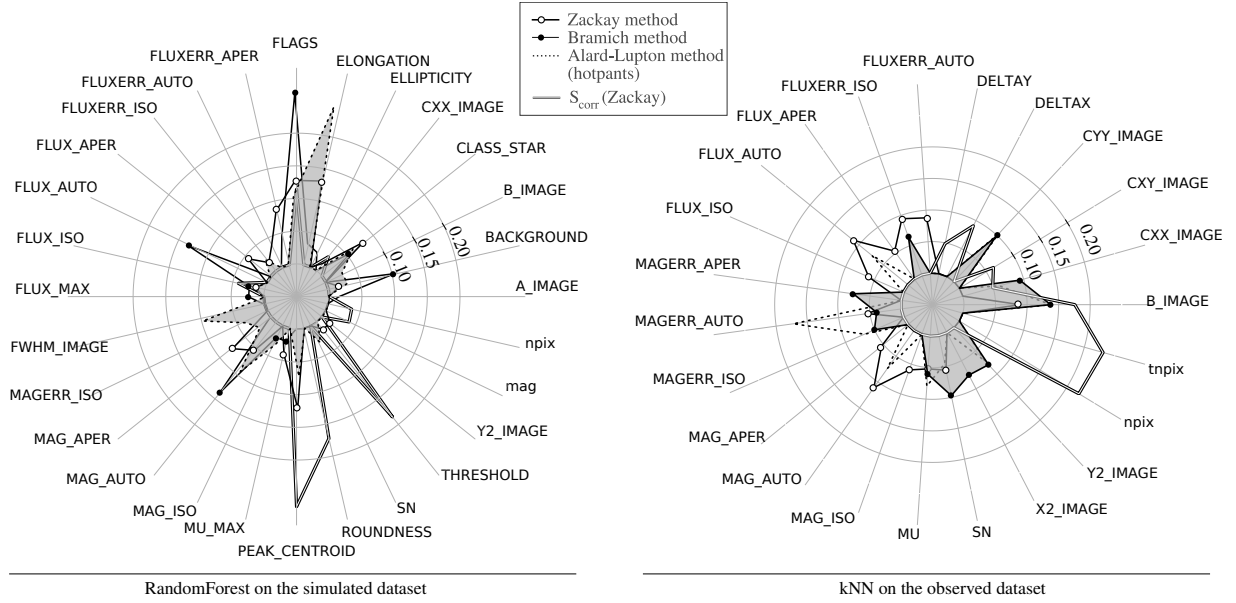


Figure 7: Radial plot of normalized feature importances. The left panel corresponds to the feature selection for the RandomForest algorithm in the simulated dataset. The right panel corresponds to the feature selection specific to the kNN algorithm as explained in Sec. 2.2.2, in the real dataset. The details on the calculation and meaning of each feature are included in Appendix A.

Recall, useful to compare performances in an unbalanced class context, and ranked the methodologies by combining them in the $F1$ statistic.

The comparison in the simulated and real data showed that the scenario was of relevance in the performance of the different combination of methodologies, bringing differences which can relate to the techniques as well as the nature of data used. Our results shows that Zackay’s image subtraction techniques, including S_{corr} , are more suitable for transient detection as a standalone technique. However it is clear that the Machine Learning algorithms are key to complete the task of selecting the relevant transient candidates, by setting apart the artifacts which contaminate and are uninteresting. After the application of these algorithms, and looking at the final performance metric $F1$ we conclude that, among the DIA+ML combinations tested, the better ranked technique were: in the real EABA images dataset *Bramich+kNN*, and in the simulated dataset *A&Lupton+RandomForest*. Although we find consistency, within the uncertainties, among every ML applied technique in the simulated data, for every DIA method. This is also valid between the real and simulated dataset, for the A&Lupton and Zackay case. For these selected methods we also report the more important features, determined by Random Forest permutation importance in the case of the simulations, and by univariate analysis in the case of the real dataset.

Seizing the large and complex simulation generated, we analyzed the dependency of the Recall metric on the environment of the injected transients, in relation to the host galaxy, and the stellar field present in the simulated images. Concluding that is more likely to generate artifacts and miss a fraction of transient objects in a dense and bright stellar field, and also to miss a greater fraction of detections if the contrast in brightness with

the host galaxy is small, independently of its spatial relative location.

These results are very important for the development of the data processing pipeline of the TOROS collaboration which comprises different telescopes and instruments. A future extension of this work is to tackle the genuine transient time-series astrophysical classification for large amounts of data.

Simulation results and data used in this work are available to the community, in the format of candidates catalog tables at O. (2019)¹⁰.

Acknowledgements

This work was partially supported by the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina) and the Secretaría de Ciencia y Tecnología de la Universidad Nacional de Córdoba (SeCyT-UNC, Argentina). M.B and M.D. acknowledge NSF support through grant nsf-hrd 1242090. JLNC is grateful for financial support received from the GRANT PROGRAMS FA9550-15-1-0167 and FA9550-18-1-0018 of the Southern Office of Aerospace Research and development (SOARD), a branch of the Air Force Office of the Scientific Research International Office of the United States (AFOSR/IO). The authors also thank for their kind suggestions and commentaries to D. Bramich and B. Zackay.

This research has made use of the adsabs.harvard.edu/, Cornell University xxx.arxiv.org repository, the SIMBAD database, operated at CDS, Strasbourg, France.

Also the Python programming language and the following scientific libraries: Numpy, Scipy, Astropy, scikit-Learn, and

¹⁰ <https://zenodo.org/record/2658714>

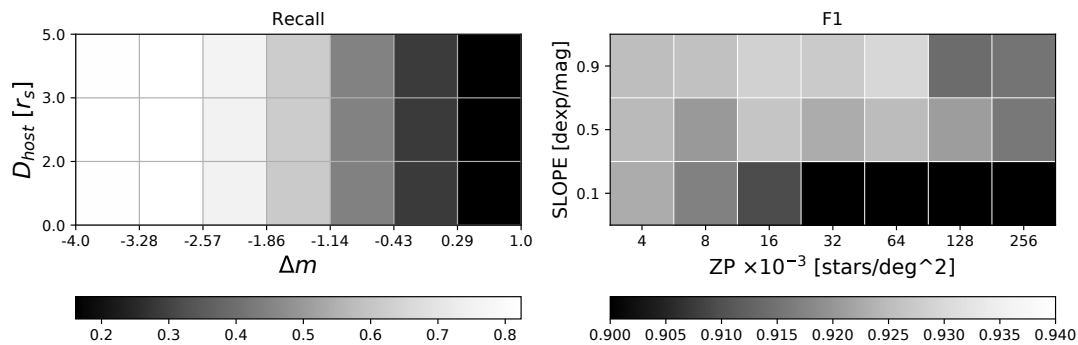


Figure 8: Maps of Recall R and $F1$, for A&Lupton+RandomForest, for the simulations in two different projections. The left panel shows the values of Recall R , as a projection of distance of the injected transient relative to the center of the host galaxy (in scale radius units) vs the difference in brightness between the transient and the host galaxy. The right panel in turn, shows values of $F1$ in the plane of slope of the stellar luminosity function (STARCOUNT_SLOPE) on the images, vs the total density of stellar sources (STARCOUNT_ZP). Notice that greyscale is not the same for both panels.

rfpimp.

References

References

Abbott, B.P., Abbott, R., Abbott, T.D., Acernese, F., Ackley, K., Adams, C., Adams, T., Addesso, P., 2017a. Multi-messenger observations of a binary neutron star merger. *ApJ* 848, L12. URL: <http://stacks.iop.org/2041-8205/848/i=2/a=L12>.

Abbott, B.P., Abbott, R., Abbott, T.D., Acernese, F., Ackley, K., Adams, C., Adams, T., Addesso, P., et al. (LIGO Scientific and Virgo Collaboration), 2017b. Gw170104: Observation of a 50-solar-mass binary black hole coalescence at redshift 0.2. *Phys. Rev. Lett.* 118, 221101. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.118.221101>, doi:10.1103/PhysRevLett.118.221101.

Abbott, B.P., Collaboration, T.L.S. (LIGO Scientific Collaboration and Virgo Collaboration), 2016. Observation of gravitational waves from a binary black hole merger. *Phys. Rev. Lett.* 116, 061102. URL: <http://link.aps.org/doi/10.1103/PhysRevLett.116.061102>, doi:10.1103/PhysRevLett.116.061102.

Alard, C., Lupton, R.H., 1998. A Method for Optimal Image Subtraction. *ApJ* 503, 325–331. doi:10.1086/305984, arXiv:astro-ph/9712287.

Astropy Collaboration et al., 2013. Astropy: A community Python package for astronomy. *aap* 558, A33. doi:10.1051/0004-6361/201322068, arXiv:1307.6212.

Barnes, J., Kasen, D., Wu, M.R., Martínez-Pinedo, G., 2016. Radioactivity and Thermalization in the Ejecta of Compact Object Mergers and Their Impact on Kilonova Light Curves. *AJ* 829, 110. doi:10.3847/0004-637X/829/2/110, arXiv:1605.07218.

Becker, A., 2015. HOTPANTS: High Order Transform of PSF ANd Template Subtraction. *Astrophysics Source Code Library*. arXiv:1504.004.

Bertin, E., 2009. SkyMaker: astronomical image simulations made easy. *Mem. Soc. Astron. Italiana* 80, 422.

Bertin, E., 2010. SWarp: Resampling and Co-adding FITS Images Together. *Astrophysics Source Code Library*. arXiv:1010.068.

Bloom, J. S. et al., 2012. Automating Discovery and Classification of Transients and Variable Stars in the Synoptic Survey Era. *PASP* 124, 1175. doi:10.1086/668468, arXiv:1106.5491.

Bramich, D., Horne, K., Alsubai, K., Bachelet, E., Mislis, D., Parley, N., 2016. Difference image analysis: automatic kernel design using information criteria. *Monthly Notices of the Royal Astronomical Society* 457, 542–574.

Bramich, D.M., 2008. A new algorithm for difference image analysis. *MNRAS* 386, L77–L81. doi:10.1111/j.1745-3933.2008.00464.x, arXiv:0802.1273.

Breiman, L., 2001. Random forests. *Machine learning* 45, 5–32.

Cabral, J.B., Sánchez, B., Beroiz, M., Domínguez, M., Lares, M., Gurovich, S., Granitto, P., 2017. Corral framework: Trustworthy and fully functional data intensive parallel astronomical pipelines. *Astronomy and Computing* 20, 140–154. doi:10.1016/j.ascom.2017.07.003, arXiv:1701.05566.

Cover, T.M., Thomas, J.A., 2012. *Elements of information theory*. John Wiley & Sons.

Díaz, M.C., Collaboration, T., 2016. GW150914: First Search for the Electromagnetic Counterpart of a Gravitational-wave Event by the TOROS Collaboration. *ApJ* 828, L16. doi:10.3847/2041-8205/828/L16, arXiv:1607.07850.

Díaz, M. C. et al., 2017. Observations of the first electromagnetic counterpart to a gravitational-wave source by the toros collaboration. *ApJ* 848, L29. URL: <http://stacks.iop.org/2041-8205/848/i=2/a=L29>.

Djorgovski, S.G., Drake, A.J., Mahabal, A.A., Graham, M.J., Donalek, C., Beshore, E., Larson, S., 2010. Exploring the Variable Sky with the Catalina Real-Time Transient Survey, in: *The First Year of MAXI: Monitoring Variable X-ray Sources*, p. 32.

Domingos, P., 2012. A few useful things to know about machine learning. *Commun. ACM* 55, 78–87. URL: <http://doi.acm.org/10.1145/2347736.2347755>, doi:10.1145/2347736.2347755.

Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* 24, 381–395. URL: <http://doi.acm.org/10.1145/358669.358692>, doi:10.1145/358669.358692.

Fortson, L. et al., 2012. *Galaxy Zoo: Morphological Classification and Citizen Science*. CRC Press, Taylor and Francis Group. pp. 213–236.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. *Machine Learning* 46, 389–422. URL: <https://doi.org/10.1023/A:1012487302797>, doi:10.1023/A:1012487302797.

Hastie, T., Tibshirani, R., Friedman, J., 2001. *The Elements of Statistical Learning*. Springer Series in Statistics, Springer New York Inc., New York, NY, USA.

Ivezic, for the LSST Collaboration, 2008. LSST: from Science Drivers to Reference Design and Anticipated Data Products. *ArXiv e-prints* arXiv:0805.2366.

Jones, E., Oliphant, T., Peterson, P., et al., 2001–. *SciPy: Open source scientific tools for Python*. URL: <http://www.scipy.org/>. [Online; accessed 7today].

Kasen, D., Badnell, N.R., Barnes, J., 2013. Opacities and Spectra of the r-process Ejecta from Neutron Star Mergers. *AJ* 774, 25. doi:10.1088/0004-637X/774/1/25, arXiv:1303.5788.

Law, N. M. et al., 2009. The Palomar Transient Factory: System Overview, Performance, and First Results. *PASP* 121, 1395. doi:10.1086/648598, arXiv:0906.5350.

LSST Science Collaboration et al., 2009. *LSST Science Book, Version 2.0*. *ArXiv e-prints* arXiv:0912.0201.

Mitchell, T.M., 1997. *Machine Learning*. 1 ed., McGraw-Hill, Inc., New York, NY, USA.

Neyman, J., Pearson, E.S., 1933a. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 231, 289–337. URL: <http://rsta.royalsocietypublishing.org/content/231/694-706/289>, doi:10.1098/rsta.1933.0009, arXiv:<http://rsta.royalsocietypublishing.org/content/231/694-706/289.full.pdf>.

Neyman, J., Pearson, E.S., 1933b. The testing of statistical hypotheses in relation to probabilities a priori. *Mathematical Proceedings of the Cambridge Philosophical Society* 29, 492510. doi:10.1017/S030500410001152X.

O., S.B., 2019. HDF5 Table for Machine Learning on Difference image analysis. URL: <https://doi.org/10.5281/zenodo.2658714>, doi:10.5281/zenodo.2658714.

Oelkers, R. J. et al., 2015. Difference Image Analysis of Defocused Observations With CSTAR. *AJ* 149, 50. doi:10.1088/0004-6256/149/2/50, arXiv:1410.4544.

Oelkers, R.J., Wang, L., Zhou, J., Macri, L.M., CSTAR, PLATO, 2013. Difference Image Analysis of 2009 CSTAR Observations from Dome A in Antarctica, in: *American Astronomical Society, AAS Meeting 221*, id.352.23.

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine* 2, 559–572.

Pedregosa, F. et al., 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12, 2825–2830.

Phillips, A.C., Davis, L.E., 1995. Registering, psf-matching and intensity-matching images in iraf, in: *Astronomical Data Analysis Software and Systems IV*, volume 77 of *ASP Conference Series*. p. 297.

Price-Whelan, A. M. et al., 2018. The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *aj* 156, 123. doi:10.3847/1538-3881/aabc4f.

Rau, A., Kulkarni, S.R., Law, N.M., Bloom, J.S., Ciardi, D., Djorgovski, G.S.e.a., 2009. Exploring the Optical Transient Sky with the Palomar Transient Factory. *PASP* 121, 1334. doi:10.1086/605911, arXiv:0906.5355.

Renzi, V. et al., 2009. Caracterización astronómica del sitio cordón macón en la provincia de salta. *Boletín de la Asociación Argentina de Astronomía* 52, 285–288.

Sanchez, B., Beroiz, M., Diaz, M., Macri, L., M., D., 2018. No bh em emission. in prep. .

Sedaghat, N., Mahabal, A., 2017. Effective Image Differencing with ConvNets for Real-time Transient Hunting. *ArXiv e-prints* arXiv:1710.01422.

Strobl, C., Boulesteix, A.L., Kneib, T., Augustin, T., Zeileis, A., 2008. Conditional variable importance for random forests. *BMC Bioinformatics* 9, 307. URL: <https://doi.org/10.1186/1471-2105-9-307>, doi:10.1186/1471-2105-9-307.

Strobl, C., Boulesteix, A.L., Zeileis, A., Hothorn, T., 2007. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics* 8, 25. URL: <https://doi.org/10.1186/1471-2105-8-25>, doi:10.1186/1471-2105-8-25.

White, D.J., Daw, E.J., Dhillon, V.S., 2011. A list of galaxies for gravitational wave searches. *Classical and Quantum Gravity* 28, 085016. URL: <http://stacks.iop.org/0264-9381/28/i=8/a=085016>.

Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2016. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Zackay, B., Ofek, E.O., 2017a. How to COADD Images. I. Optimal Source Detection and Photometry of Point Sources Using Ensembles of Images. *ApJ* 836, 187. doi:10.3847/1538-4357/836/2/187, arXiv:1512.06872.

Zackay, B., Ofek, E.O., 2017b. How to COADD Images. II. A Coaddition Image that is Optimal for Any Purpose in the Background-dominated Noise Limit. *ApJ* 836, 188. doi:10.3847/1538-4357/836/2/188, arXiv:1512.06879.

Zackay, B., Ofek, E.O., Gal-Yam, A., 2016. Proper Image Subtraction: Optimal Transient Detection, Photometry, and Hypothesis Testing. *ApJ* 830, 27. doi:10.3847/0004-637X/830/1/27, arXiv:1601.02655.