Searching for gravitational waves from the coalescence of high mass black hole binaries

Lau Ka Tung
Department of Physics, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong, China

Mentors: Surabhi Sachdev, Tjonnie Li, Kent Blackburn, and Alan Weinstein
LIGO Laboratory, California Institute of Technology, Pasadena, California 91125, US
(Dated: September 26, 2015)

The coalescence of binary black holes is a promising class of sources of gravitational waves which can be detected by ground-based gravitational-wave detectors. The frequencies of gravitational waves generated by the coalescence of stellar-mass black holes lie in the advanced LIGO (aLIGO) frequency band. aLIGO uses a search pipeline called GstLAL to search for coalescence signals from the detector output. This search pipeline uses matched filtering to compute the signal-to-noise ratio (SNR) and χ^2 value of the detector signal. The maxima in the SNR time series which have SNRs higher than a threshold are known as triggers. The challenge is to discriminate triggers induced by gravitational waves from those induced by noise based on the output of the matched filter. In this project, we investigated the use of machine learning to achieve this goal. Random Forest of Bagged Decision Tree (RFBDT) was used as the learning algorithm. Real detector data and simulated signals were used as input to the classifier for training and evaluation. We tuned the RFBDT parameters and the feature vector in order to optimize the performance of the classifier. We also implemented the RFBDT in the GstLAL pipeline.

I. GRAVITATIONAL WAVES AND THEIR DETECTION

A. Gravitational waves

General Relativity predicts that changes gravitational fields produces ripples of curvature of spacetime. Gravitational waves carry out energy and angular momentum away from the source and propagate at speed of light. When a gravitational wave passes through, it causes stretching and squeezing between test masses. We have not detected the stretching and squeezing effect from gravitational waves directly, but we have indirect evidence for the existence of gravitational waves.

In General Relativity, gravity is described by the spacetime curvature. The relationship between spacetime curvature and energy-momentum is governed by the Einstein field equations

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu}, \tag{1}$$

where $R_{\mu\nu}$ is the Ricci curvature tensor, R is the scalar curvature, $g_{\mu\nu}$ is the metric tensor, G is the gravitational constant, c is the speed of light and $T_{\mu\nu}$ is the stress-energy tensor. Einstein field equation illustrates that mass curves spacetime, and curvature dictates the flow of mass.

Considering a system which is far from the source such that $T_{\mu\nu} = 0$, the Einstein's equation becomes

$$R_{\mu\nu} = 0. \tag{2}$$

We consider the space-time as a small perturbation to the Minkowski space-time, the metric tensor can be written as

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}, \tag{3}$$

where

$$\eta_{\mu\nu} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and $h_{\mu\nu} \ll 1$. Under these assumptions, Eq. 3 admits a transversely-propagating wave solution, which travels at the speed of light and has two independent degrees of freedom, which is known as the polarizations. If we choose our coordinates such that the wave travels in the +z direction, we can write the solution in terms of the metric as

$$g_{\mu\nu} = \eta_{\mu\nu} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & h_+ & h_\times & 0 \\ 0 & h_\times & -h_+ & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{4}$$

where h_+ and $h_\times$ are functions of time and space which satisfy the wave equation

$$\left(\nabla^2 - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right) h = 0, \tag{5}$$

where $h = h_+$ or $h_\times$. This is the description of gravitational wave in General Relativity, and it travels at the speed of light.

Eq. 5 shows that gravitational waves have two polarizations, which are plus polarization h_+ and cross polarization $h_\times$. FIG. 1 shows how an initially circular array of test masses will move in response to a gravitational wave.

In 1974, Joseph Taylor and Russell Hulse discovered the first binary pulsar [6]. The binary consists of a neutron star and a pulsar, the pulsar emits electromagnetic pulse regularly towards the Earth. After more than a

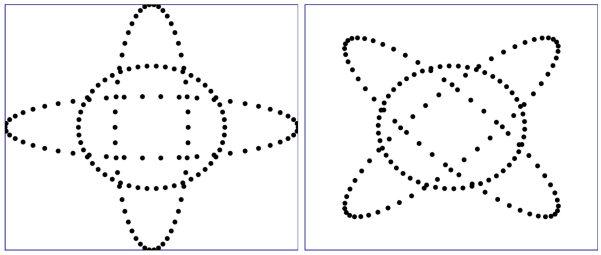


FIG. 1. Plus polarization h_+ (left) and cross polarization $h_\times$ (right) of gravitational waves [1].

decade observation, Taylor and Hulse discovered an orbital decay from the shortening of the period of the pulse. The energy loss of the binary pulsar matches the loss due to gravitational radiation. This provides an evidence for the existence of gravitational waves.

B. Laser Interferometer Gravitational-wave Observatory (LIGO)

LIGO is a large experiment aiming to detect the stretching and squeezing of spacetime caused by passing gravitational waves directly. There are two observatories in the United States, one is the LIGO Livingston Observatory located in Livingston, Louisiana, another one is the LIGO Hanford Observatory located next to Richland, Washington.

LIGO is a Michelson interferometer which can detect a small change between two arms. A laser beam is emitted to the beam splitter and split the light to the two arms. If two arms have the same length, then the light bounces back to the splitter with a destructive interference, and therefore, no light will be detected by the photosensor. However, if there is a gravitational wave passing through the detector and causes a length difference between two arms, some light can travel to the photosensor and the gravitational wave signal is detected. FIG. 2 shows a simplified picture of a LIGO detector.

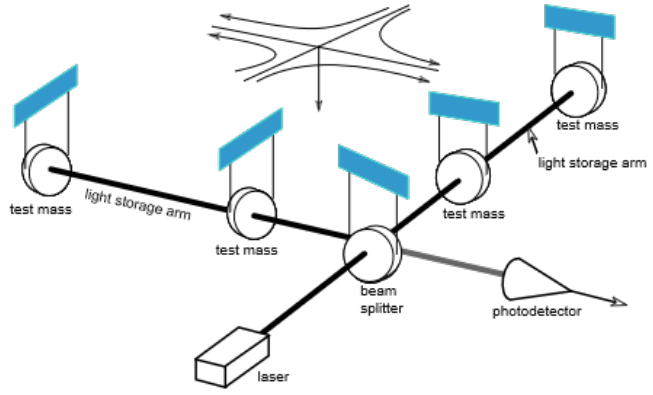


FIG. 2. A simplified schematic of a LIGO interferometer [1].

In the operation of initial LIGO (iLIGO), no gravitational waves were detected. After a five years upgrade, Advance LIGO (aLIGO) is planned to begin a science run in late 2015. Seismic noise, thermal noise and shot noise are reduced in aLIGO as shown in FIG. 3. The sensitive of aLIGO is expected to be improved by 10 times with respect to iLIGO.

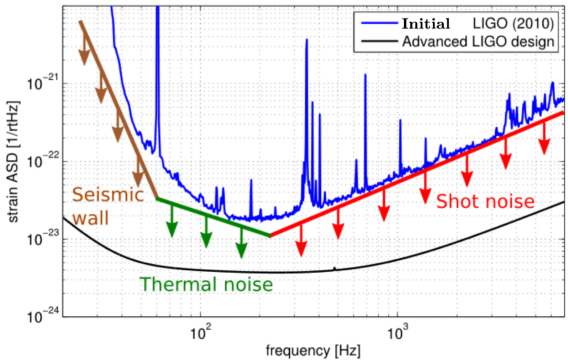


FIG. 3. Expected noise curve of aLIGO [14]. Advanced LIGO is expected to increase the sensitivity by reducing seismic, thermal and shot noise.

C. Compact binary coalescence

The frequency band of LIGO is about 10Hz - 10^3Hz. The frequency of gravitational waves from the coalescence of compact binaries, such as binary neutron star, binary black hole and neutron star black hole binary, lies in the frequency band of LIGO, we expect to detect the gravitational wave signals from this astronomical process. Coalescence consists of three stages, which is known as inspiral, merger and ringdown. When the gravitational waves carry away energy and angular momentum from the binary, the result in orbital decay and decreasing orbital period is known as inspiral. When the black holes get close enough, they will merge into a single black hole. After the merge, any distortion will dissipate in form of gravitational waves, which is known as ringdown.

The inspiral process of binary black hole can be described in quasi-Newtonian limit. From General Relativity, the total energy loss can be written in quadrupole approximation as

$$\frac{dE}{dt} = -\frac{64}{5}\frac{G^4}{c^5}\frac{\mu^2 M^3}{r^5}, \tag{6}$$

where $M = m_1 + m_2$ is the total mass of the binary, $\mu = m_1 m_2/M$ is the reduced mass and r is the orbital separation. By virial theorem, the energy of the system can be written as

$$E = -\frac{1}{2}\frac{G m_1 m_2}{r}, \tag{7}$$

after differentiate both sides with respect to time t, the orbital decay can be described by

$$\frac{dr}{dt} = \frac{1}{2}\frac{r^2}{Gm_1m_2}\frac{dE}{dt}. \tag{8}$$

Substitute Eq. 7 into Eq. 9, the time derivative of orbital radius can be written as

$$\frac{dr}{dt} = -\frac{64}{5}\frac{G^3}{c^5}\frac{m_1m_2M}{r^3}, \tag{9}$$

and the evolution of the orbital separation can be obtained after integration

$$r(t) = \left({r_o}^4 - \frac{256}{5}\frac{G^3}{c^5}m_1m_2Mt\right)^{1/4}. \tag{10}$$

The stretching and squeezing effect caused by gravitational waves between test masses can be described by the strain amplitude $h = \Delta L/L$. The strain amplitude h generated by the source at distance D is related with the orbital separation with

$$h(t) = \left(\frac{2G\mu}{c^2D}\right)\left(\frac{2GM}{c^2r(t)}\right)\cos\left(\Phi(t)\right), \tag{11}$$

where

$$\Phi(t) = \int 2\pi f_{GW}(t)dt. \tag{12}$$

$f_{GW}(t)$ can be expressed in the quadrupole approximation

$$f_{GW}(t) = \frac{c^3}{8\pi GM}\left(\frac{c^3\eta}{5GM}(t_c - t)^{-3/8}\right), \tag{13}$$

where $\eta = \mu/M$ and t_c is the time for the orbit to reach the intermost stable circular orbit (ISCO). This is the inspiral part of gravitational waves from the compact binary coalescence under quasi-Newtonian limit. We take the innermost stable circular orbit (ISCO),

$$f_{ISCO} = \frac{c^3}{6\sqrt{6}\pi GM} \tag{14}$$

as the cutoff frequency for the post-Newtonian approximation. For the stage of merger and ring-down, the post-Newtonian approximation breaks down since the relativistic effects are required to be taken in consideration. Examples of time domain and frequency domain waveforms with varying parameters can be found in appendix.

II. SEARCHING STRATEGIES FOR GRAVITATIONAL WAVES FROM COALESCENCE OF BINARY BLACK HOLE

GstLAL is a search pipeline which is used by aLIGO in searching for gravitational-wave signals from coalescence of compact binary. We aim to join gravitational wave observations with electromagnetic wave observations. GstLAL is a low-latency search pipeline which can send out astronomical alert to the electromagnetic telescopes within a few minutes. In order to achieve the low-latency search, several technique is used to reduce the computational resource. A detailed explanation of the pipeline can be found in Ref. [4, 5].

GstLAL uses matched filtering to find the coalescence signals buried in noise. The search pipeline first computes signal-to-noise ratio (SNR) of the detector output with all the waveform templates in the bank of templates. The maxima in SNR time series which is higher than a threshold is known as a trigger. The matched filtering is the optimal method to find the signal if the signal is buried in stationary Gaussian noise. However, LIGO data contains non-stationary noise, which are known as glitches. The glitches can have a high SNR and cause a false alarm in analysis, a χ^2 veto is used to reject the false signals.

A. Matched filtering

Matched filtering is a method to extract the signals from a noisy data by comparing the detector output with a predicted waveform template. The matched filtering method is the optimal filter to get the largest signal-to-noise ratio in stationary Gaussian noise. Consider the detector output signal $s(t) = n(t) + q(t)$ where $n(t)$ is the noise and $q(t)$ is the gravitational wave signal, we can compute the cross correlation between the detector output with a template h

$$c(\tau) = \int_{-\infty}^{\infty} s(t)h(t+\tau)dt. \tag{15}$$

We can transform it in to frequency domain, such that

$$c(\tau) = \int_0^{\infty} \tilde{s}(f)\tilde{h}^*(f)e^{2\pi i f\tau}df, \tag{16}$$

where $\tilde{s}(f)$ is the Fourier transform of $s(t)$ and $\tilde{h}^*(f)$ is the complex conjugate of the Fourier transform of $h(t)$. In order to whiten the signals, the correlation is weighted by the power spectral density $S_n(f) = \langle\tilde{n}(f)\tilde{n}^*(f')\rangle$. The matched filter output is

$$x(\tau) = 4\mathrm{Re}\int_0^{\infty} \frac{\tilde{s}(f)\tilde{h}^*(f)e^{2\pi i f\tau}}{S_n(f)}df. \tag{17}$$

Since the waveform contains some unknown parameters such as amplitude and the coalescence phase. The unknown phase can be searched over by forming the complex matched filter outer

$$z(\tau) = x(\tau) + iy(\tau) = 4\int_0^{\infty} \frac{\tilde{s}(f)\tilde{h}^*(f)e^{2\pi i f\tau}}{S_n(f)}df. \tag{18}$$

The waveform templates are constructed for systems with an effective distance $D_{\mathrm{eff}} = 1Mpc$. The normalized constant for computing the SNR, which is the measure of

the sensitivity of the detector, is

$$\sigma = \sqrt{4\int_0^\infty \frac{|\tilde{h}(f)|^2}{S_n(f)} df}. \tag{19}$$

The time series of SNR is defined as

$$\rho(\tau) \equiv \frac{|z(\tau)|}{\sigma}. \tag{20}$$

B. χ^2 veto

To reject the false alarm, χ^2 method is used to distinguish the true signals from glitches. χ^2 compares the detector output with the waveform template.

Consider the detector output signal $s(t) = n(t) + q(t)$ illustrated in FIG. 4, where $n(t)$ is the noise and $q(t)$ is the gravitational wave signal, and the template waveform $h(t)$. We divide the frequency range of integration into a finite number of bins $f_k \leq f \leq f_{k+1}$, where $k = 1, ..., p$. We define the contribution to the matched filtering statistic coming from the k-th bin by

$$z_k \equiv \langle s, h\rangle_k \equiv 2\int_{f_k}^{f_(k+1)} \left[\tilde{h}^*(f)\tilde{s}(f) + \tilde{h}(f)\tilde{s}^*(f)\right] \frac{df}{S_n(f)}, \tag{21}$$

where $\tilde{s}(f)$ and $\tilde{h}(f)$ are the Fourier transform of $s(t)$ and $h(t)$ respectively, $\tilde{s}^*(f)$ and $\tilde{h}^*(f)$ is the complex conjugate of the Fourier transform of $s(t)$ and $h(t)$ respectively. If we sum over the matched filtering statistic from $f_1 = 0$ to $f_p = \infty$, it gives

$$z = \langle s, h\rangle \equiv 2\int_0^\infty \left[\tilde{h}^*(f)\tilde{s}(f) + \tilde{h}(f)\tilde{s}^*(f)\right] \frac{df}{S_n(f)}. \tag{22}$$

We can construct the χ^2 as

$$\chi^2 = p\sum_{k=1}^{p}\left(z_k - \frac{z}{p}\right)^2. \tag{23}$$

A true signal looks similar to the template waveform, so it has a small χ^2. But for a glitch, the difference between the output and the template is large, therefore it has a large χ^2. We can use this method to separate the signal from noise and increase the sensitivity of the search pipeline.

C. Template bank

Gravitational wave signals of compact binary depend on at least fifteen parameters which are listed in TABLE I. The intrinsic parameters such as mass and spin affect the waveform, and most of the extrinsic parameters only affect the amplitude of waveform.

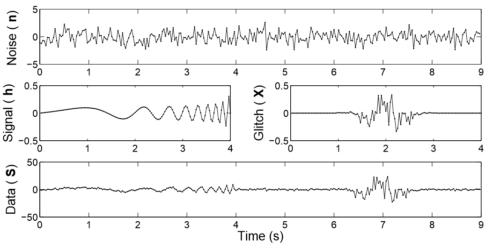


FIG. 4. Components that may contribute to a detector data stream (exaggerated for illustration). Top: Most of the time the data stream is simply Gaussian noise **n**. Center Left: A simulated binary inspiral signal **h**. Center Right: A simulated transient **x**. Bottom: The combination of all contributions **s** [3].

TABLE I. The compact binary parameter space. There are at least fifteen parameters required to specify the orbit of a compact binary (we have ignored parameters associated with eccentricity and the finite size of neutron stars). We refer to the parameters (1)-(8) as intrinsic parameters, while (9)-(15) are called extrinsic. Parameters (9)-(13) enter only in the overall amplitude of the signal, (14) can be maximized over analytically, and (15) can be efficiently searched over with an inverse Fourier transform [1].

Parameters	
component masses	m_1, m_2
component spin vectors	$\vec{S}_1$, $\vec{S}_2$
sky position (right ascension, declination)	α, δ
binary orientation (inclination, polarization angle)	ι, ϕ
luminosity distance	D
coalescence phase	ϕ_{coal}
coalescence time	t_{coal}

The coalescence signals are parameterized by a set of continuous parameters. Since we cannot construct infinite set of templates for matched filtering, a finite set of templates are chosen to construct the bank of templates. The templates are chosen such that any possible signal will have a loss of SNR ≥ 0.97 with at least one template in the template bank. The template bank is said to have a minimal match of 0.97 [5]. In GstLAL, the stochastic placement algorithm is used to construct the template bank. A detailed explanation of the stochastic method can be found in Ref. [7]. FIG. 5 is an example of template bank used in GstLAL and FIG. 6 is an example of injected simulated waveforms into GstLAL.

D. Transformations of templates

GstLAL is a search pipeline designed for a low-latency search. In order to reduce the computational cost, two transformations of the templates are used to reduce computational costs [5].

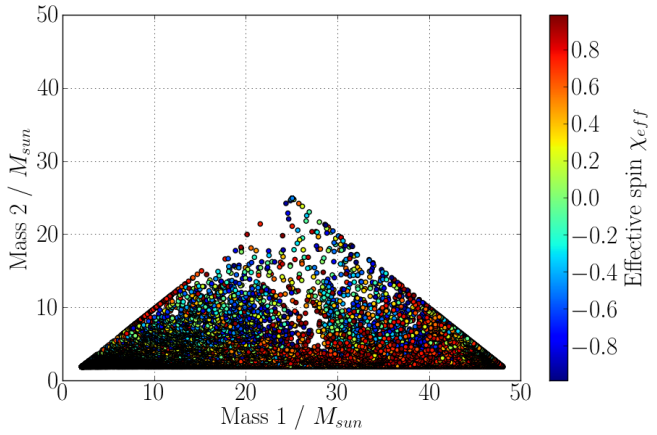


FIG. 5. An example of template bank used in search pipeline with mass and effective spin are shown.

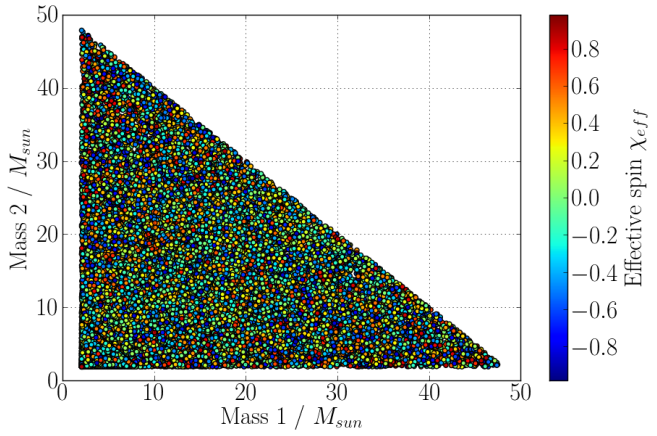


FIG. 6. An example of injections used in search pipeline with mass and effective spin are shown.

1. *Multibanding*

a. Nyquist frequency Nyquist frequency $f = \frac{f_s}{2}$ is the half of the sampling frequency f_s. The discretely sampled data with sampling rate f_s can completely represent a continuous signal which only has frequency content below the Nyquist frequency. The information of signal with frequency higher than the Nyquist frequency will be lost or aliased to lower frequency.

Since the beginning of the inspiral stage has a low frequency, a smaller sample rate is used to reduce the computational cost. The template is divided into time slices in time domain, each template $h_i[k]$ is decomposed into a sum of S non-overlapping templates

$$h_i[k] = \sum_{s=0}^{S-1} \begin{cases} h_i^s[k] & \text{if } t^s \le k/f^0 < t^{s+1} \\ 0 & \text{otherwise} \end{cases}, \tag{24}$$

for S integer $\{f^0 t^s\}$ such that $0 = f^0 t^0 < f^0 t^1 < \ldots < f^0 t^s = N$. The sampling frequency of each time slice must be smaller than Nyquist frequency. For the time-sliced template intervals $[t^0, t^1), [t^1, t^2), \ldots, [t^{S-1}, t^S)$ sampling at frequency $f^0, f^1, \ldots, f^{S-1}$ can be downsampled into

$$h_i^s[k] \equiv \begin{cases} h_i[k\frac{f}{f^s}] & \text{if } t^s \le k/f^s < t^{s+1} \\ 0 & \text{otherwise} \end{cases}. \tag{25}$$

FIG. 7 illustrates how multibanding works. Downsampling reduces the total number of filter coefficients by a factor of ≈ 100 by treating the earliest part of the waveform at $\approx 1/100$ of the full sample rate [5].

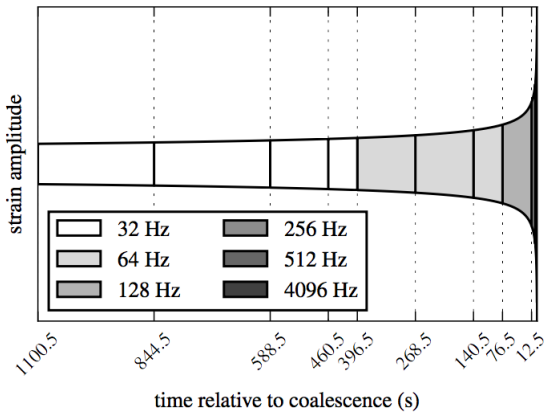


FIG. 7. Multi-banding of templates [5]. Different time slices using different sample rate increase the computational efficiency.

2. *Singular Value Decomposition (SVD)*

The templates in the template bank are highly similar. SVD is a method to reduce the number of filters. A set of templates can be factorized in form of

$$h_i^s[k] = \sum_{l=0}^{M-1} v_{il}^s \sigma_l^s u_l^s[k], \tag{26}$$

where $u_l^s[k]$ are the orthonormal basis templates and $v_{il}^s \sigma_l^s$ is the reconstruction matrix. The number of templates can be reduces from M to L^s by taking away the least important bases indicated by σ_l^s

$$h_i^s[k] \approx \sum_{l=0}^{L^s-1} v_{il}^s \sigma_l^s u_l^s[k], \tag{27}$$

SVD reduces the number of filters needed by another factor of ≈ 100 [5].

E. Ranking events

GstLAL ranks an event from most signal-like to least signal-like using a likelihood-ratio statistic. For a detec-

tion using D detectors, the likelihood ratio is defined as

$$\mathcal{L}(\rho_1, \chi_1^2, \dots \rho_D, \chi_D^2, \bar{\theta}) = \frac{P(\rho_1, \chi_1^2, \dots \rho_D, \chi_D^2, \bar{\theta}|s)}{P(\rho_1, \chi_1^2, \dots \rho_D, \chi_D^2, \bar{\theta}|n)}, \tag{28}$$

where $P(\dots|s)$ is the probability of observing $(\dots)$ given a signal, $P(\dots|n)$ is the probability of observing $(\dots)$ given a noise, ρ_i is the SNR in detector i, χ_i^2 is the χ^2 value in detector i and $\bar{\theta}$ are the intrinsic parameters of the template. Assuming ρ and χ^2 are independent variables in different detectors, the likelihood ratio can be approximated as

$$\mathcal{L}(\rho_1, \chi_1^2, \dots \rho_D, \chi_D^2, \bar{\theta}) \approx \prod_i^D \mathcal{L}_i(\rho_i, \chi_i^2, \bar{\theta}). \tag{29}$$

The likelihood-ratio ranking for each event can be calculated by histogramming [9]. This ranking has a good performance. However, it is difficult to include the information from auxiliary channels. We want to calculate the likelihood ratio using machine learning because it is more flexible to include the information from auxiliary channels.

III. MACHINE LEARNING FOR SIGNAL CANDIDATES RANKING IN SEARCH OF GRAVITATIONAL WAVES FROM COMPACT BINARY COALESCENCE

Ranking the candidates from least signal-like to most signal-like is an important part of gravitational wave searches. A set of features such as signal-to-noise ratio, different types of χ^2, chirp mass and spin of a trigger are obtained from the search pipeline. We are going to rank the candidates using these parameters.

A computational technique called machine learning can be used to train the computer to classify the triggers from signal to noise. We will update the ranking in the search pipeline using our machine learning ranking. The efficiency of machine learning ranking is then compare with that of likelihood-ratio ranking [8] under a given false-alarm rate. The ranking statistic from machine learning is expected to be more feasible to include the information of data quality from auxiliary channels.

A. Machine learning

Machine learning is a method to learn from data. Machine learning is very useful when a pattern exists in a problem, while we have data on it but we cannot pin the relation down mathematically. We want to classify the triggers into real gravitational wave signals and noise. If a real signal has a different set of parameters from a glitch, a pattern exist such that we can try to classify a signal from noise using the output of search pipeline as features. Besides, it is difficult to analytically write down an equation that relates the classification and the parameters, this leads to the help of computer. Moreover, we can simulate waveforms and the parameters can be found through injection to the pipeline, this allows us to use the simulation data to train the computer. Based on these properties, we are going to use machine learning to reduce the false alarm rate and optimize the sensitivity of search pipeline.

1. *Feasibility of learning*

We are going to use finite set of data for the classification learning. The hypothesis found using the in-sample data can have a good performance with the in-sample data, but there is no guarantee that the hypothesis preforms good outside the data. In order to generalize the learning from the in-sample data to out of sample data, a theory is required to ensure the learning is probable outside the sample data. It is important for us because the theory can guarantee that the model found using the data from simulation can also be used in real situation.

The Vapnik-Chervonenkis Inequality (VC inequality) states that the probability of the difference between the error of in-sample data and the target function E_{in} and the error of out of sample data and the target function E_{out}, which less than a value ϵ, is bounded by

$$P\left[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon\right] \leq 4 m_{\mathbb{H}}(2N) e^{-\frac{1}{8}\epsilon^2 N}, \tag{30}$$

where N is the number of in-sample data, and

$$m_{\mathbb{H}}(2N) = \sum_{i=0}^{d_{VC}} \binom{2N}{i}, \tag{31}$$

where d_{VC} is a quantity depends on the hypothesis set which measure the complexity of the hypothesis set. $m_{\mathbb{H}}(2N)$ is a polynomial with maximum power of $N^{d_{VC}}$. The right hand side of the VC inequality contains a decay exponential term. Since exponential increases faster than polynomial, which can be proved by doing $b+1$ times L'Hoptial's rule

$$\lim_{n\to\infty} \frac{n^b}{a^n} = 0, \tag{32}$$

the probability will bound by a value less than one when the number of data N increase to certain amount. Therefore, machine learning is feasible for classifying the signal and noise if the training data is enough.

2. *Overfitting*

Overfitting or overtraining is a big issue in machine learning. FIG. 8 shows an example of overfitting. Overfitting is fitting the data more than it is warranted. Overfitting appears that the in sample error is small, but the

out of sample error is huge, leading to a poor generalization. The cause of overfitting is fitting the noise. We can distinguish the noise into stochastic noise and deterministic noise. The noise is the part that the hypothesis set $\mathbb{H}$ cannot capture whole information of target function f. Stochastic noise is due to the randomness of the input data (ϵ of $y = f(x) + \epsilon(x)$), deterministic noise is come from the limitation of the complexity between target function and the hypothesis set. In order to deal with overfitting, regularization and validation are used in machine learning.

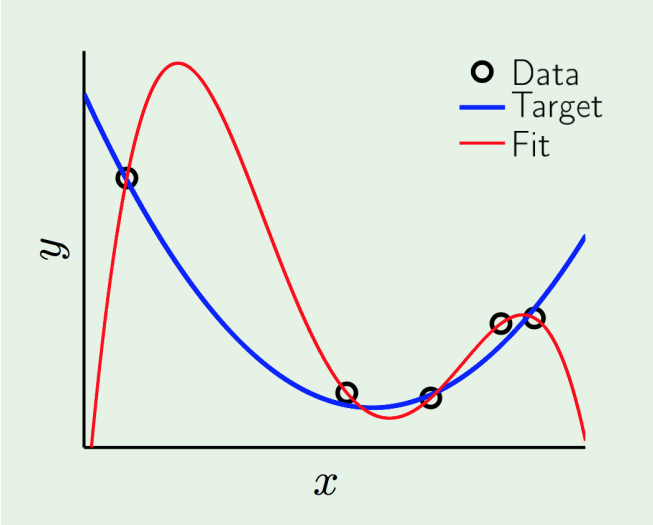


FIG. 8. An example of overfitting [12]. The data on the plot consists of stochastic noise. Although the fitting curve passing through all the data point on the plot such that in sample error is 0, the performance outside the data points is poor, the out of sample error is huge.

a. Regularization Regularization is an approach to deal with overfitting. Since overfitting is caused by fitting the noise, in order to prevent fitting the noise, we can constrain our learning model to prevent fitting it. From practical observation, noise has the following properties

- **Stochastic noise** high frequency
- **Deterministic noise** non-smooth

Regularization is a method to prevent overfitting by constraining learning towards the smoother hypotheses. FIG. 9 is an example of regularization. Mathematically, regularization is a method to deal with the ill-posed problems in function approximation.

b. Validation Validation is another method to prevent overfitting. Recall the relation between in sample error and out of sample error

$$E_{\text{out}}(h) = E_{\text{in}}(h) + \text{overfit penalty}$$

Regularization estimates the overfit penalty and try to reduce it to achieve a good generalization, while validation

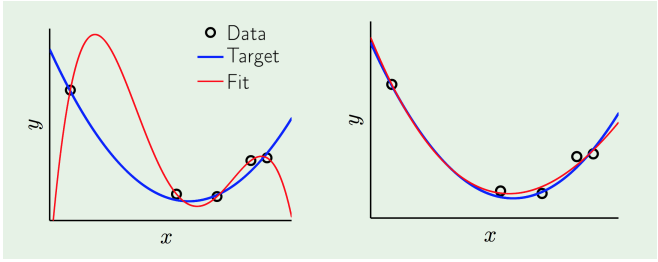


FIG. 9. An example of regularization [12]. The fitting without regularization (left) and the fitting with regularization (right).

estimates the out of sample error E_{out} using a validation set. The estimation can help us to choose a suitable model.

Validation set is a data set other than the training set and testing set. FIG. 10 is an example using validation. The validation set can be extracted from the training set. Validation set gives the validation error E_{val}, which is a better estimation of out of sample error E_{out} than the in sample error E_{in} since the validation set does not affect learning as much as training set does. The final hypothesis is always chosen with the smallest validation error E_{val}.

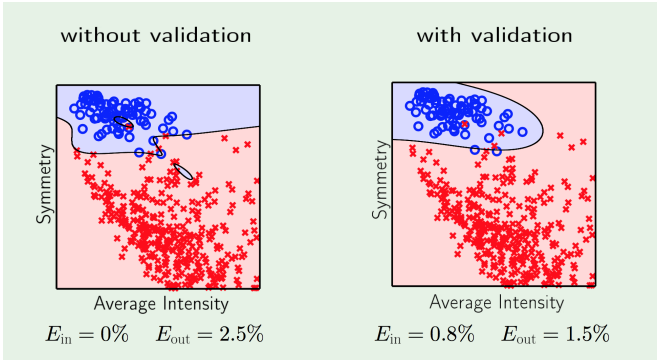


FIG. 10. An example of validation [12]. The classification without validation(left) and the classification with validation(right).

B. Binary classification

In binary classification, we can separate the outcome of an evaluation set into four groups: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

	True class	
	Background	Signal
Class 0	True negatives (TN)	False negatives (FN)
Class 1	False positives (FP)	True positives (TP)

The number of data points fall in these four groups is used for evaluating the ability of a classifier. The false alarm probability (FAP) is defined as

$$FAP = \frac{FP}{FP + TN} \tag{33}$$

and the true positive probability or sensitivity (TPP) is defined as

$$TPP = \frac{TP}{TP + FN} \tag{34}$$

The classifier gives out a ranking statistic instead of a class, false alarm rate (FAR) of an event with a ranking R can be defined as

$$FAR = \frac{N(R > R^*)}{T} \tag{35}$$

where $N(R > R^*)$ is the number of events with a ranking R larger than some threshold R^* and T is the total background time. A receiver operating characteristic (ROC) curve, combining the false alarm probability, true positive probability and threshold, is used to illustrate the performance of a classifier.

C. Random Forest of Bootstrap aggregated Decision Tree (RFBDT) algorithm

RFBDT algorithm is a supervised learning algorithm that can be used for our classification. RFBDT combines the idea of random forest and Bootstrap AGGregatING (bagging) which improves the overtraining problem of a single decision tree and reduces the variance of the result. We use the StatPatternRecognition (SPR) package [13], which is a C++ package for statistical analysis of high energy physics data. RFBDT applies the technique of bagging, which tends to have a better performance in high noise situations compare to other techniques such as boosting [10].

We want to distinguish signal from noise for each trigger. Each signal candidate is characterized by a feature vector containing a set of parameters which is useful for the classification. We construct a decision tree using the training set. The class (signal or background) of each data point in the training set is known. A decision tree consists of series of binary splits depends on the feature vector parameters. The training set begins at the first node of the tree and split into branches. The threshold of each node is selected based on the optimization criterion. The nodes split into branches continuously. When no node can optimize the optimization criterion or reaches the minimal leaf size that is predefined, the node no longer split and becomes a leaf. When all the nodes become leaves, a decision tree is created.

Bagging is a technique that resamples a set of data from the training data. A different set of training events is chosen randomly for each decision tree. A different subset of feature vector parameters is also chosen randomly at each node for the next split. We construct multiple bagged trees to form a random forest. This method makes each tree in the forest is unique and the result from each tree can be averaged to reduce the variance in the statistical classification.

The SPR package provides several optimization criterion, the Gini index and negative cross-entropy are found to have a suitable performance for the search [11]. The Gini is defined by

$$G(p) = -2p\bar{p} \tag{36}$$

where p is the fraction of signals in a node and $\bar{p} = 1 - p$ is the fraction of background in a node. The data is split at node to minimize the Gini index. The negative cross-entropy is defined by

$$H(p) = -p\log_2 p - \bar{p}\log_2 \bar{p} \tag{37}$$

The data is split at node to minimize the negative cross-entropy function.

After the training using training set, the forest can be used to classify signal to noise. An event is first start at the initial node of each tree and passes through branches until it reaches the leaf. We can compute the probability of that event is an signal from the bagged decision trees in the forest

$$p_{\text{forest}} = \frac{\sum s_i}{\sum s_i + b_i} = \frac{1}{N}\sum s_i \tag{38}$$

where s_i and b_i are the number of signal and background events from the training in the ith leaf and N is the total number of events in all finial leaves. FIG. 11 shows how this value is calculated from a forest. The ranking statistic M_{forest}, which is the ratio of probability that the event is a signal to the probability that the event is a background, is given by

$$M_{\text{forest}} = \frac{P(\text{event}|\text{signal})}{P(\text{event}|\text{background})} = \frac{p_{\text{forest}}}{1 - p_{\text{forest}}} \tag{39}$$

IV. RESULTS

A. Data set

The background is extracted from the data in aLIGO engineering run 7 (ER7) and the signals are from the simulated waveform injection. The data set with 18553 injections and 27607 background triggers is split into a training set, a validation set and a testing set.

a. Background We estimate the background by matching the singles of L1 and H1 detectors. A coincidence signal has the same intrinsic parameters (masses and spins) in the detection between two detectors, we match the single data of L1 and H1 with the same masses and spins while the difference between arrival time is larger than a coincidence time window, to form an accidental coincidence event. A uniform time difference distribution is used for background in order to simulate the time difference distribution in accidental coincidence.

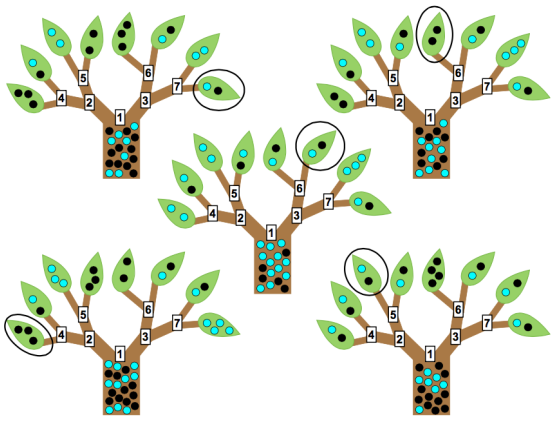


FIG. 11. A cartoon of a random forest. There are five decision trees in this random forest. Each was trained on a training set of objects belonging to either the black set or the cyan set. Note that the training set of each decision tree is different from the others. At each numbered node, or split in the tree, a binary decision based on a threshold of a feature vector parameter value is imposed. The decisions imposed at each node will differ for the different trees. When no split on a node can reduce the entropy or it contains fewer events than a preset limit, it is no longer divided and becomes a "leaf". Consider an object that we wish to classify as black or cyan. Suppose the object ends up in each circled leaf. Then the probability that the object is black is the fraction of black objects in all leaves, $p_{\text{forest}} = 73\%$ [11].

We define a default setting as n = 100, s = 5, l = 6 and c = Gini index. FIG. 12 is a histogram of the ranking statistic of the evaluation set data using the default setting. Assuming the threshold equal to 0.5, 98.1% of the total evaluation data has been correctly identified. Since the false alarm probability and the true positive probability depend on the threshold we choose, we can step through the threshold from 0 to 1 to get the corresponding false alarm probability and true positive probability to obtain a ROC curve, which illustrate the performance of that classifier. In each tuning, we change one parameter from the default setting to investigate the optimal value of this parameter.

a. Number of decision trees We vary the number of decision trees with $n = 100$, $n = 500$ and $n = 2000$. In FIG. 13, the ROC curve shows that there is no improvement in increasing the number of decision trees more than 100. It is enough to have 100 decision trees in training.

b. Minimum leaf size We vary the minimum entries per leaf from $l = 1$ to $l = 100$. In FIG. 14, the ROC curve shows that $l = 5$ has the best performance. As

b. Signal There is no gravitational-wave signal detected until now. We use simulated waveform as our signal. The coincidence signal events can be obtained by injecting simulated waveforms into GstLAL.

1. *Tuning parameters of RFBDT*

RFBDT has several tunable parameters listed in TABLE II. We use 9 features listed in TABLE III to illustrate the tuning. The information about the distribution of the data set we used and the ROC curve of the classification with more features used can be found in appendix. In order to get the parameters which can maximize the performance of the classifier the most, we train the classifiers with varying parameters using the training set, then we evaluate the performance of each classifier by plotting a ROC curve using the validation set data. The classifier with the largest true positive probability (efficiency) at a given false alarm probability has the best performance.

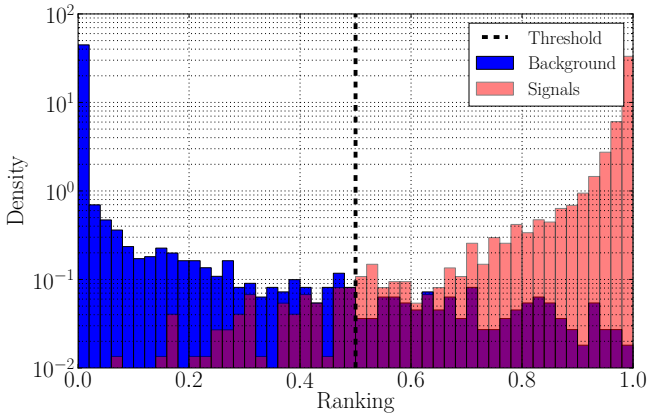


FIG. 12. A histogram of the ranking statistics of the evaluation data using n = 100, s = 5, l = 6 and c = Gini index setting. Assuming the threshold equal to 0.5, 98.1% of the total evaluation set has been correctly identified.

mentioned in TABLE II, overtraining occurs when l is too smaller while undertraining occurs when l is too large. It

TABLE II. A summary of tunable parameters in RFBDT.

Option	Parameter	Description
n	Number of decision trees	A larger number of decision trees provides a more stable ranking statistic M_{forest}. However, the training set contains finite information, too many trees in a forest make the trees become redundant and consume more computational resources.
l	Minimum leaf size	When the number of events in a node reaches the minimum leaf size, the data stop from splitting into two nodes and the node becomes a leaf. The choice of the leaf size corresponds to the overfitting issue. If the leaf size is too small, the tree can classify the training data perfectly but leads to a poor generalization, overtraining occurs. If the leaf size is too large, the trees are undertrained and cannot classify an event, result the ranking statistic M_{forest} concentrates at half between two classes.
s	Number of sampled parameters	At each node, m sampled parameters are chosen randomly to from a subset of original feature vector. The split criterion is evaluated for each parameter inside the subset and a split parameter is chosen out of m parameters. If m is too large, each node has the same number of parameters for splitting as the original feature vector, some of the parameters may be used again and again, the training set is not fully explored. The trees in the forest are similar as a result of overtraining. If m is too small, each node is forced to split base on poor parameters. The split is inefficient and undertraining occurs.
c	Optimization criterion	The optimization criterion is used to choose the parameters used for splitting out of sampled parameters and the splitting threshold. SPR package provides several optimization criteria: correctly classified fraction, signal significance $s/\sqrt{(s+b)}$, purity s/(s+b), tagger efficiency Q, Gini index, cross-entropy, 90% Bayesian upper limit with uniform prior and discovery potential $2*(\sqrt{(s+b)}-\sqrt{(b)})$.

TABLE III. The features from GstLAL output.

Feature	Description
Δt	Time difference between triggers in two detectors H1 an
m_1	The mass of the template matched to the data. $m_1 > m$
m_2	The mass of the template matched to the data. $m_1 > m$
s_1	The aligned spin of m_1.
s_2	The aligned spin of m_2.
ρ_{H1}	Signal-to-noise ratio of the trigger in H1 data.
ρ_{L1}	Signal-to-noise ratio of the trigger in L1 data.
χ^2_{L1}	χ^2 value of trigger in L1 data.
χ^2_{H1}	χ^2 value of trigger in H1 data.

is reasonable for $l = 5$ giving the best performance.

c. Number of sampled parameters We vary the number of sampled parameters from $s = 1$ to $s = 10$. In FIG. 15, the ROC curve shows that $s = 6$ has the best performance while we are using 9 dimensions feature space.

d. Optimization criterion We vary the optimization criterion in SPR package. In FIG. 16, the ROC curve shows that Gini index, negative cross-entropy and discovery potential have a similar performance. We can vary the optimization criterion with these three option in our future training. Using Gini index or cross-entropy

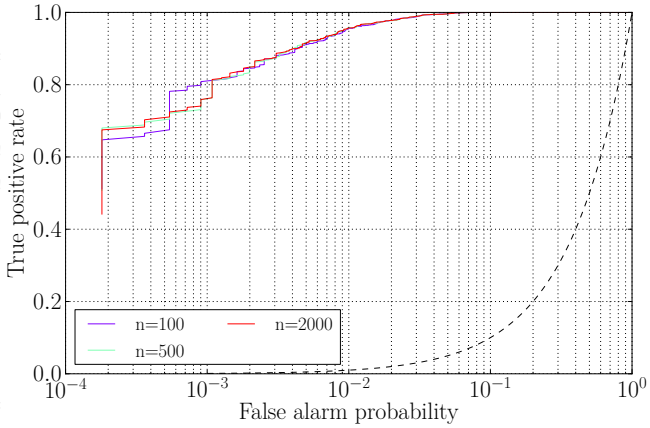


FIG. 13. ROC curve with varying number of decision trees. No improvement can be achieved using more than 100 decision trees.

can minimize the false alarm probability and maximize the true positive probability which we are concerned the most, while other optimization criterion may optimize false negative probability or true negative probability which we are less concerned.

The optimal setting of the classifier using GstLAL output as the features is $n = 100$, $l = 5$, $s = 6$ and $c =$ Gini index.

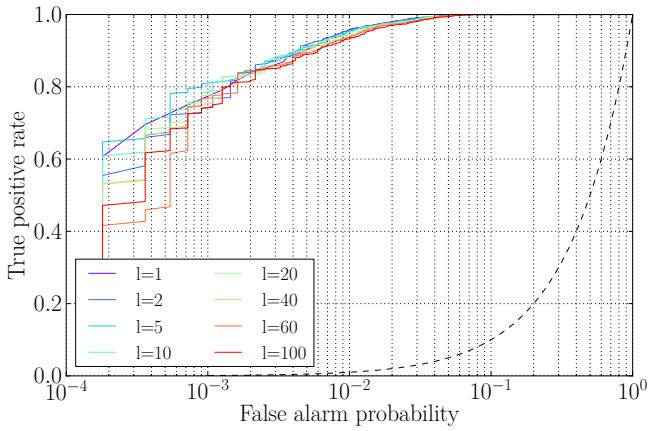


FIG. 14. ROC curve with varying minimum entries per leaf. Using $l = 5$ has the best performance.

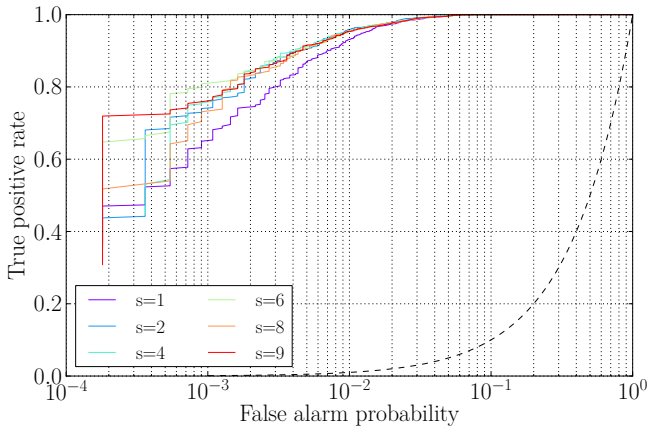


FIG. 15. ROC curve with varying number of sampled features. Using $s = 6$ has the best performance.

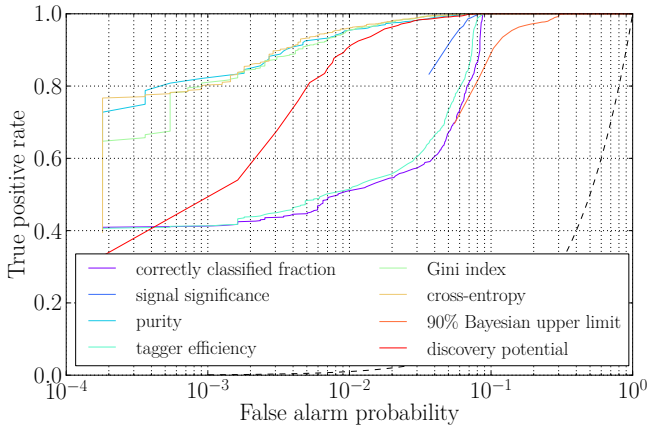


FIG. 16. ROC curve with varying the optimization criterion. Gini index, negative cross-entropy and discovery potential have a comparable performance.

B. Expanding feature space

Features of an event is a characteristic of it. A better representation of an event can have a better performance of the classifier. We investigate the performance of the classifier by transforming the GstLAL output into other quantities with physical meaning listed in TABLE IV. Classifiers used different features listed in TABLE V are first trained with the training set, then they are tuned to achieve optimal setting using the validation set as mentioned in the previous section. After that, the classifiers are re-trained using the training set with the optimal setting and compare their performance using a testing set.

TABLE IV. The features from transforming GstLAL output.

Feature	Description
$\mathcal{M}$	Chirp mass. $\mathcal{M} = \frac{(m_1 m_2)^{3/5}}{(m_1+m_2)^{1/5}}$
η	Symmetric mass ratio. $\eta = \frac{m_1 m_2}{(m_1+m_2)^2}$
$\mathcal{M}^{-5/3}/\eta$	Related to chirp time coordinate. $\mathcal{M}^{-5/3}/\eta = \frac{(m_1+m_2)^{11/5}}{(m_1 m_2)^2}$
$\mathcal{M}^{-2/3}/\eta$	Related to chirp time coordinate. $\mathcal{M}^{-2/3}/\eta = \frac{(m_1+m_2)^{37/15}}{(m_1 m_2)^{7/5}}$
$\chi_{\rm eff}$	Effective spin. $\chi_{\rm eff} = \frac{m_1 s_1 + m_2 s_2}{m_1+m_2}$
$\rho_{\rm eff,H1}$	Effective SNR of H1. $\rho_{\rm eff,H1} = \frac{\rho_{\rm H1}}{\left[\frac{\chi^2_{\rm H1}}{10}\left(1+\rho^2_{\rm H1}/50\right)\right]^{1/4}}$
$\rho_{\rm eff,L1}$	Effective SNR of L1. $\rho_{\rm eff,L1} = \frac{\rho_{\rm L1}}{\left[\frac{\chi^2_{\rm L1}}{10}\left(1+\rho^2_{\rm L1}/50\right)\right]^{1/4}}$

TABLE V. The features from transforming GstLAL output.

Number of Features	Feature space
9	$\Delta t, m_1, m_2, s_1, \rho_{\rm H1}, \rho_{\rm L1}, \chi^2_{\rm H1}, \chi^2_{\rm L1}$
12	$\Delta t, m_1, m_2, s_1, \rho_{\rm H1}, \rho_{\rm L1}, \chi^2_{\rm H1}, \chi^2_{\rm L1}, \mathcal{M}, \eta, \chi_{\rm eff}$
14	$\Delta t, m_1, m_2, s_1, \rho_{\rm H1}, \rho_{\rm L1}, \chi^2_{\rm H1}, \chi^2_{\rm L1}, \mathcal{M}, \eta, \chi_{\rm eff}$ $\mathcal{M}^{-5/3}/\eta, \mathcal{M}^{-2/3}/\eta$
16	$\Delta t, m_1, m_2, s_1, \rho_{\rm H1}, \rho_{\rm L1}, \chi^2_{\rm H1}, \chi^2_{\rm L1}, \mathcal{M}, \eta, \chi_{\rm eff}$ $\mathcal{M}^{-5/3}/\eta, \mathcal{M}^{-2/3}/\eta, \rho_{\rm eff,H1}, \rho_{\rm eff,L1}$

We trained the classifier with 9, 12, 14 and 16 features, the ROC curve which illustrates the performance of the classifiers is shown in FIG. 17. From the curve, the classifier trained with 14 features has a little better performance compare with the classifier trained with only GstLAL output. The result shows that the classification is related to the chirp time coordinate. Moreover, adding effective SNR into the feature vector generates noise in classification and leads to a poorer performance. It may due to the redundancy of normal SNR and the effective SNR.

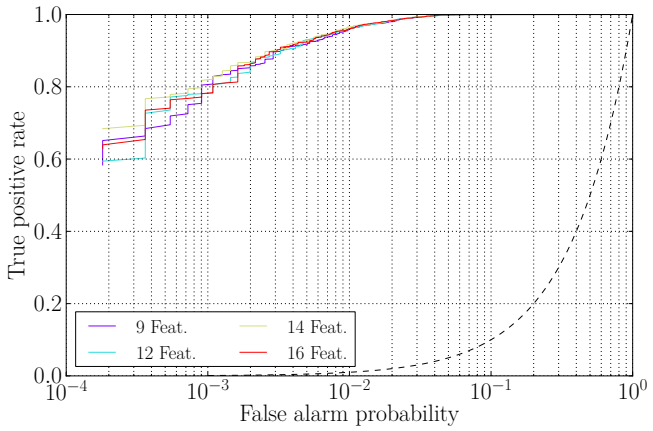


FIG. 17. ROC curve of classifiers trained with different feature space. Using 14 features for training has the best performance. The result shows that the classification is related to the chirp time coordinate. Moreover, adding effective SNR into the feature vector generates noise in classification and leads to a poorer performance. It may due to the redundancy of normal SNR and the effective SNR.

C. Implementation of RFBDT in GstLAL pipeline

Ranking a gravitational-wave candidate is a part of the whole analysis. We implement the RFBDT in GstLAL pipeline. The flow chart of the pipeline is shown in FIG. 18. Single data of two detectors is obtained from a database (SQLite format), which also contains the coincident events whose ranking are going to be evaluated. Then, the background coincidence events are generated by matching the singles. Injection events are obtained from another database file containing the GstLAL output of the simulated signals. After that, the training data (PAT format) is sent to train the RFBDT classifier. Once the classifier is trained, the coincident events are sent to the classifier for evaluation. Finally, we update the ranking of each candidate using our RFBDT ranking.

V. CONCLUSION AND FUTURE WORK

We tuned the RFBDT parameter to obtain the optimal setting of the classifier. We expanded the feature vector and investigate the performance of the classifiers. Besides, we implemented the RFBDT in GstLAL pipeline.

GstLAL is using another method to calculate the ranking of a gravitational-wave candidate. We would like to compare the performance between our ranking using the technique of machine learning with the likelihood-ratio ranking currently use in GstLAL.

Features selection and extraction are important in machine learning. A more systematic method can be conducted in features selection. For example, we can use some selection algorithm for selecting useful features. Moreover, we can do a principal component analysis

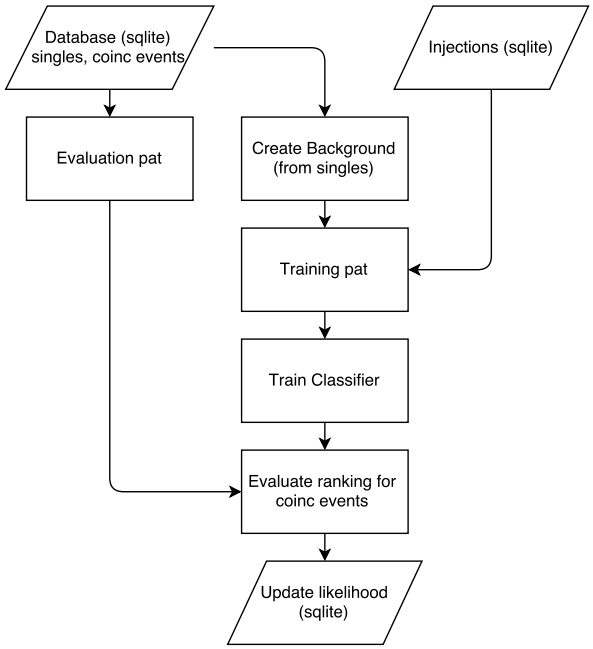


FIG. 18. Flow chart of RFBDT pipeline. We implement this part in GstLAL pipeline.

(PCA) to extract the linear transform of features from GstLAL output and investigate the relation between the GstLAL output.

Since RFBDT has several tunable parameters which have to choose before training, we can update the RFBDT pipeline to choose the parameters by validation automatically.

The ultimate goal of multivariate statistical classification is to include data quality information from auxiliary channels. More investigation is required in order to include the data quality as a feature.

VI. ACKNOWLEDGEMENT

I would like to thank Surabhi Sachdev, Tjonnie Li, Kent Blackburn and Alan Weinstein for giving me this opportunity to learn and work as a LIGO SURF student. I would also like to thank them for their great support on my research. I would like to thank LIGO Scientific Collaboration, Caltech SURF and NSF for funding.

[1] Privitera, Stephen M., (2014), *The importance of spin for observing gravitational waves from coalescing compact binaries with LIGO and Virgo.* Disserta-

tion (Ph.D.), California Institute of Technology. http://resolver.caltech.edu/CaltechTHESIS:05282014-160218103

[2] B.S. Sathyaprakash and Bernard F. Schutz, *Physics, Astrophysics and Cosmology with Gravitational Waves*, Living Rev. Relativity, **12**, (2009), 2. URL (cited on September 26, 2015): http://www.livingreviews.org/lrr-2009-2

[3] Chad Hanna., (2008), *Searching for gravitational waves from binary systems in non-stationary data.* Dissertation (Ph.D.), Louisiana State University.

[4] Bruce Allen, Warren G. Anderson, Patrick R. Brady, Duncan A. Brown, Jolien D.E. Creighton, FINDCHIRP: *An Algorithm for detection of gravitational waves from inspiraling compact binaries*, Phys. Rev. **D85** (2012) 122006, [arXiv:gr-qc/0509116v2].

[5] K. Cannon et. al., *Toward Early-warning Detection of Gravitational Waves from Compact Binary Coalescence*, ApJ. **748** (2012) 136, [arXiv:1107.2665v4].

[6] R. A. Hulse and J. H. Taylor, *Discovery of a pulsar in a binary system*, Astrophys. J., Lett. **195**, L51 (1975).

[7] I. W. Harry, B. Allen, and B. Sathyaprakash, *A stochastic template placement algorithm for gravitational wave data analysis*, Phys. Rev. D **81**, 024004 (2009), 0908.2090, [arXiv:0908.2090].

[8] K. Cannon, C. Hanna, J. Peoples. *Likelihood-Ratio Ranking Statistic for Compact Binary Coalescence Candidates with Rate Estimation*, [DCC: P1400175-v4]

[9] K. Cannon, C. Hanna, and D. Keppel, Phys. Rev. **D88**, 024025 (2013), arXiv:1209.0718 [gr-qc].

[10] E. Bauer and R. Kohavi, Machine Learning **36**, 105 (1999).

[11] P. Baker, S. Caudill, K. Hodge, D. Talukder, C. Capano, N. Cornish, *Multivariate Classification with Random Forests for Gravitational Wave Searches of Black Hole Binary Coalescence*, [DCC:P1400231-v3]

[12] Y. S. Abu-Mostafa, (2012). *Machine Learning* [PowerPoint slides]. Retrieved from http://work.caltech.edu/lectures.html

[13] I. Narsky, *StatPatternRecognition: A C++ Package for Statistical Analysis of High Energy Physics Data*, [arXiv:physics/0507143]

[14] Tjonnie Li, (2014), Powerpoint of colloquium in CUHK: *Gravitational Waves: A new window on the Universe.*

Appendix A: Distribution of features in data set

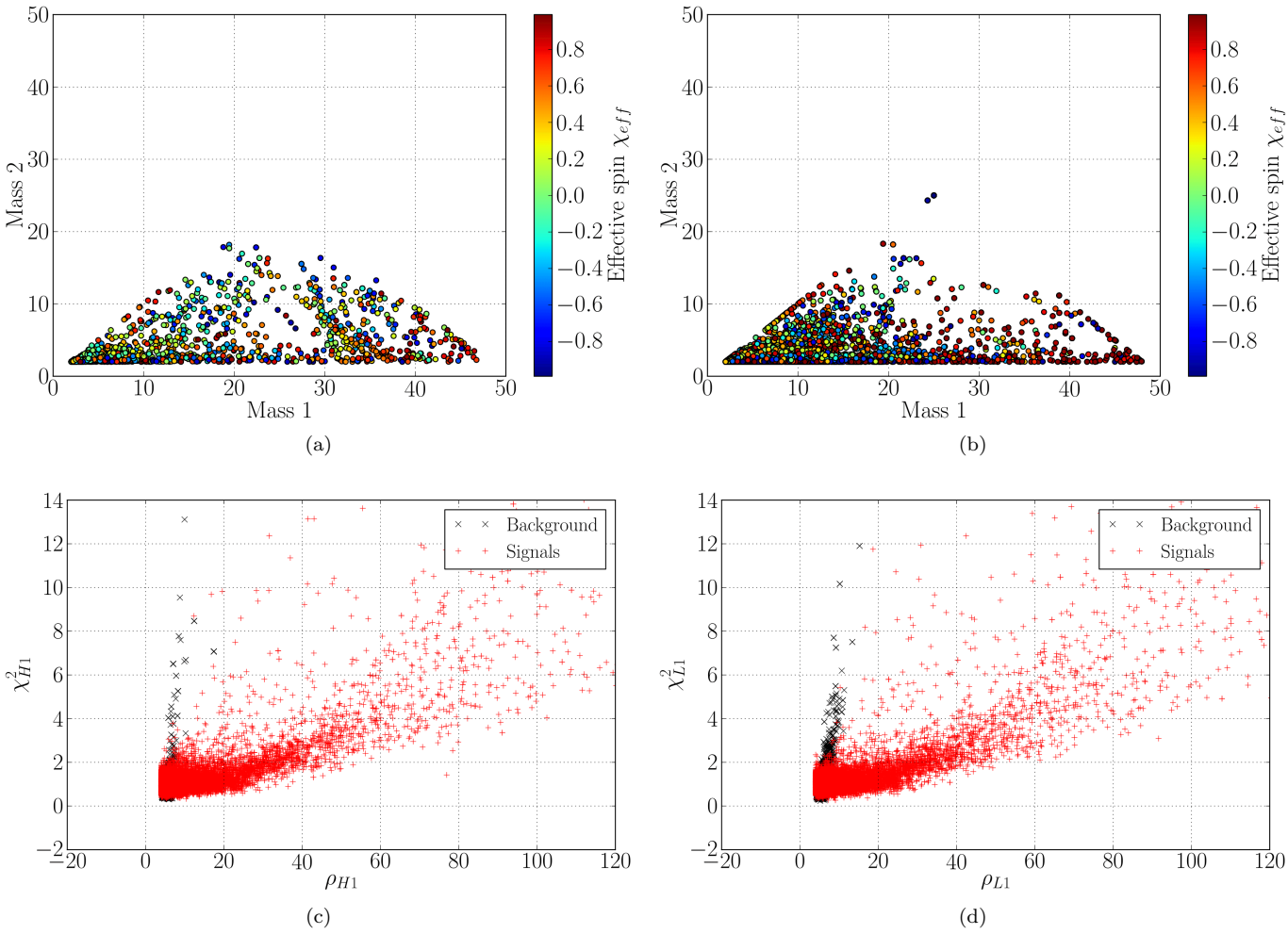


FIG. 19. The intrinsic parameters of (a) signals and (b) background. A plot of SNR against χ^2 of (c) H1 and (d) L1.

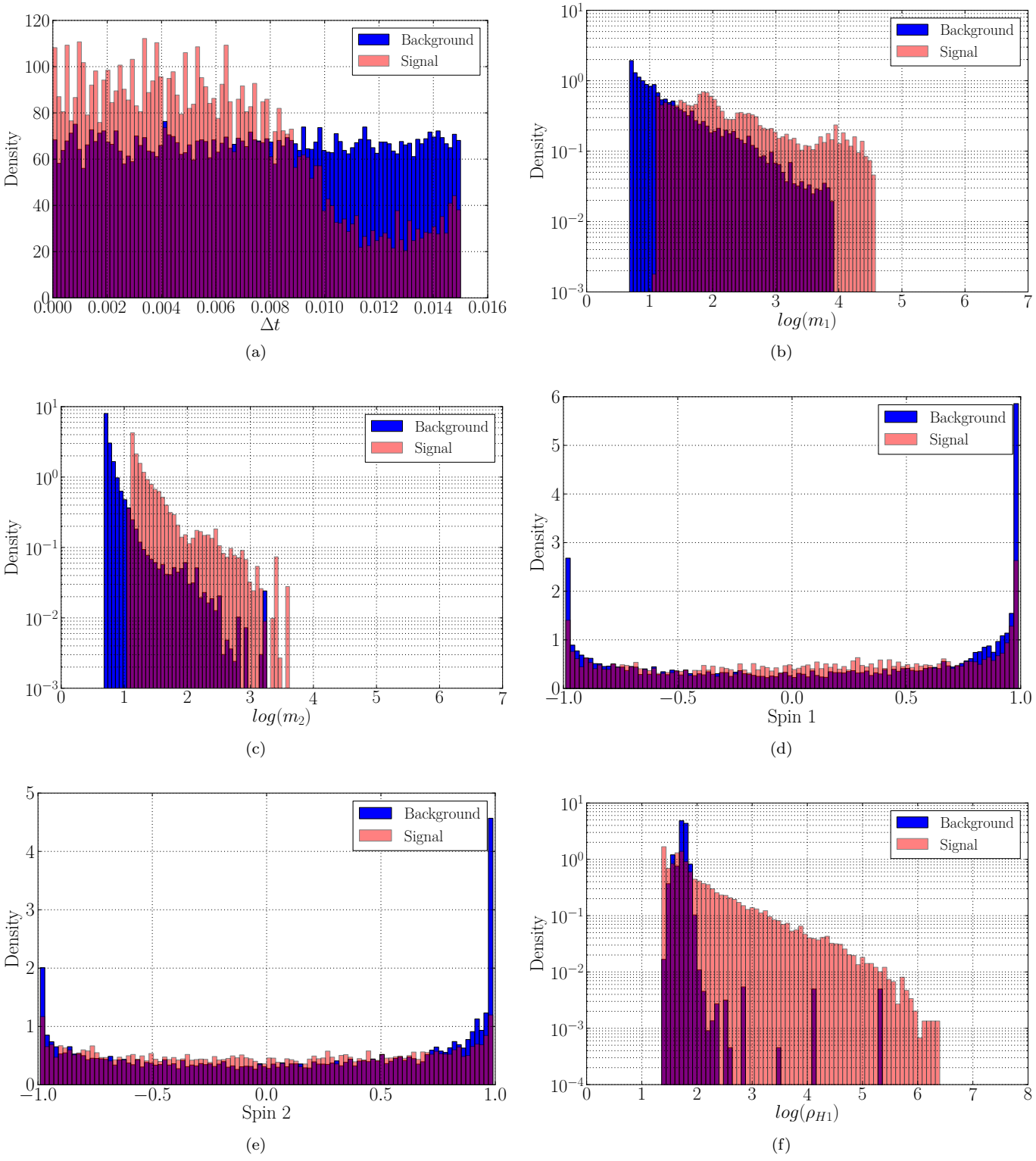


FIG. 20. Density distribution of (a) Δt, (b) m_1, (c) m_2, (d) s_1, (e) s_2 and (f) ρ_{H1} of the data set.

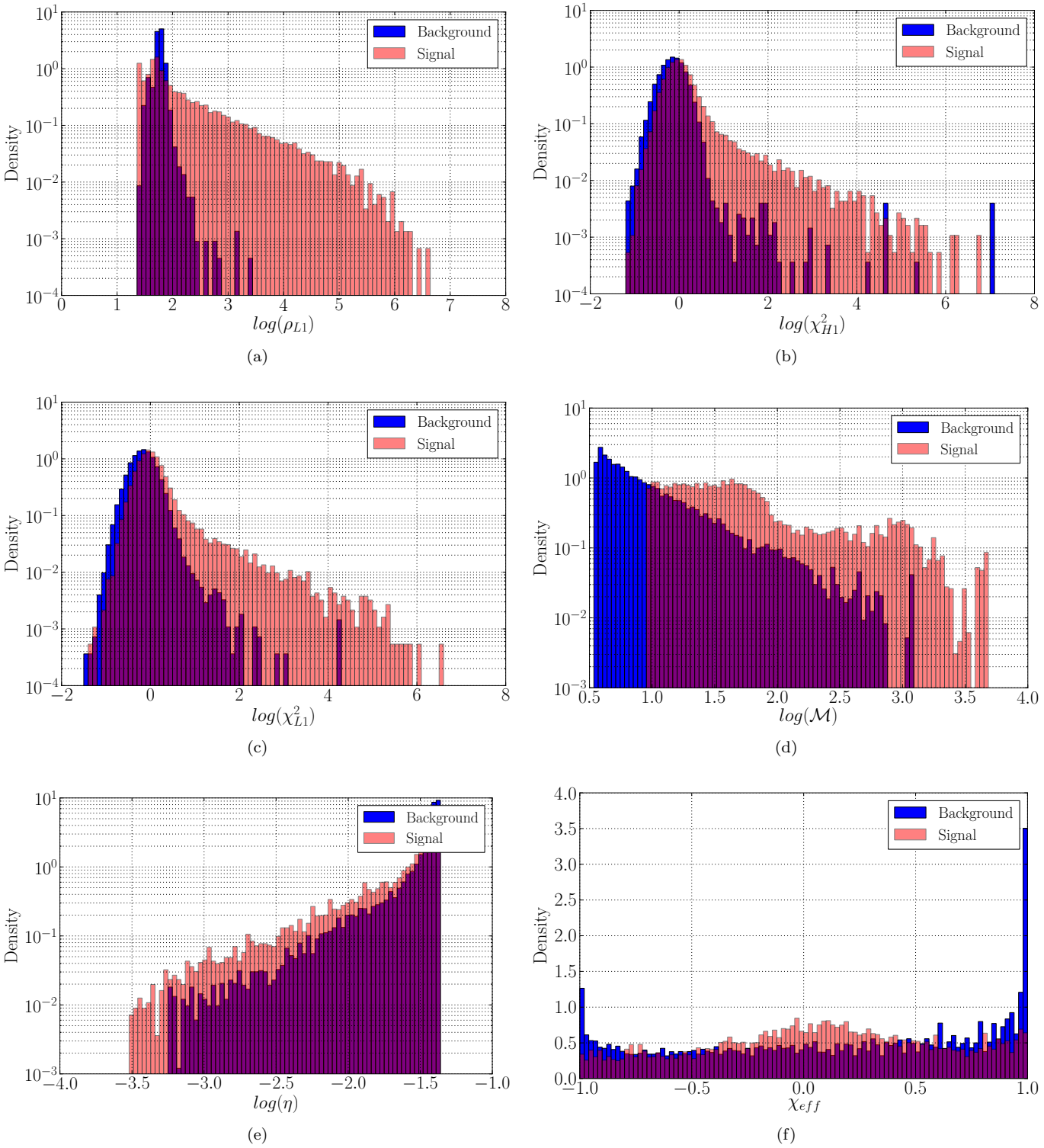


FIG. 21. Density distribution of (a) $\rho_{\rm L1}$, (b) $\chi^2_{\rm H1}$, (c) $\chi^2_{\rm L1}$, (d) $\mathcal{M}$, (e) η and (f) $\chi_{\rm eff}$ of the data set.

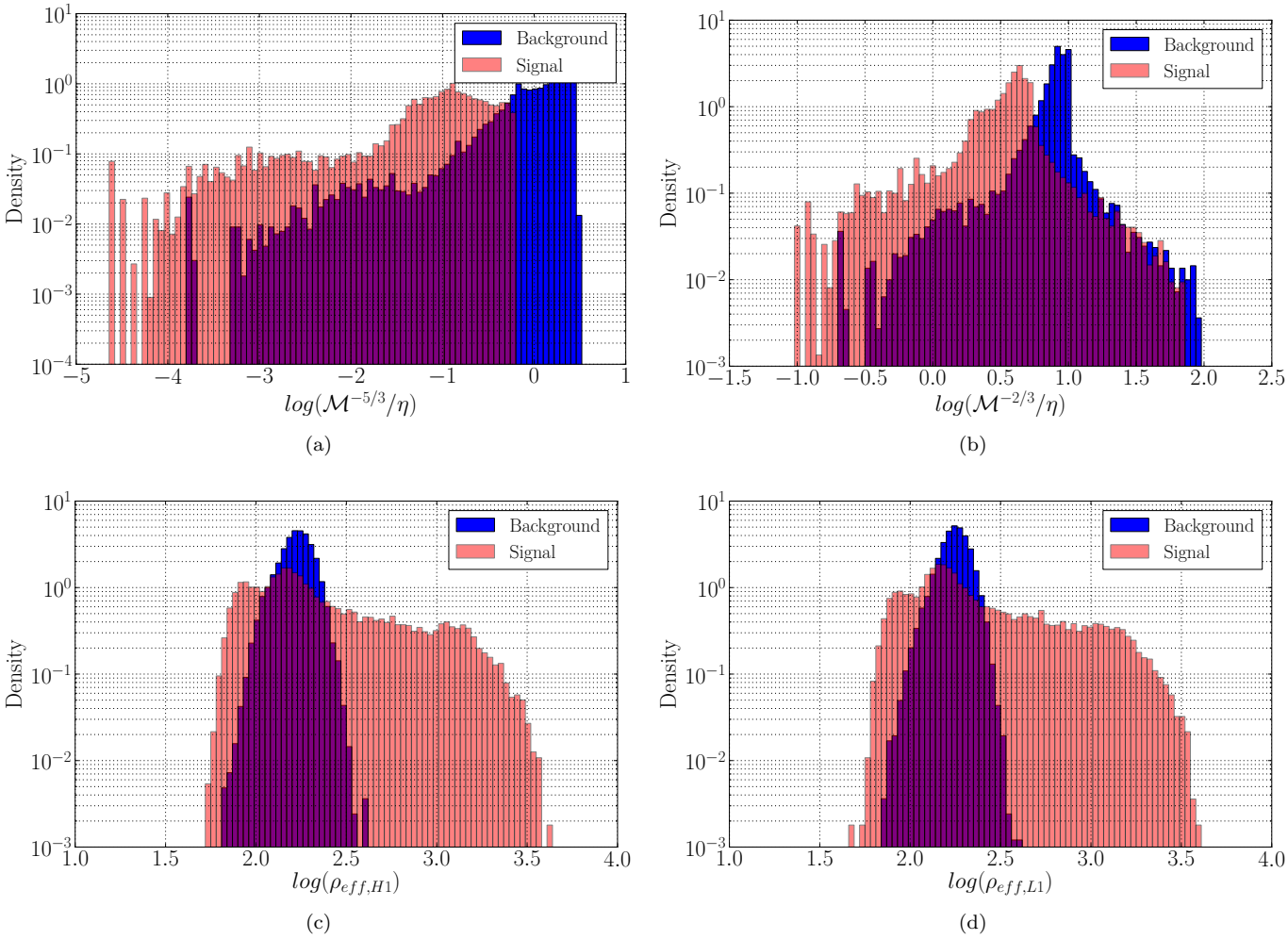


FIG. 22. Density distribution of (a) $\mathcal{M}^{-5/3}/\eta$, (b) $\mathcal{M}^{-2/3}/\eta$, (c) $\rho_{\mathrm{eff,H1}}$ and (d) $\rho_{\mathrm{eff,L1}}$ of the data set.

Appendix B: Tuning for different features

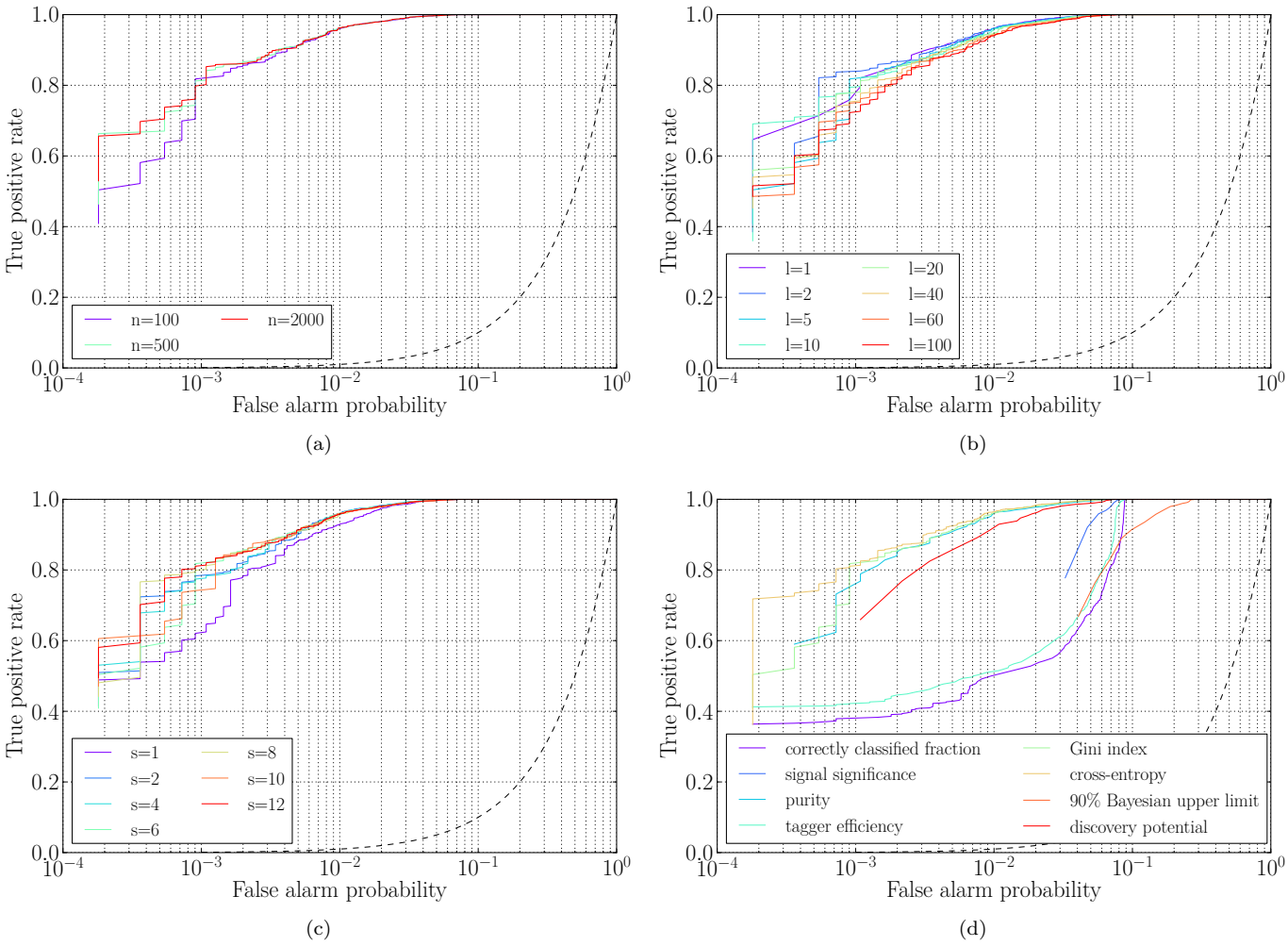


FIG. 23. ROC curve of 12 features with (a) varying number of decision trees, (b) minimum entries per leaf, (c) number of sampled features and (d) the optimization criterion.

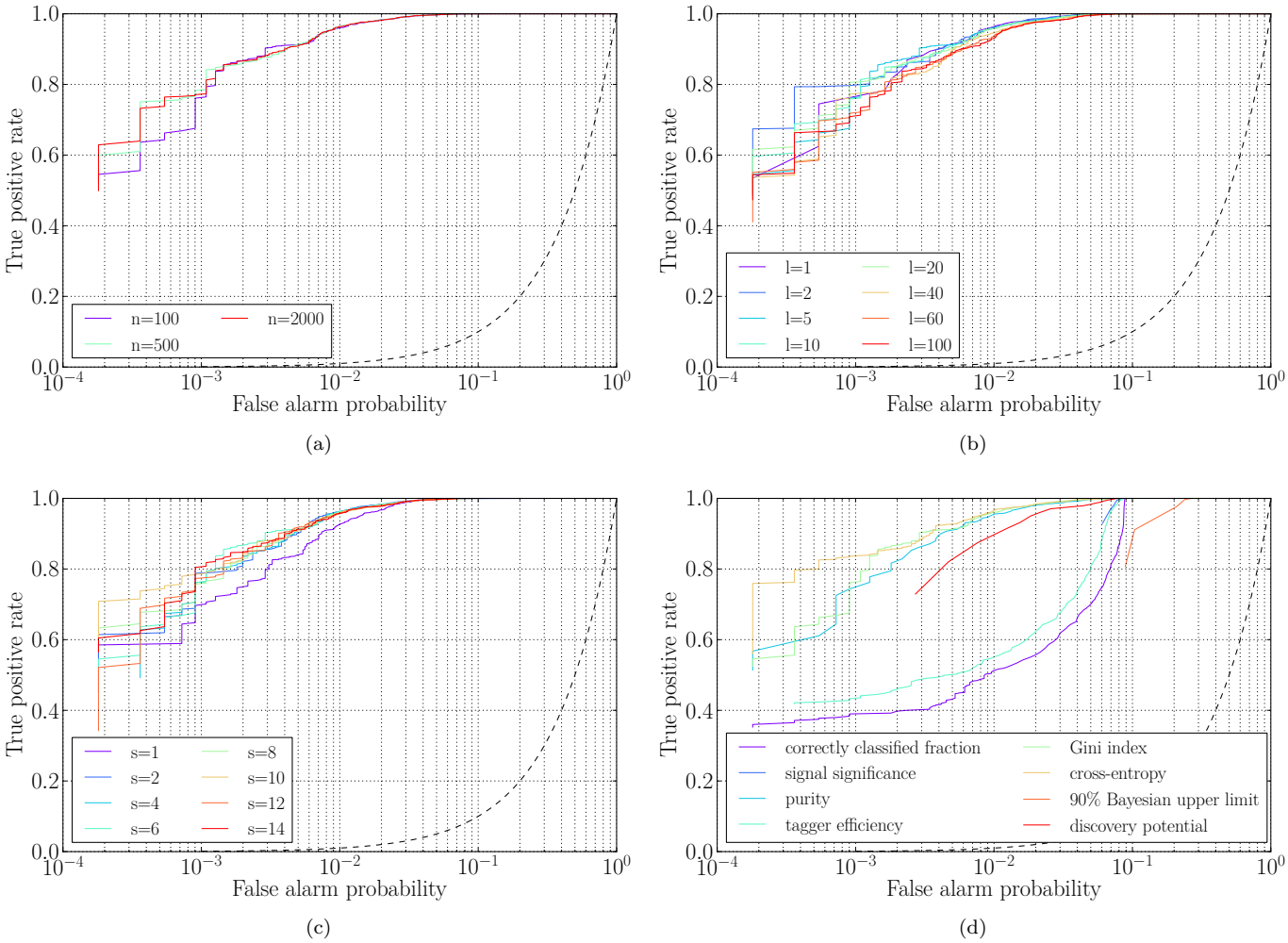


FIG. 24. ROC curve of 14 features with (a) varying number of decision trees, (b) minimum entries per leaf, (c) number of sampled features and (d) the optimization criterion.

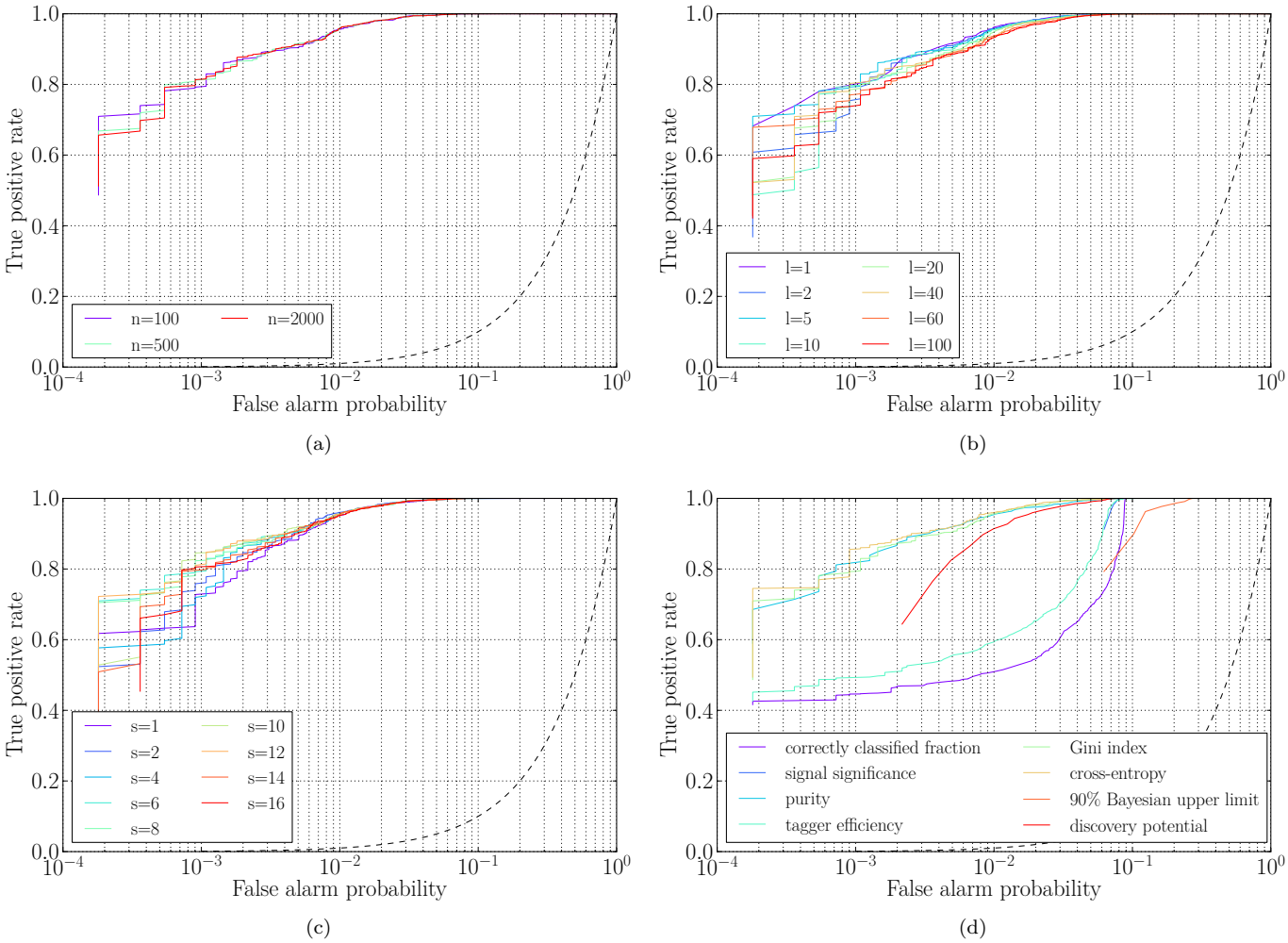
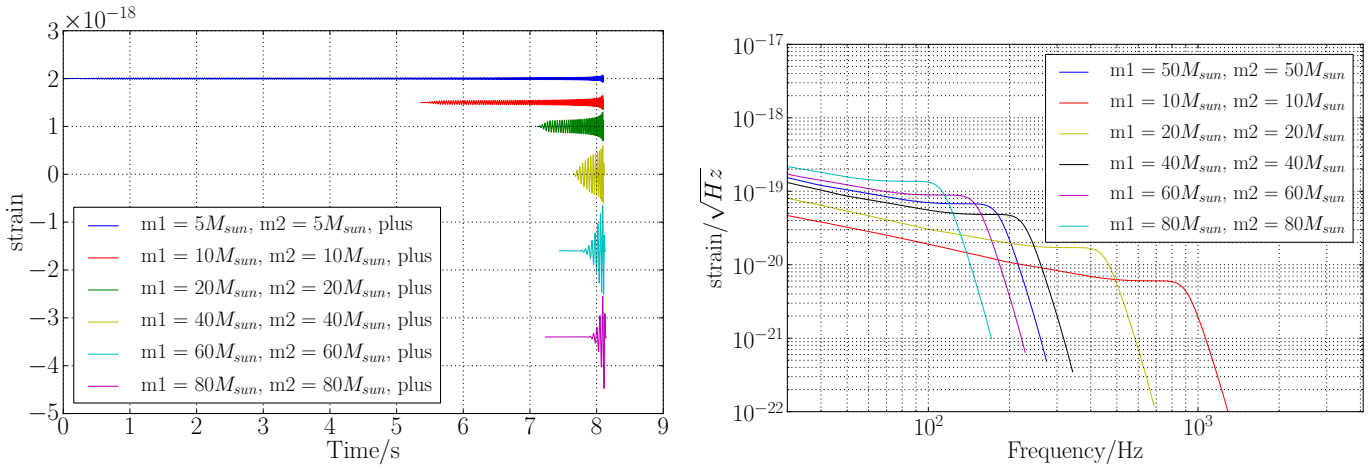


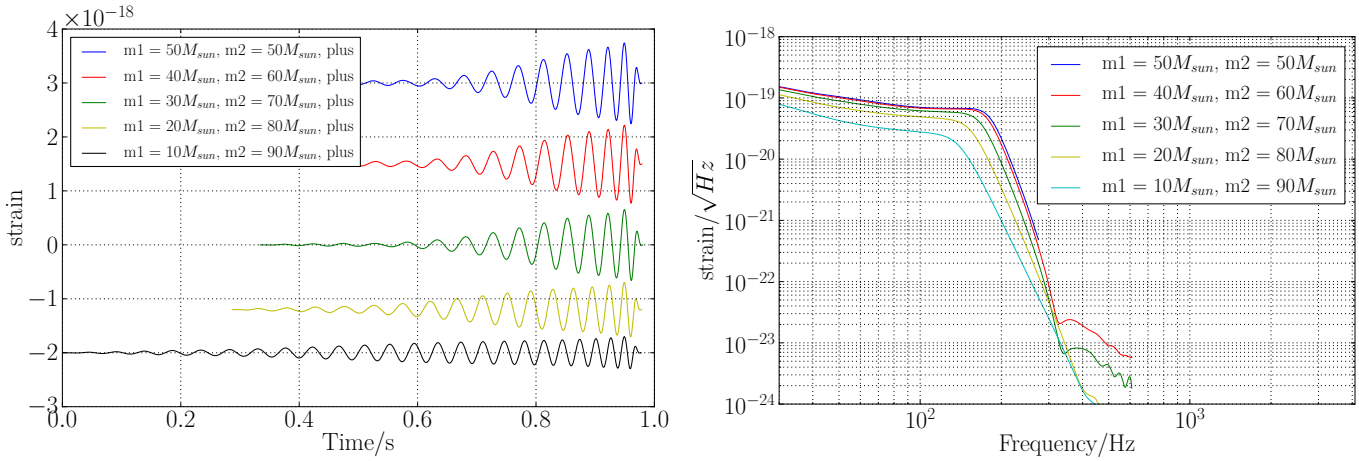
FIG. 25. ROC curve of 16 features with (a) varying number of decision trees, (b) minimum entries per leaf, (c) number of sampled features and (d) the optimization criterion.

Appendix C: Waveforms with varying parameters



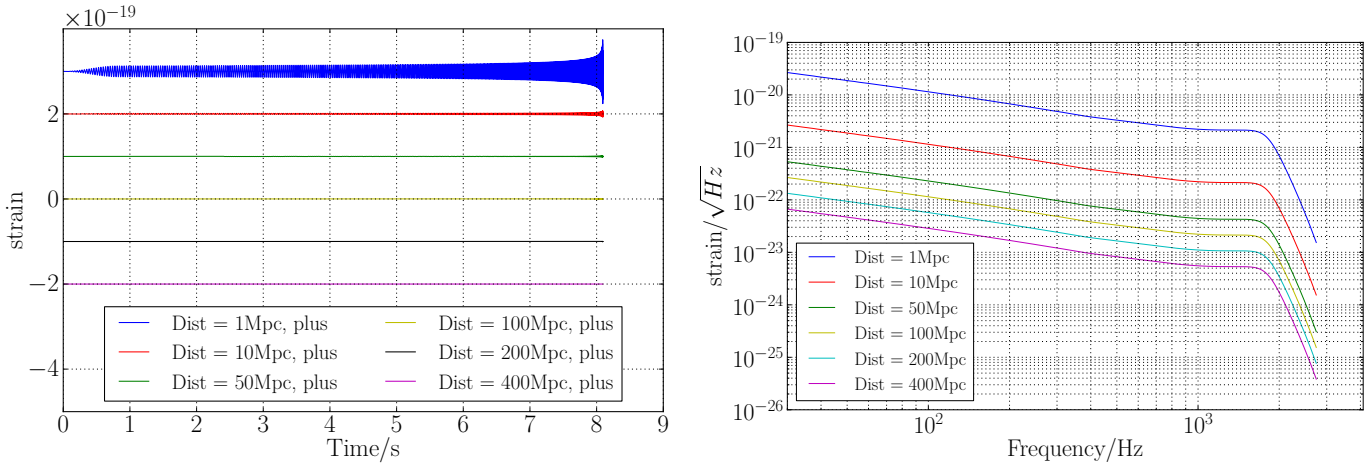
(a) (b)

FIG. 26. (a) Time domain and (b) frequency domain waveforms with varying mass. For a more massive system, the amplitude of strain is larger. A more massive system also end the coalescence at a lower frequency since the energy loss of a higher mass system due to gravitational radiation is larger.



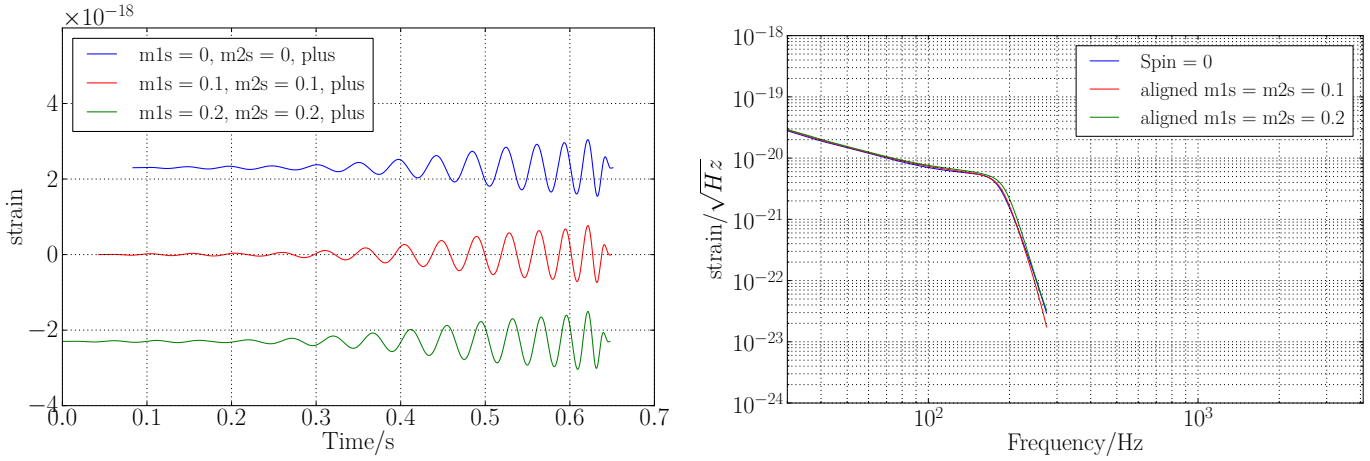
(a) (b)

FIG. 27. (a) Time domain and (b) frequency domain waveforms with varying mass ratio. For a more massive binary, the reduced mass is smaller. A higher mass ratio system has a smaller strain. Besides, the energy loss of the system due to the gravitational waves is proportional to the reduced mass. Therefore, a higher mass ratio system end the coalescence quicker with a lower frequency.



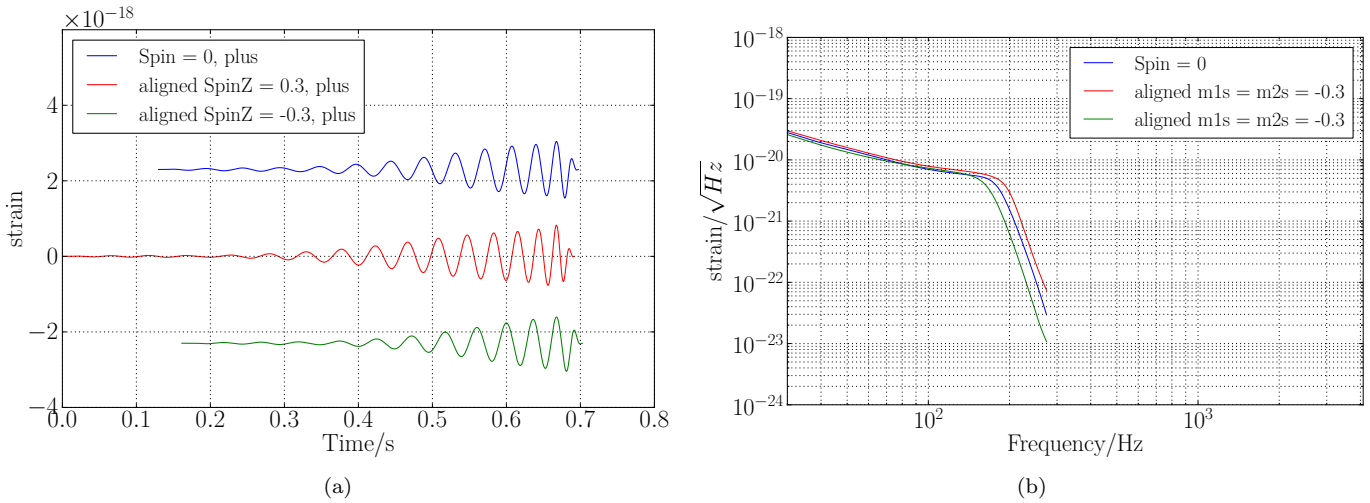
(a) (b)

FIG. 28. (a)Time domain and (b) frequency domain waveforms with varying distance. A more distant binary has a smaller strain.



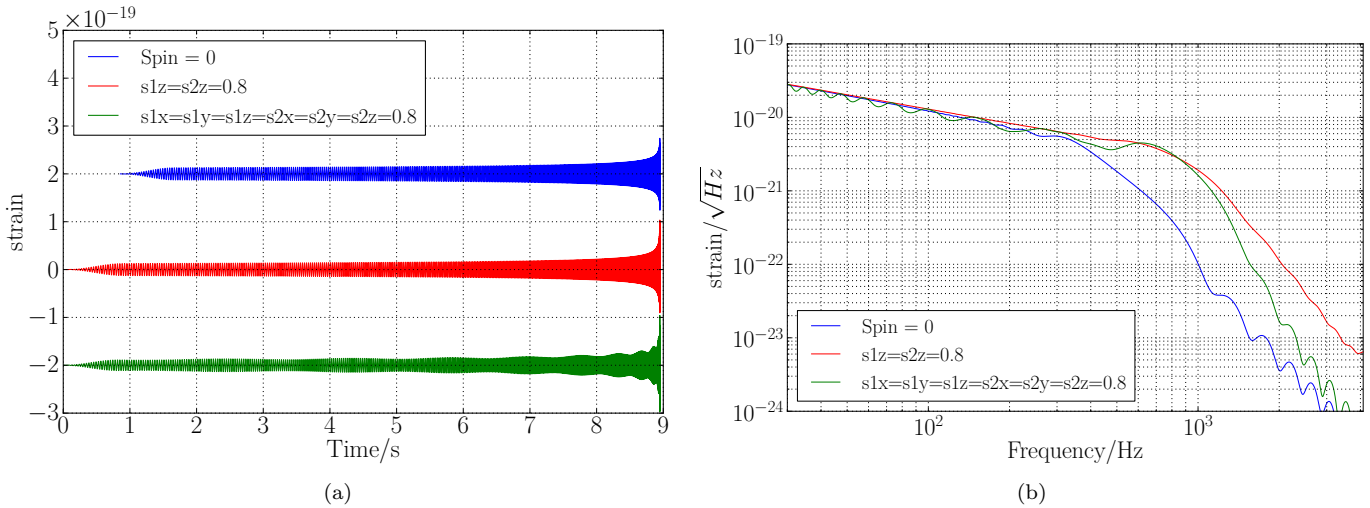
(a) (b)

FIG. 29. (a) Time domain and (b) frequency domain waveforms with varying spin. The chirp time of a larger spin system is longer. A lager spin also has a slightly increase of strain and the merger occurs at a lower frequency comparing with the system with same parameters.



(a) (b)

FIG. 30. (a) Time domain and (b) frequency domain waveforms with opposite spin. A negative spin has a smaller strain and a shorter chirp time.



(a) (b)

FIG. 31. (a) Time domain and (b) frequency domain waveforms with precessing spin. For processing spin, the amplitude modulation effect can be seen.