

Classification of gravitational-wave glitches via dictionary learning

Miquel Llorens-Montegudo¹, Alejandro Torres-Forné^{1,2}, José A. Font^{1,3} and Antonio Marquina⁴

¹Departamento de Astronomía y Astrofísica, Universitat de València, C/ Dr. Moliner, 50, Burjassot (València E46100, Spain.

²Max Planck Institute für Gravitationsphysik (Albert Einstein Institute), D-14476 Potsdam-Golm, Germany

³Observatori Astronòmic, Universitat de València, C/ Catedrático José Beltrán 2, 46980, Paterna (València) E46100, Spain.

⁴Departamento de Matemáticas, Universitat de València, Dr. Moliner 50, 46100, Burjassot (València) E46100, Spain.

Abstract. We present a new method for the classification of transient noise signals (or glitches) in advanced gravitational-wave interferometers. The method uses learned dictionaries (a supervised machine learning algorithm) for signal denoising, and untrained dictionaries for the final sparse reconstruction and classification. We use a data set of 3000 simulated glitches of three different waveform morphologies, comprising 1000 glitches per morphology. These data are embedded in non-white Gaussian noise to simulate the background noise of advanced LIGO in its broadband configuration. Our classification method yields a 96% accuracy for a large range of initial parameters, showing that learned dictionaries are an interesting approach for glitch classification. This work constitutes a preliminary step before assessing the performance of dictionary-learning methods with actual detector glitches.

Keywords: gravitational waves, detector characterization, machine learning

1. Introduction

After the landmark observations of gravitational waves (GWs) from mergers of compact binaries [1, 2, 3, 4, 5, 6], GW astronomy has been established as a brand new way to study the cosmos. The first two observational campaigns of Advanced LIGO and Advanced Virgo have provided the first few detections. The current upgrade and commissioning of these detectors will lead to an increase in their sensitivity, in the scale of the cosmic horizon to search for sources, and in the event rates. The upcoming observing run of Advanced LIGO and Advanced Virgo, O3, due to start in early 2019, promises a plethora of new GW data and discoveries [7].

Despite the recent detections, noise removal remains one of the most challenging problems in GW data analysis. The sensitivity of current ground-based detectors sharply degrades at frequencies below a few tens of Hz due to gravity-gradient (seismic) noise

and above ~ 2 kHz, due to quantum fluctuations of the laser [8]. At intermediate frequencies is where interferometers become the most sensitive, being mainly limited by thermal noise due to Brownian motion of the suspensions and mirrors. Many noise sources affecting detectors are non-Gaussian and non-stationary, altering the sensitivity of the detectors in real time. In addition, transient noise signals of both instrumental and environmental origin, commonly known as ‘glitches’, may not only disturb astrophysical GW signals (as dramatically manifested in the infamous glitch affecting the merger signal of GW170817 [6]) but also mimic true signals, increasing the false-alarm rate and producing a decrease in the detectors’ duty cycle. Uninterrupted efforts in detector commissioning and characterization are made to reduce the effects of glitches. In particular, improving the identification and classification of glitches is fundamental to increase the efficiency of the detection. Current efforts employ approaches as diverse as Bayesian inference, Principal Component Analysis, machine learning, deep learning, and even combinations of machine-learning and citizen science (see e.g. [9, 10, 11, 12, 13, 14]).

In this paper we study the suitability of dictionary-learning algorithms for glitch denoising and classification, taking as starting point the method introduced in our work for GW denoising using dictionaries [15]. Mathematically speaking, a dictionary is a matrix of m atoms (signals in our case) of length n organized as columns. The dictionary-learning approach is based on the so-called sparse representation, which states that a given signal can be reconstructed as a linear combination of only a few atoms, $\mathbf{u} = \mathbf{D}\boldsymbol{\alpha}$, where \mathbf{u} is the reconstructed signal, $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_m]$ is the (overcomplete) dictionary composed of a atoms of length n such that $m > n$, and $\boldsymbol{\alpha} \in \mathbb{R}^p$ is a sparse vector containing the coefficients of the representation. Since the dictionary is overcomplete, the solution vector $\boldsymbol{\alpha}$ is not unique, hence we use the *basis pursuit decomposition* proposed in [16]. The goal of this paper is to test the suitability of dictionary-learning techniques to denoise glitches and classify them by their morphology. One should note that by “denoising glitches” we do not mean to remove glitches from the background but to obtain a clear shape from the glitch morphology. In this sense, we treat glitches as signals like we did in our first paper [15]. To test our approach we follow the analysis carried out by [9] where the authors compared three different pipelines to classify simulated glitches embedded in Gaussian noise.

For training the dictionaries and validate our method in a controlled environment, we follow [9] and generate several sets of synthetic glitches which can be classified into three different waveform morphologies: sine Gaussian (SG), Gaussian (G) and Ring-Down (RD). All glitches to be denoised and classified are injected into simulated non-white Gaussian noise similar to that of advanced LIGO in the proposed broadband configuration with different values of the signal-to-noise ratio (SNR).

The classification method we introduce in this paper uses dictionaries in two steps. First, we perform the denoising step aiming for the most faithful reconstructions using three trained dictionaries, one dictionary per waveform morphology. Due to our signal configurations (which will be justified later on) all three dictionaries are likely to yield reconstructions from all glitches, regardless of their morphology. For example, a SG

glitch (injected in noise) may be reconstructed by all three dictionaries with relative fidelity. However, the reconstruction from a SG dictionary should be closer to a clean model of a SG glitch than those produced by both the G and RD dictionaries, whose atoms contain glitches of other waveform morphologies. Motivated by this hypothesis, in the second step of our procedure we look for the dictionary whose reconstruction is closer to a perfect waveform morphology, for each unknown glitch to be classified (i.e. closer to an ideal SG, G, or RD glitch). In order to achieve this we use three untrained dictionaries whose atoms are whole centered glitches, each one serving as examples of ideal glitches of a single waveform morphology. Every denoised glitch from the first step is reconstructed by all three untrained dictionaries using as few atoms as possible, so that instead of generating a faithful reconstruction each dictionary will yield a “reconstruction” close to an ideal glitch of its own morphology while trying to get as close to the denoised glitch as possible. Finally, the predicted waveform morphology of an unknown glitch will be that of the untrained dictionary that produced the reconstructions closest to the three respective reconstructions of the denoising dictionaries. Our study shows that learned dictionaries are an interesting new approach for glitch classification as we manage to successfully classify about 96% of simulated glitches for a large range of initial parameters.

This paper is organized as follows. In section 2 we summarize the mathematical background of the sparse representation and dictionary learning techniques. In section 3 we describe the morphology of the three classes of simulated glitches and the parameters that define them. The classification algorithm we have developed using the dictionaries is presented in section 4. In section 5 we describe the tests we have performed in order to determine the best set of parameters of the method that produces optimal results and we discuss the application of these techniques (denoising + classification) to a long run of 3000 glitches. Finally, in section 6 we present the main conclusions of our work and outline possible future directions of research.

2. Mathematical framework

2.1. Sparse reconstruction

In [15] we considered sparse reconstructions of GW signals over trained dictionaries, employing numerical relativity catalogs of core-collapse supernova signals and binary black hole waveforms. In this paper, we apply the same approach to classify and reconstruct simulated GW glitches embedded in Gaussian noise. Similar to our work on GW denoising using dictionaries [15] we assume that the way in which glitches are embedded into noise can be described by the linear degradation model

$$\mathbf{f} = \mathbf{u} + \mathbf{n}, \quad (1)$$

where $\mathbf{f} \in \mathbb{R}^n$ is the data from the detector, $\mathbf{u} \in \mathbb{R}^n$ is the glitch to be recovered for later classification, and $\mathbf{n} \in \mathbb{R}^n$ is random Gaussian noise. Given an overcomplete dictionary

$\mathbf{D} \in \mathbb{R}^{n \times m}$, where the number of atoms m is greater than their length n , there is a sparse vector $\boldsymbol{\alpha} \in \mathbb{R}^m$ for which $\mathbf{D}\boldsymbol{\alpha} \sim \mathbf{u}$, where

$$\boldsymbol{\alpha} = \arg \min_{\boldsymbol{\alpha}} \{ \|\mathbf{f} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_0 \} . \quad (2)$$

In this equation $\|\cdot\|_0$ and $\|\cdot\|_2$ stand for the L^0 -norm and the L^2 -norm, respectively. The former is just the number of nonzero components of its argument. This problem can be written as a convex and unconstrained variational problem by substituting the L^0 -norm of $\boldsymbol{\alpha}$ by the L^1 -norm as a penalty term of the problem weighted by a Lagrangian multiplier λ ,

$$\boldsymbol{\alpha} = \arg \min_{\boldsymbol{\alpha}} \{ \|\mathbf{f} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \} , \quad (3)$$

an approach which is known as *basis pursuit* [16] or *LASSO* [17]. The regularization in the L^1 -norm promotes zeros in the components of the vector coefficient $\boldsymbol{\alpha}$ and, thus, the solution of this variational problem is typically the sparsest one.

The Lagrangian multiplier λ is also called the regularization parameter, as it regulates the level of detail to be recovered in the sparse representation $\boldsymbol{\alpha}$ of the input signal \mathbf{f} . The higher the value of λ the more the L^1 -norm term weights, making the coefficients of $\boldsymbol{\alpha}$ tend to zero when solving problem (3), which results in less atoms being used for the sparse representation (i.e. for the reconstruction of \mathbf{f}). On the other hand, using low values of λ favours the *fidelity* term (the L^2 -norm term in Eq. (3)), which results in more (or even all) atoms being used. This transforms the problem into a simple least-squares problem and yields a reconstruction as similar as possible to the input data. The optimal value for a given signal, λ_{opt} , is defined to be the one which gives the best results according to a suitable metric function applied to the denoised signal and to the original one, measuring the quality of the recovered signal. In this work we use two estimators, namely the Mean Squared Error,

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 , \quad (4)$$

where \hat{Y} and Y are the reconstructed and original signals respectively, and n is the number of samples, and the structural similarity (SSIM) index [18] which takes into account the structural information. The SSIM index varies between -1 (minimum similarity) and 1 (maximum similarity) and is defined as

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} , \quad (5)$$

where c_1 and c_2 are constants, μ_x (μ_y) is the average of x (y), σ_x^2 (σ_y^2) the variance of μ_x (μ_y) and σ_{xy} the covariance of x and y . In order to facilitate the comparison between the two estimators we rewrite the SSIM index so that its value ranges from 0 (maximum similarity) to 1 (minimum similarity), like the MSE,

$$\text{DSSIM}(x, y) = \frac{1 - \text{SSIM}(x, y)}{2} . \quad (6)$$

This metric function is called the structural dissimilarity index.

To solve the LASSO problem we use the modified least-angle regression (LARS) algorithm [19]. This algorithm is similar to a forward stepwise regression: it starts with the regression coefficients equal to zero, and for each iteration it finds the predictor u_i most correlated with the response f and takes a large step in the same direction until other predictor achieves a similar correlation. However, instead of continuing along the first predictor, LARS proceeds following recursively the equiangular direction between the predictors of equal correlation. The main advantage of this method is the efficiency when dealing with more dimensions than points ($p > n$), which is the case of overcomplete dictionaries (see [19] for further details).

2.2. Dictionary learning problem

Before the denoising, the dictionaries are trained with a set of signals splitted in p patches of length n , $\mathbf{U} = [u_1, \dots, u_p] \in \mathbb{R}^{n \times p}$. The number of training patches is large compared with the number of atoms and their length, $p \gg m, n$, because of the sparsity condition and the overcompleteness of the dictionary.

The trained dictionary is obtained by adding the dictionary matrix \mathbf{D} as a variable in the minimization problem,

$$\boldsymbol{\alpha} = \arg \min_{\boldsymbol{\alpha}, \mathbf{D}} \frac{1}{n} \sum_{i=1}^p \{ \|\mathbf{u}_i - \mathbf{D}\boldsymbol{\alpha}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i\|_1 \} , \quad (7)$$

where the summation index i indicates the i -th row of matrix $\boldsymbol{\alpha} \in \mathbb{R}^{p \times n}$, which contains the coefficients of the sparse representation of each atom. The columns $(\mathbf{d}_i)_{i=1}^p$ of the dictionary are constrained to have an L^2 -norm less or equal to one, $\mathbf{d}_i^T \mathbf{d}_i \leq 1$, to prevent \mathbf{D} from being arbitrarily large.

The problem is solved by the algorithm proposed by Mairal et al in [20] with the mini-batch optimization. This is a block-coordinate descend method which minimizes \mathbf{D} and $\boldsymbol{\alpha}_i$ separately for each iteration t ,

$$\boldsymbol{\alpha}^t = \arg \min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2} \|\mathbf{u}_t - \mathbf{D}^{t-1} \boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right\} , \quad (8)$$

$$\mathbf{D}^t = \arg \min_{\mathbf{D}} \frac{1}{t} \sum_{i=1}^t \left\{ \frac{1}{2} \|\mathbf{D}\boldsymbol{\alpha}_i^t - \mathbf{u}_i\|_2^2 + \lambda \|\boldsymbol{\alpha}_i^t\|_1 \right\} , \quad (9)$$

with the advantage of being parameter-free and not requiring any learning rate.

3. Signal set

In this work we do not use actual detector glitches. Instead, following [9], we simulate three simple kinds of glitch morphologies, namely sine Gaussian (SG), Gaussian (G)

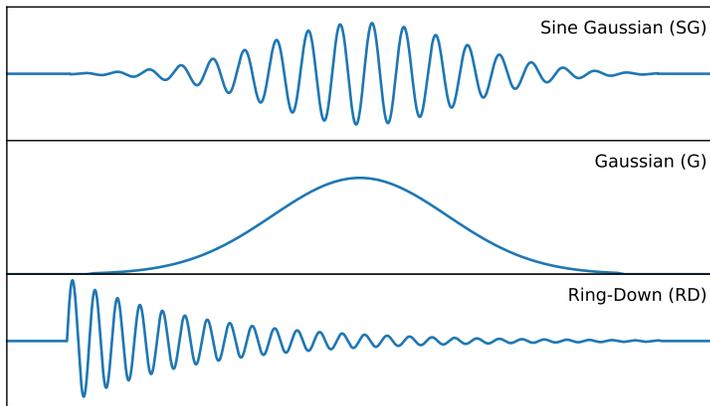


Figure 1. Examples of the three different glitch morphologies used in our data sets.

Table 1. Minimum and maximum parameters of the simulated glitches, from [9].

	Waveform	Minimum	Maximum
f_0 (Hz)	All	40	1500
h_{rss} ($\text{Hz}^{-1/2}$)	All	5×10^{-22}	4×10^{-21}
Q	SG, RD	2	20
Duration (s)	G	0.001	0.01

and ring-down (RD). They are defined by the following functions

$$h_{\text{SG}}(t) = h_0 \sin \{2\pi f_0(t - t_0)\} e^{-(t-t_0)^2/2\tau^2}, \quad (10)$$

$$h_{\text{G}}(t) = h_0 e^{-(t-t_0)^2/2\tau^2}, \quad (11)$$

$$h_{\text{RD}}(t) = h_0 \sin \{2\pi f_0(t - t_0)\} e^{-(t-t_0)/2\tau}, \quad (12)$$

where f_0 is the central frequency, t_0 is a characteristic time for each of the waveforms, namely the time at the centre of the SG and G waveforms and at the beginning of the RD waveform, $\tau = Q/\sqrt{2\pi}f_0$, with Q being the quality factor, and $h_0 = h_{\text{rss}}/\sqrt{\tau}$, with h_{rss} being the root sum squared amplitude of the glitch. These parameters are randomly chosen within the ranges shown in Table 1, with a linear distribution on their logarithms to get enough samples of all orders of magnitude. Examples of the waveform morphology for all three glitches are shown in Figure 1. The SG and RD waveforms are relatively similar between them, both displaying a distinctive oscillatory pattern, and more complex than the G waveform. This variety of signals serves the purpose of testing the capability of the classification algorithm in a realistic scenario involving different types of glitches.

We generate three separate sets of glitches with the sampling rate of the Advanced LIGO/Virgo detectors, 16384 Hz. Since for training each dictionary we want to use the same amount of training patches (namely 20000), and each morphology has its own range of durations (which translates to different range of samples), we generate different

amounts of glitches. Moreover, we will test dictionaries with different atoms' lengths. Therefore, we also need to be sure we will be able to generate enough patches with the largest atoms. Taking both conditions into account, we generate a training set of 915 SG glitches, 62440 G glitches, and 603 RD glitches. These signals are normalized to their L^2 -norm to ensure the best convergence conditions for the learning algorithm. Then we use a set of 300 glitches (100 per waveform morphology) for parameter optimization, and a bigger set of 3000 glitches (1000 glitches per waveform morphology) to test the final classification algorithm. All signals from these two sets are scaled so that their maximum value is equal to one.

The glitches used for training and testing are injected into white Gaussian noise weighted by the power spectral density (PSD) of Advanced LIGO in the proposed broadband configuration, as explained in [21]. We rescale the signals to a SNR value of 20. The SNR is defined as

$$\text{SNR} = \sqrt{4\Delta t^2 \Delta f \sum_{k=1}^{N_f} \frac{|\tilde{u}(f_k)|^2}{S(f_k)}}, \quad (13)$$

where \tilde{u} indicates the Fourier transform of signal u , S is the PSD of the noise corresponding to the sensitivity curve of the detector, f_k is each of the components of the frequency vector, N_f is the number of positive frequencies, and Δt and Δf are the time step and frequency step, respectively.

4. Classification method

The classification method we propose in this work makes use exclusively of dictionaries for sparse coding, and can be divided in two phases: the denoising phase and the discrimination phase. For convenience, a block diagram outlining the classification method that we discuss next is displayed in Figure 3.

In the first phase the goal is to recover as much oscillations as possible from all glitches while keeping the spurious oscillations to a reasonable low level (let us call them 'parent' reconstructions). To this end we use three trained dictionaries, one per waveform morphology, with a constant value for the regularization parameter λ_{tr} (the subindex 'tr' stands for 'transformation' as it refers to the sparse encoding of the signals). Each denoising dictionary is composed of atoms of the same length, which are shorter than the length of the input signals. Because of this, part of the atoms are fragments of glitches, and none of them are aligned in any way. Therefore, each dictionary is initialized and trained with a large number of patches randomly extracted from the corresponding training set, with the only condition that the patches include a minimum number of nonzero bins, which we choose to be around $\frac{1}{4}$ of the atom's length. Since we do not know the glitch morphology yet, each glitch needs to be reconstructed by all denoising dictionaries.

After this phase we end up with three denoising reconstructions per unknown glitch. Ideally, each dictionary should be used with an optimal regularization parameter λ_{opt} so

that the dictionary would yield a relatively clean reconstruction only if the input signal contained a glitch of the same morphology. However, in practice some factors like a high resemblance between some morphologies or waveforms with extremely low SNR make impossible for our dictionaries to achieve an acceptable level of glitch discrimination. To overcome this issue we take a different approach with respect to our previous work [15]; we redefine the value of λ_{opt} to be that which gives us the best classification results with our set of testing glitches. As shown in section 5.1, this new value happens to be quite lower than the original λ_{opt} , making all dictionaries to yield relatively faithful reconstructions regardless of the morphology of the glitch. However, given a noisy signal containing (for example) a SG glitch, we expect the reconstruction from the SG dictionary to be closer to a perfect SG waveform than those from the G and RD dictionaries, since the first one is made up of patches of perfect glitches of the same morphology. In other words, for each glitch we need to determine which denoising dictionary produces the reconstruction closest to an ideal glitch of any of the three morphologies.

The goal of the second phase is to perform the aforementioned discrimination using untrained dictionaries, which we call classification dictionaries, composed of atoms containing whole centered glitches. Each of these dictionaries is nothing more than a collection of random examples of “perfect” glitches belonging to the same waveform morphology. With them, every parent will be “reconstructed” using as few atoms as possible, so that instead of generating a faithful reconstruction, each dictionary will choose the few atoms that most “resemble” to the parent glitch but maintaining the morphology of the dictionary. The number of atoms used, which will be referred to as ‘nonzeros’ from now on, is a hyperparameter which will be fixed instead of λ_{opt} , and whose optimum value needs to be determined. After this procedure, we are left with three new reconstructions (let us call this new reconstructions “children”) per parent glitch, with a total of 12 reconstructions for each unknown glitch, forming a ternary tree structure as shown in figure 2.

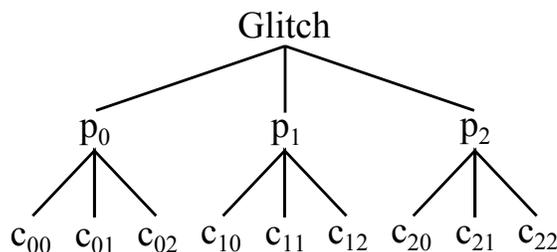


Figure 2. Tree diagram of the reconstructions obtained for each original glitch, where p_j stands for parent glitches, and c_{ji} for children glitches. The numbering corresponds to the three morphologies ordered as follows: sine Gaussian ($i, j = 0$), Gaussian ($i, j = 1$), Ring-Down ($i, j = 2$).

Finally, we use the SSIM estimator to calculate the value of the DSSIM index between each child and its parent. In this part of the algorithm we do not make use

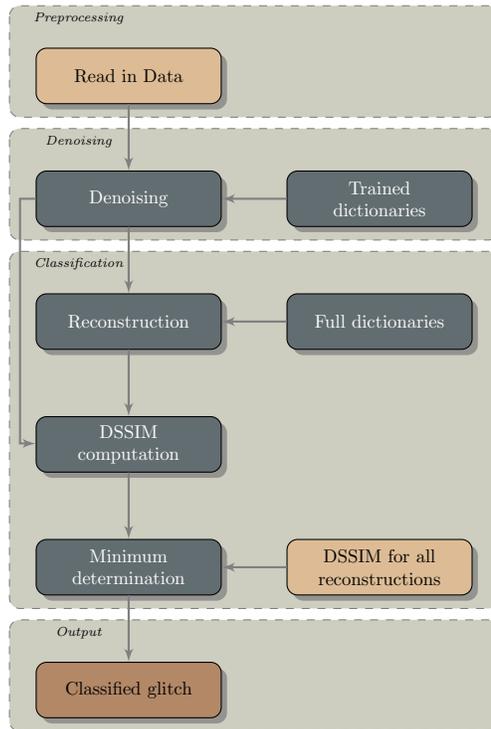


Figure 3. Block diagram outlining the classification method.

of the MSE estimator because it does not take into account the structural information. The predicted morphology will be that of the dictionary which produced the least total DSSIM value,

$$i_{\text{dict}} = \arg \min_i \prod_j \text{DSSIM}(p_j, c_{ji}), \quad (14)$$

where i_{dict} is the index of the dictionary. If a child c_{ji} is not reconstructed (i.e. it is not recognized) the DSSIM will take the worst value, and if none of the children or parents are reconstructed, then we will consider the glitch lost (which means it will not be classified whatsoever).

5. Tests and results

5.1. Parameter optimization

We first optimize the parameters of the denoising dictionaries, namely the regularization parameter of the learning step λ_{learn} , the number of atoms m and their lengths n , and the regularization parameter of the reconstruction step λ_{tr} . We then next optimize the parameters of the classification dictionaries, namely the number of atoms and the number of nonzero coefficients to use for the reconstructions of the second phase. For each parameter we test several different values, reconstructing all 100 glitches from the

validation set and computing the average MSE (similar results are obtained when using the DSSIM index, since now we are not comparing between different morphologies). We consider the optimum value of a parameter to be the one that yields the lowest average MSE.

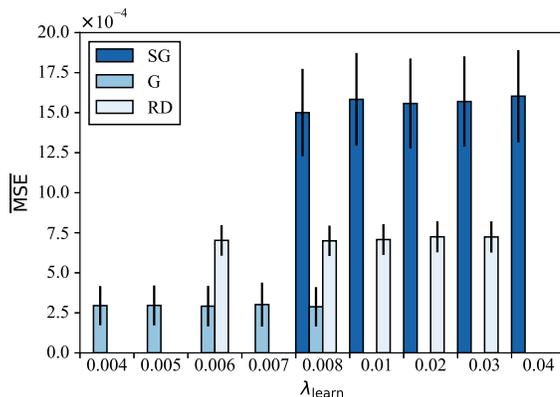


Figure 4. Histogram of the average MSE for all validation signals reconstructed by the denoising dictionaries trained with different values of λ_{learn} . Each colour represents a different dictionary (i.e. a different glitch morphology). The standard deviation of each average is represented by a black line, and downscaled by a factor 10 for SG and RD.

We start by studying how the regularization parameter of the dictionary learning step, λ_{learn} , affects the reconstructed signals. As a starting point we choose 1024 atoms for the SG and RD dictionaries with a length of 512 samples, and 256 atoms for the G dictionary with a length of 128 samples. Those lengths are close to the validation set’s average. An initial test shows that we obtain trained atoms without noise only when λ_{learn} is between a certain range, which is different for each morphology as shown in figure 4. The absence of a histogram for certain values of λ_{learn} indicates that the dictionary does not yield a reconstruction for that value. However, there is no significant variation in the quality of the reconstruction for different values of λ_{learn} (once a reconstruction has been obtained). This can be seen by comparing the histogram values with their standard deviation. Therefore, we choose to use a mean value for each dictionary: 0.02 for SG, 0.006 for G, and 0.01 for RD.

Next, we test the effect of using different atom lengths, $n = 64, 128, 256$ and 512 for SG and RD dictionaries, and $n = 16, 32, 54$ and 128 for G dictionary (Gaussian glitches are shorter than the other two). The results displayed in figure 5 show that shorter atoms produce worse reconstructions (i.e. larger values of the MSE index). This is due to the fact that if atoms are too short they are more sensible to noise oscillations, becoming hard to recognize low frequency glitches. The best results are achieved with the two greatest lengths of each dictionary. Therefore, we choose to use a length of 256 for SG and RD dictionaries, and 128 for the G dictionary, which gives a good trade-off between quality and performance.

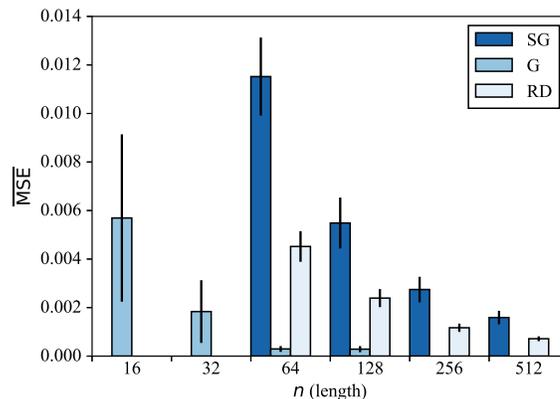


Figure 5. Histogram of the average MSE for all validation signals reconstructed by the denoising dictionaries with different atoms' lengths. Each colour represents a different dictionary (i.e. a different morphology). The standard deviation of each average is represented by a black line, and downscaled by a factor 10 for SG and RD.

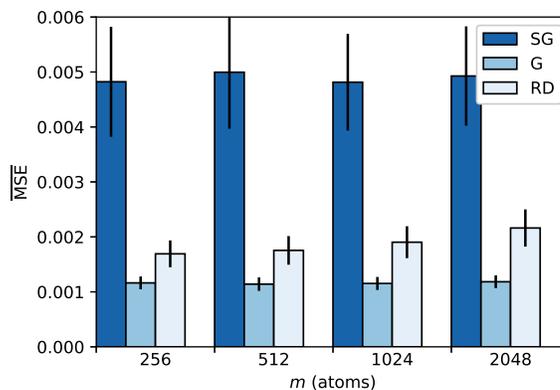


Figure 6. Histogram of the average MSE for all validation signals reconstructed by the denoising dictionaries with different number of atoms. Each colour represents a different dictionary (i.e. a different morphology). The standard deviation of each average is represented by a black line, all downscaled by a factor 10.

In general, the larger the number of atoms the better the results should be. To check this, we carry out a test with $m = 256, 512, 1024$ and 2048 atoms. We set all their lengths to $n = 128$ (only for this test) so that dictionaries remain over-completed. As can be seen in figure 6 there is no clear improvement; even 256 atoms are more than enough for the dictionaries to recognize their own waveforms, which explains why increasing their number does not provide better results anymore. Therefore, we use the smallest number of atoms needed for the dictionaries to remain overcomplete: $m = 512$ for SG and RD, and 256 for Gaussian.

Every glitch signal to be denoised has a specific optimum value of the regularization parameter, λ_{opt} , used in its sparse reconstruction (see [15] for details on the computation of λ_{opt}). However, our initial tests showed that although using a mean value of λ_{opt}

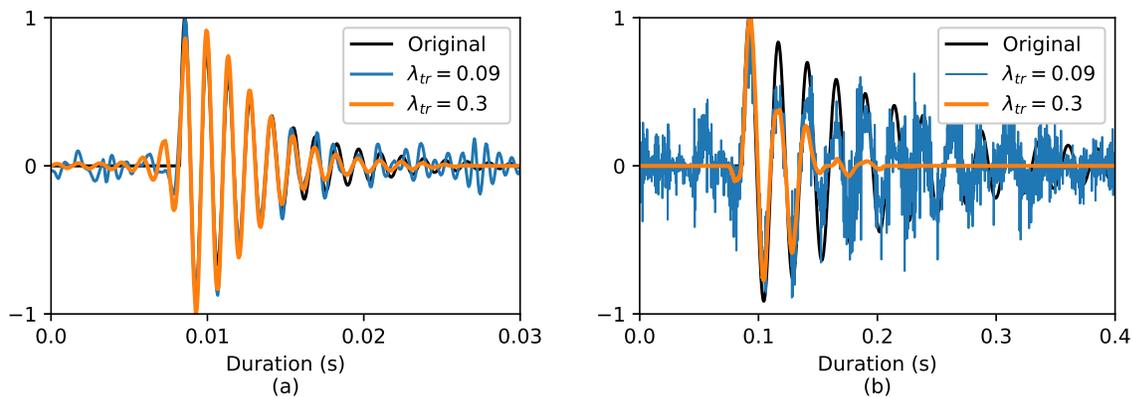


Figure 7. Example of two denoised RD glitches which were injected into Gaussian noise at 20 SNR. Both plots show the original glitch (before being injected into noise), and two reconstructions using different values for λ_{tr} , namely 0.3, which is close to the average optimal value and yields almost noise-free reconstructions, and 0.09, which improves the classification properties of the dictionaries but introduces more spurious oscillations.

(estimated from the validation set) yielded the most accurate reconstructions, it did not offer the best classification results. This is because the distinctive oscillatory features of a large number of glitches of our sample cannot be fully recovered with the optimum values of the regularization parameter, as e.g. happens with the RD glitch shown in figure 7(b). Instead, more oscillations can be recovered by using a smaller value, λ_{tr} , making the glitch morphology more easy to differentiate (and, hence, improving glitch classification) but at the expense of a poorer denoising. Nevertheless, those glitches that could be completely reconstructed with the optimum value (like the example shown in figure 7(a)) can still be partly reconstructed using λ_{tr} . Therefore, we carry out a test to find a new (average) optimum value for the regularization parameter of the transformation. We use the same value for all three dictionaries in order to spend a reasonable amount of computational time. More precise classifications may be achieved by looking for individual values, although judging by our tests we would not expect a big improvement.

From all tested values, only two extremes and the optimum result are shown in the confusion matrices of figure 8. All Gaussian glitches are always correctly classified due to their simplicity compared to SG and RD glitches. The latter two are sometimes mismatched because of their similarity. When the value of λ_{tr} of the denoising dictionaries is too low (figure 8a) more RD glitches are predicted as SG glitches, which means that the SG dictionary is more capable of reproducing the RD morphology than otherwise. On the other hand, when λ_{tr} is too high (figure 8c) some glitches are lost (because the reconstruction capability of the dictionaries is reduced), and more SG glitches are mistaken as RD glitches because the number of irregular reconstruction increases (improving the chances of the RD dictionary to provide better reconstructions

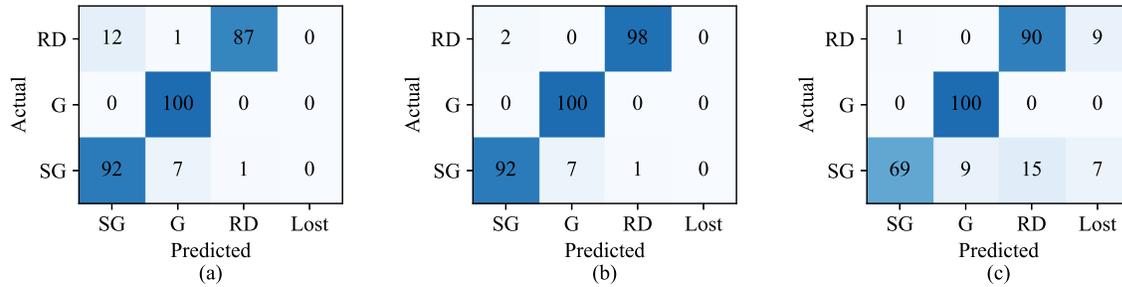


Figure 8. Confusion matrices of the classification results for three different values of the regularization parameter of the transformation of the denoising dictionaries: $\lambda_{tr} = 0.05$ (a), 0.09 (b), 0.5 (c). Rows correspond to the actual morphology of validation glitches, and columns to the morphology predicted by our classification dictionaries. We chose as a starting values 256 atoms for the classification dictionaries, and 4 nonzero coefficients for their reconstructions.

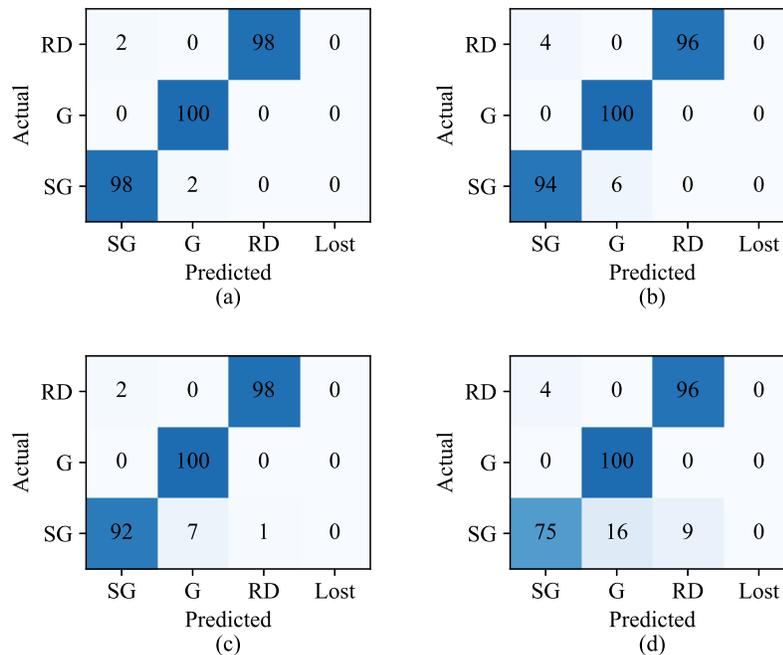


Figure 9. Confusion matrices of the classification results using different numbers of nonzero coefficients: 1 (a), 2 (b), 4 (c), and 8 (d). Rows correspond to the actual morphology of validation glitches, and columns to the morphology predicted by classification dictionaries. We keep using 256 atoms as a starting value for the classification dictionaries.

of the denoised glitches). The best configuration found is $\lambda_{opt} = 0.09$ (figure 8b) for all denoising dictionaries, with no glitches lost and only a few and fairly well balanced mismatches.

For the classification dictionaries we need to choose the number of nonzero components we will be using for the second reconstruction step. The goal of these

dictionaries is to discriminate between the different morphologies by reconstructing again the denoised glitches and finding the best fit. Hence we are not interested in using too many atoms in each reconstruction (which eventually would lead to too similar reconstructions from different dictionaries). In fact, the results plotted in figure 9 show that the best discrimination level is achieved with only 1 nonzero coefficient (figure 9a), while the worst results are obtained with 8 nonzero coefficients (figure 9d) as expected.

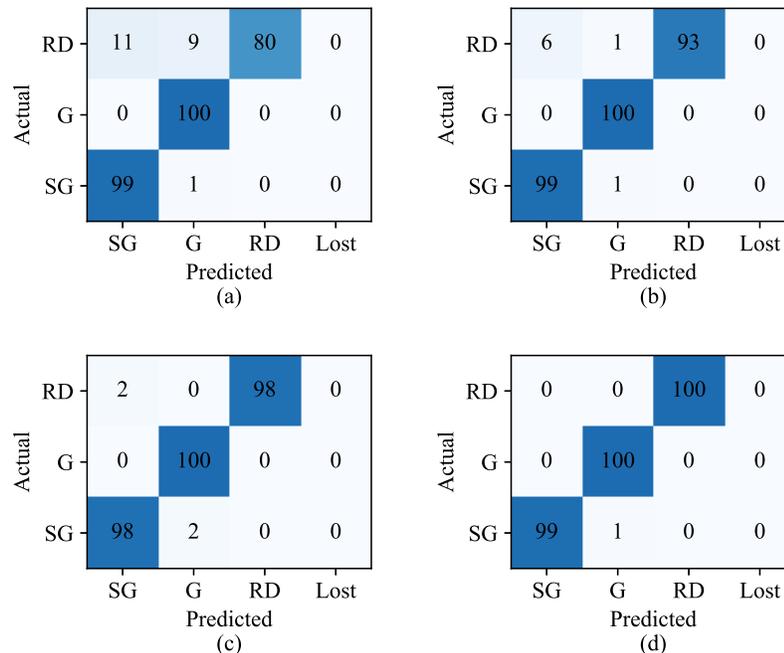


Figure 10. Confusion matrices of the classification results using different numbers of atoms: 64 (a), 128 (b), 256 (c), and 512 (d). Rows correspond to the actual morphology of validation glitches, and columns to the morphology predicted by classification dictionaries.

Finally, since for each reconstruction we are using only 1 atom (which in this case corresponds to a whole glitch), the classification dictionaries' effectiveness will increase with their number of atoms, as can be seen in results of figure 10. From these, we consider 256 atoms to be a good trade-off between accuracy and performance, although sensibly better results could be achieved by using even more.

5.2. Final test

So far we have evaluated every configuration using a few testing signals, which was enough to make comparisons. However, for the final configuration almost all glitches were classified correctly, not leaving enough data to evaluate whether the dictionaries are well balanced between them. Therefore we repeat the case study using the testing set of 3000 glitches, with the same parameters as before (summarized in Table 2). In addition, following [9], for each glitch we now choose a random SNR linearly distributed between 1 and 400.

Table 2. Parameter values of the final configuration for the denoising (*den*) and classification (*clas*) dictionaries.

Dictionary	λ_{learn}	a_{den}	n_{den}	λ_{tr}	nonzero	a_{clas}
sine Gaussian	0.02	512	256	0.09	1	256
Gaussian	0.006	256	128	0.09	1	256
Ring-Down	0.01	512	256	0.09	1	256

The results of this test are shown in figure 11. In total, 3000 glitches have been processed, with 2879 (96%) correctly classified. The Gaussian dictionary recognized all but 1 of its glitches, the one with the lowest SNR, while 92 SG and 28 RD glitches were not correctly classified. The great accuracy of the G dictionary is due to the difference between the Gaussian morphology and the other two. SG and RD glitches can be relatively easy to reproduce using a few Gaussian atoms but not the other way around, because most of their atoms contain more than one oscillation. Hence, SG and RD glitches are more likely to be misclassified.

Actual	RD	21	7	972	0
	G	0	999	1	0
	SG	908	26	66	0
		SG	G	RD	Lost
		Predicted			

Figure 11. Confusion matrix of the classification results of the final configuration. Rows correspond to the actual morphology of validation glitches, and columns to the morphology predicted by classification dictionaries.

This, however, does not explain the differences found in the number of misclassified glitches (also called false positives) between SG and RD dictionaries. For the sake of understanding the results we analyze how the classification method behaves for misclassified glitches depending on their SNR and frequency. The results of this analysis are shown in figure 12. The left panel corresponds to SG glitches misclassified as G or RD and the right panel to RD glitches misclassified as G or SG. In figure 12 we can see roughly the same ratio of false positives within all the range of SNR values, except for the lowest SNR (below 10) where most mismatches occur (as expected). These false positives come mostly from the Gaussian dictionary; at such low SNR both SG and RD dictionaries yield more irregular reconstructions because of noise fluctuations, while Gaussian reconstructions are cleaner, making them more likely to be identified as such. Therefore, the SNR may unbalance the number of false positives only in favor of the Gaussian morphology.

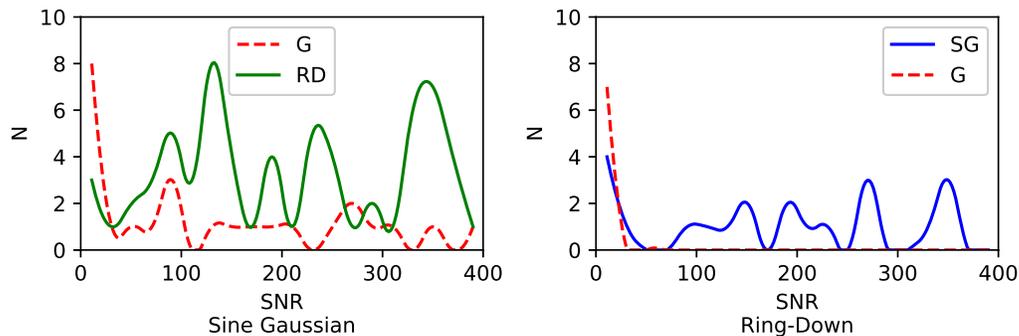


Figure 12. Distribution of misclassified SG glitches (left) and RD glitches (right) as a function of their SNR. Each line represents a (wrongly) predicted morphology.

On the other hand, the results for the distribution in terms of frequency, shown in figure 13, are more inhomogeneous. The left panel indicates that SG glitches are misclassified as G and RD more often at high frequencies, while RD glitches are misclassified mainly as SG at low frequencies. High frequency SG glitches have in general shorter durations (see Eq. (10)), which combined with noise fluctuations makes them more similar to short Gaussian or RD glitches, explaining the high frequency peaks in the left plot and the greater number of wrongly predicted RD glitches compared to that of SG glitches. At the same time, glitches with lowest frequencies are generally much larger than the atoms' window (about an order of magnitude), so they need to be split in multiple samples, each of them containing usually only a few oscillations. Individually, these windows are almost indistinguishable from both SG and RD's low frequency atoms in terms of morphology, which explains the similar increase in misclassified glitches in both plots at lower frequencies.

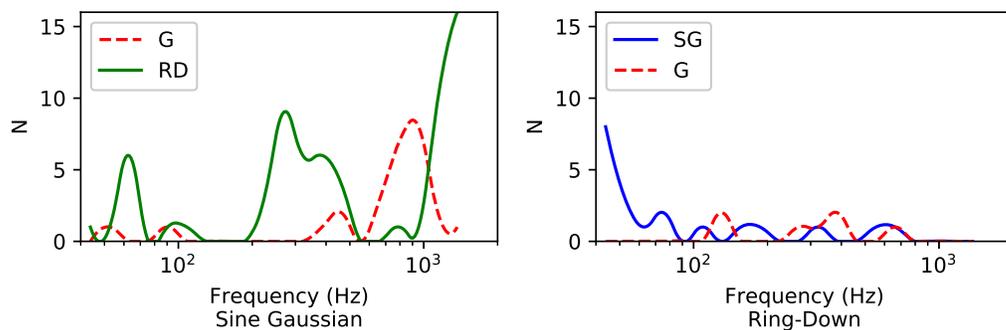


Figure 13. Distribution of misclassified SG glitches (left) and RD glitches (right) as a function of their frequency. Each line represents a (wrongly) predicted morphology.

In summary, our method was able to classify by morphology 96% of testing glitches with a wide range of parameters and SNR. It succeeded in recognizing Gaussian glitches, and achieved a good discrimination level between SG and RD glitches, being slightly

unbalanced mainly due to their morphological dissimilarities. More precision may be achieved by using more atoms for the untrained dictionaries, or even splitting each denoising dictionary into several ones for different frequency intervals.

6. Conclusions

In this paper we have introduced a new method for the classification of transient noise signals (or glitches) by their waveform morphology in advanced GW interferometers. The method uses learned dictionaries (a supervised machine learning algorithm) for the denoising, and untrained dictionaries for the final sparse reconstruction and classification. To test the accuracy of our method we have used a set of simulated glitches embedded in non-white Gaussian noise, simulating the background noise of advanced LIGO in the proposed broadband configuration.

Using a data set consisting of 3000 glitches divided into three different waveform morphologies with a large range of parameters, the method has shown a 96% classification accuracy. Furthermore, its performance has been found to barely decrease with the SNR, down to low SNR values (~ 10). Its main limitation appears at extreme frequencies; most of misclassified glitches have the highest or lowest frequencies. In the case of low SNR, it is challenging to discriminate the glitches from the noise and the algorithm tends to misclassify them. Something similar explains what happens for low and high frequencies. As the detector sensitivity is not flat in frequency, those frequencies are more affected by noise than the middle ones. Nevertheless, our study with simulated glitches and Gaussian noise shows that dictionaries are successful at discriminating even the two most similar morphologies of our sample.

There are possible extensions of this study we plan to undertake next. Certainly, the accuracy of our method can be significantly improved by using more untrained atoms in order to expand the model population (especially at extreme frequencies), as the results of figure 6 indicate. It may be interesting to study by how much the accuracy could be improved by using several learned dictionaries for each waveform morphology, in order to split the frequency range into smaller intervals. We also plan to implement additional methods to perform the LASSO algorithm more efficiently, since the denoising phase of our current approach is the most computationally expensive. Reconstructing all 3000 glitches, each one inside one patch of 16384 samples, by all three learned dictionaries takes about 36 hours with an AMD Phenom II x4 processor and 12 GB of RAM. In addition, in this work we have assumed all signals to be glitches and their positions to be already known. Our method does not provide a trigger for glitches, therefore a real GW signal or even a pure noise sample could be classified as a glitch. The first case can be avoided by removing coincident signals before the classification. For the latter, it may be worth analyzing whether a previous denoising phase using learned dictionaries with lower sensitivities could be used as a trigger layer before employing our current method.

This work constitutes a preliminary step before assessing the performance of

dictionary-learning methods with actual detector glitches. A natural next step is to test our method employing glitches from Advanced LIGO's first observing run (O1) whose waveform morphologies have been already classified [11]. In a longer time frame we plan to combine learned dictionaries for both glitches and GW signals and try to discriminate if a detector trigger can be classified as a glitch or a signal. This would allow to compute the probabilities of false alarm (identifying a glitch as a true signal) and of losing a signal (either because it has been classified as a glitch or as background noise). Such a study appears necessary to explore the usefulness of dictionary-learning algorithms for GW data analysis.

Acknowledgments

Work supported by the Spanish MINECO (grants AYA2015-66899-C2-1-P and MTM2014-56218-C2-2-P), by the Generalitat Valenciana (PROMETEOII-2014-069), and by the European Gravitational Observatory (EGO-DIR-51-2017).

References

- [1] Abbott B P, Abbott R, Abbott T D, Abernathy M R, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X and et al 2016 *Physical Review Letters* **116** 061102 (*Preprint* 1602.03837)
- [2] Abbott B P, Abbott R, Abbott T D, Abernathy M R, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X and et al 2016 *Physical Review Letters* **116** 241103 (*Preprint* 1606.04855)
- [3] Abbott B P, Abbott R, Abbott T D, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X, Adya V B and et al 2017 *Physical Review Letters* **118** 221101 (*Preprint* 1706.01812)
- [4] Abbott B P, Abbott R, Abbott T D, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X, Adya V B and et al 2017 *Astrophys. J. Lett.* **851** L35 (*Preprint* 1711.05578)
- [5] Abbott B P, Abbott R, Abbott T D, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X, Adya V B and et al 2017 *Physical Review Letters* **119** 141101 (*Preprint* 1709.09660)
- [6] Abbott B P, Abbott R, Abbott T D, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X, Adya V B and et al 2017 *Phys. Rev. Lett.* **119** 161101 (*Preprint* 1710.05832)
- [7] Abbott B P, Abbott R, Abbott T D, Abernathy M R, Acernese F, Ackley K, Adams C, Adams T, Addesso P, Adhikari R X and et al 2018 *Living Reviews in Relativity* **21** 3 (*Preprint* 1304.0670)
- [8] Martynov D V, Hall E D, Abbott B P, Abbott R, Abbott T D, Adams C, Adhikari R X, Anderson R A, Anderson S B, Arai K and et al 2016 *Phys. Rev. D* **93** 112004 (*Preprint* 1604.00439)
- [9] Powell J, Trifirò D, Cuoco E, Heng I S and Cavaglià M 2015 *Classical and Quantum Gravity* **32** 215012 (*Preprint* 1505.01299)
- [10] Powell J, Torres-Forné A, Lynch R, Trifirò D, Cuoco E, Cavaglià M, Heng I S and Font J A 2017 *Classical and Quantum Gravity* **34** 034002 (*Preprint* 1609.06262)
- [11] Zevin M, Coughlin S, Bahaadini S, Besler E, Rohani N, Allen S, Cabero M, Crowston K, Katsaggelos A K, Larson S L, Lee T K, Lintott C, Littenberg T B, Lundgren A, Østerlund C, Smith J R, Trouille L and Kalogera V 2017 *Classical and Quantum Gravity* **34** 064003 (*Preprint* 1611.04596)
- [12] Mukund N, Abraham S, Kandhasamy S, Mitra S and Philip N S 2017 *Phys. Rev. D* **95** 104059 (*Preprint* 1609.07259)

- [13] George D and Huerta E A 2018 *Physics Letters B* **778** 64–70 (*Preprint* 1711.03121)
- [14] Razzano M and Cuoco E 2018 *Classical and Quantum Gravity* **35** 095016 (*Preprint* 1803.09933)
- [15] Torres-Forné A, Marquina A, Font J A and Ibáñez J M 2016 *Phys. Rev. D* **94** 124040 (*Preprint* 1612.01305)
- [16] Chen S S, Donoho D L and Saunders M A 2001 *SIAM Review* **43** 129–159
- [17] Tibshirani R 1996 *Journal of the Royal Statistical Society. Series B (Methodological)* 267–288
- [18] Wang Z, Bovik A C, Sheikh H R and Simoncelli E P 2004 *IEEE Transactions on Image Processing* **13** 600–612
- [19] Efron B, Hastie T, Johnstone I and Tibshirani R 2004 *The Annals of Statistics* **32** 407–499 (*Preprint* math/0406474)
- [20] Mairal J, Bach F, Ponce J and Sapiro G 2009 *Proceedings of the 26th International Conference on Machine Learning* 1–8 ISSN 0016450X (*Preprint* 0908.0050) URL <http://dl.acm.org/citation.cfm?id=1553463>
- [21] Torres A, Marquina A, Font J A and Ibáñez J M 2014 *Phys. Rev. D* **90** 084029 (*Preprint* 1409.7888)