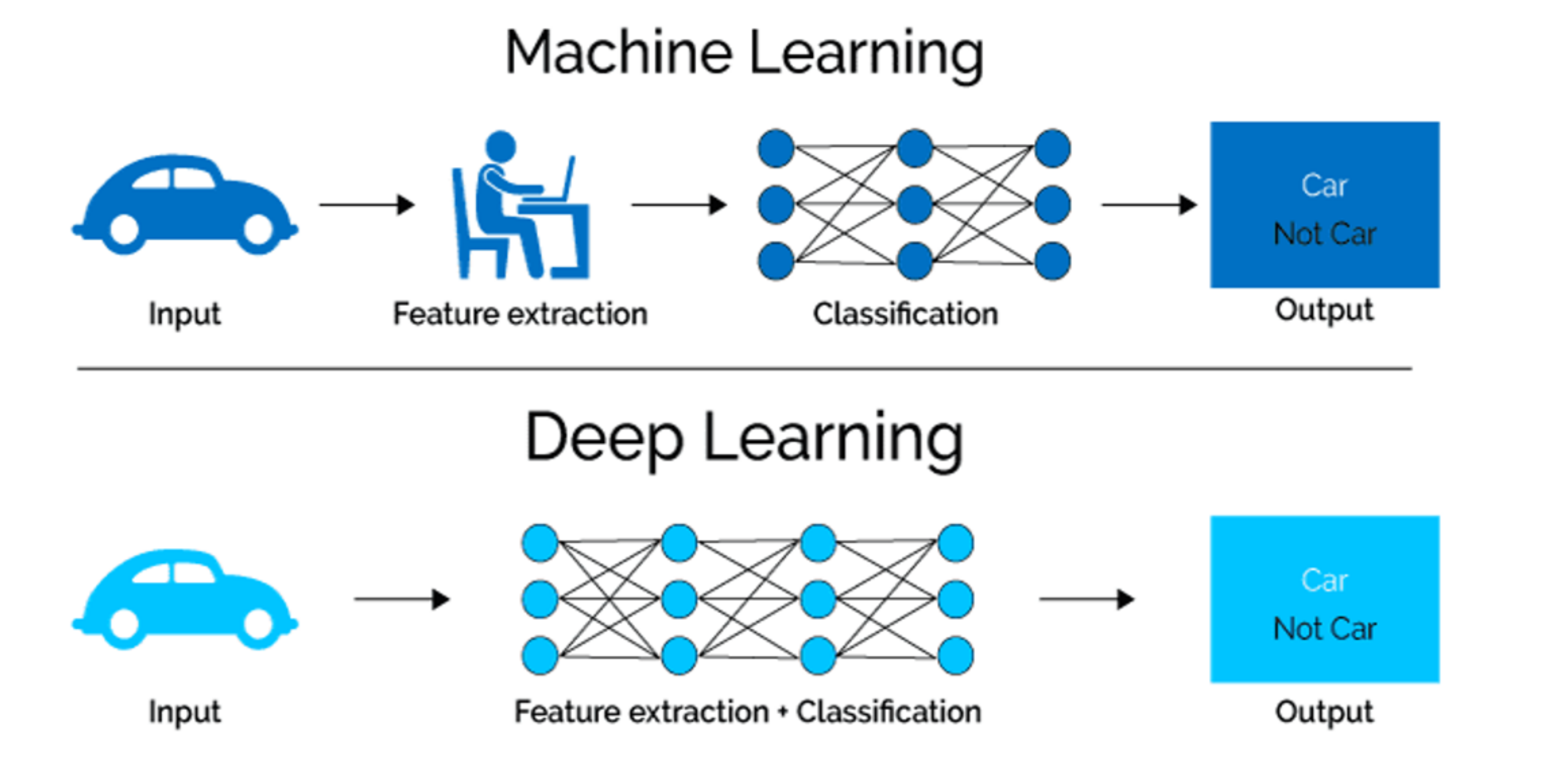
Machine-learning in gravitational wave (data) analyses

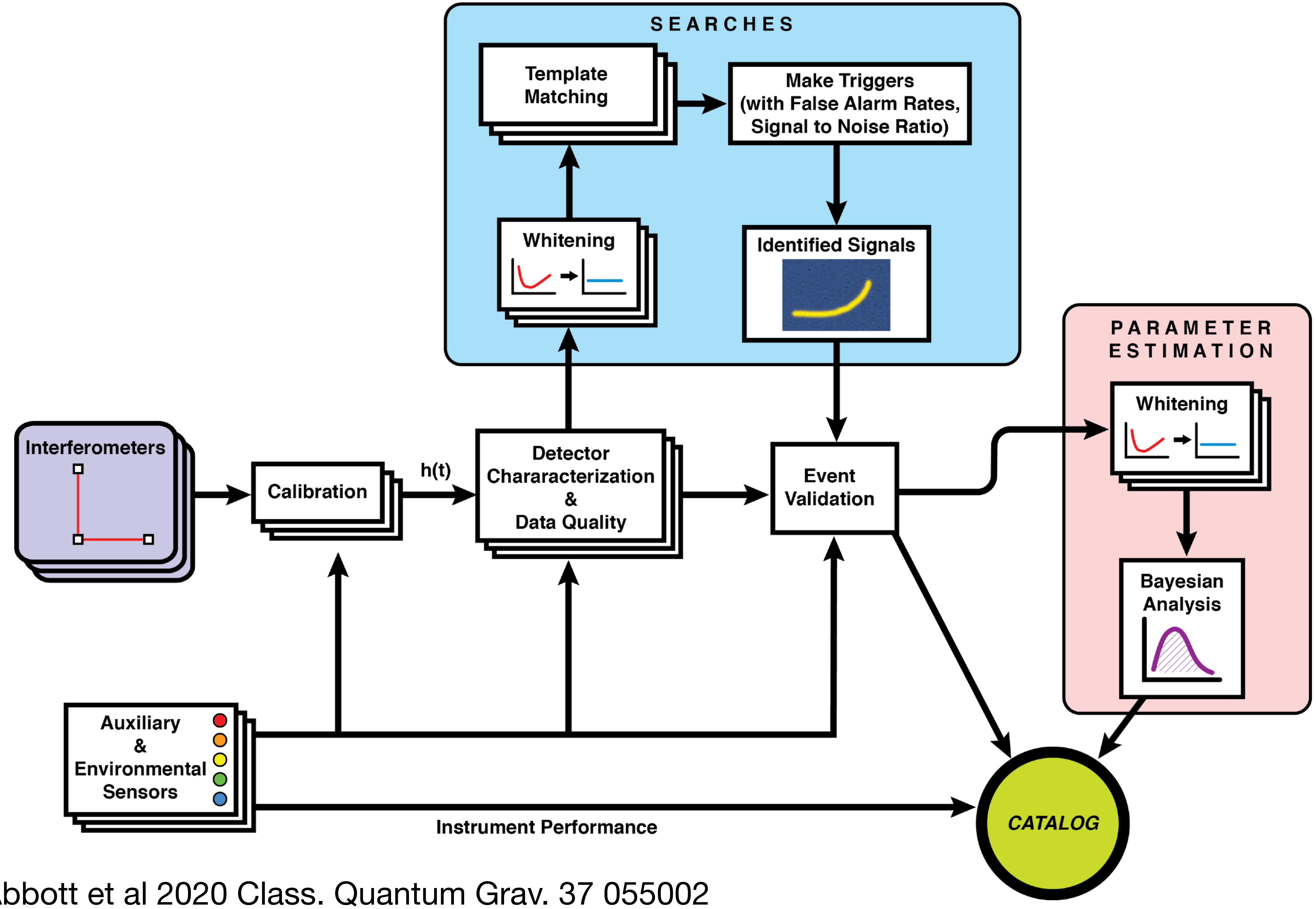
A. Trovato
Università di Trieste, INFN-Sezione Trieste

UNIVERSITÀ DEGLI STUDI DI TRIESTE
Dipartimento di **Fisica**
Dipartimento d'Eccellenza 2023-2027
INFN
Istituto Nazionale di Fisica Nucleare

Machine-learning

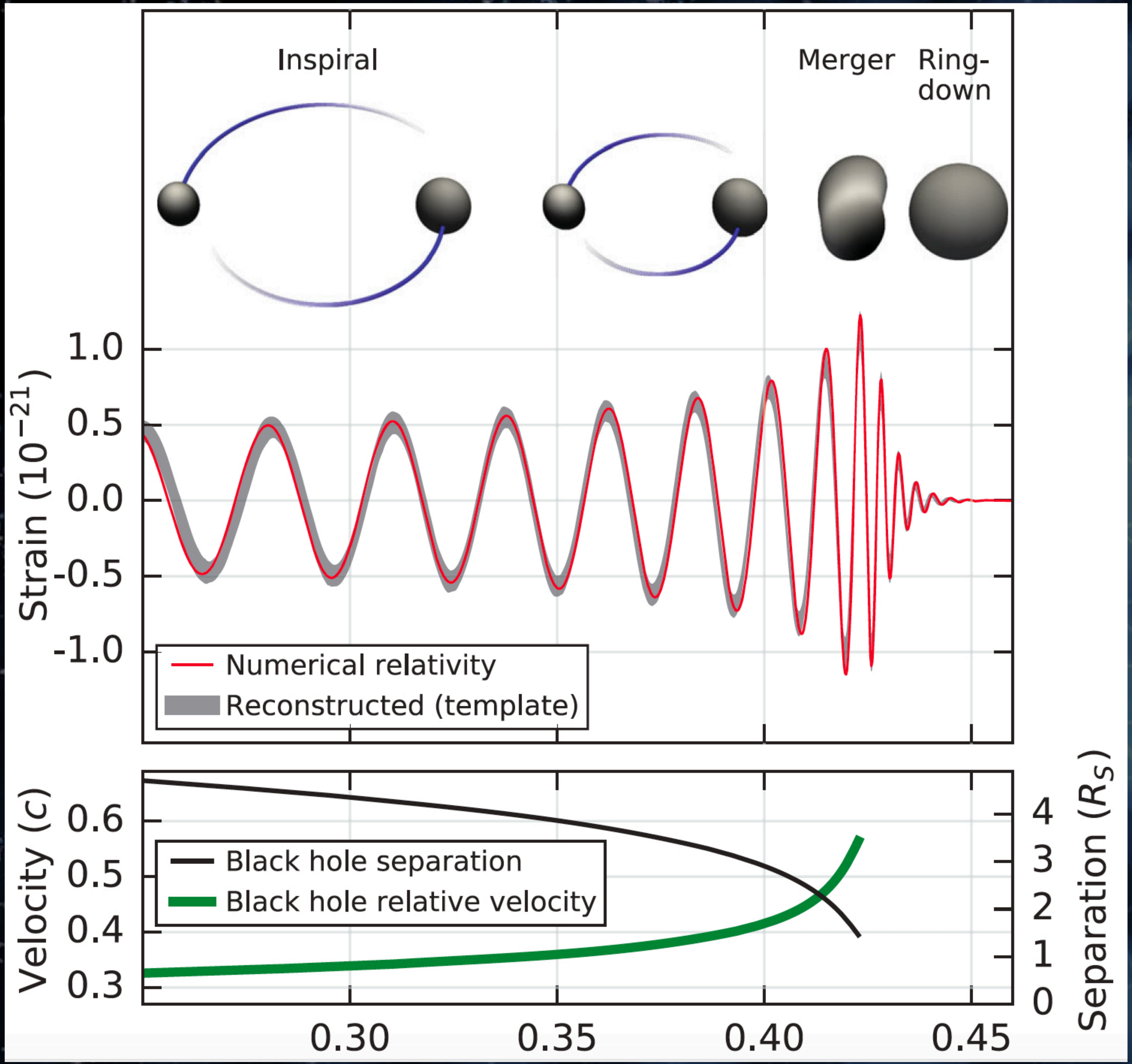


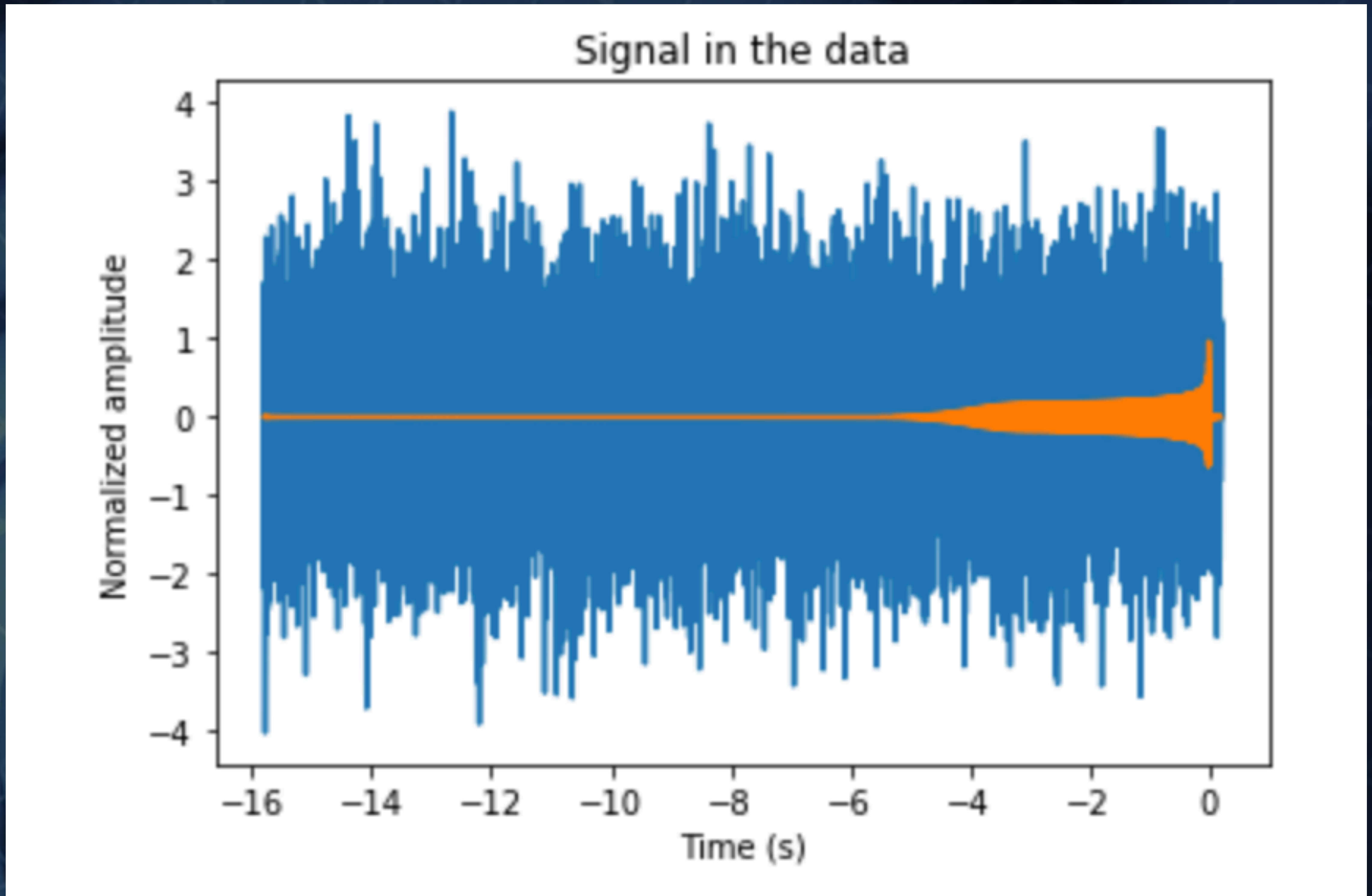
Typical GW analysis workflow



B P Abbott et al 2020 Class. Quantum Grav. 37 055002

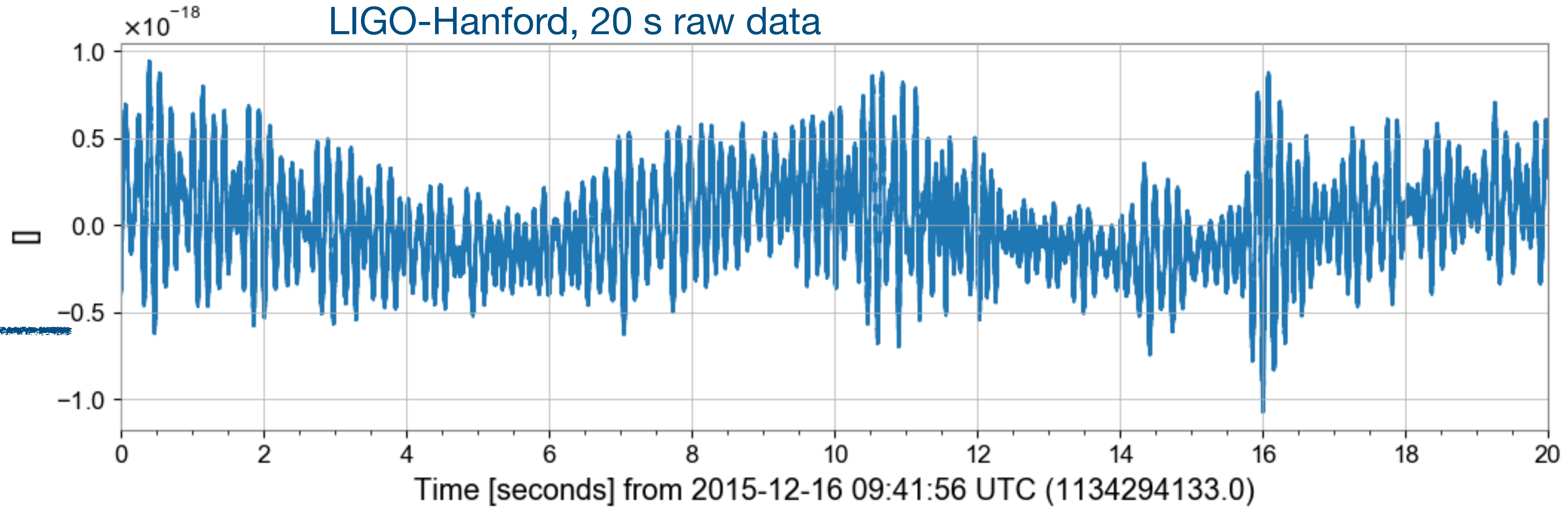
Gravitational waves detection problem

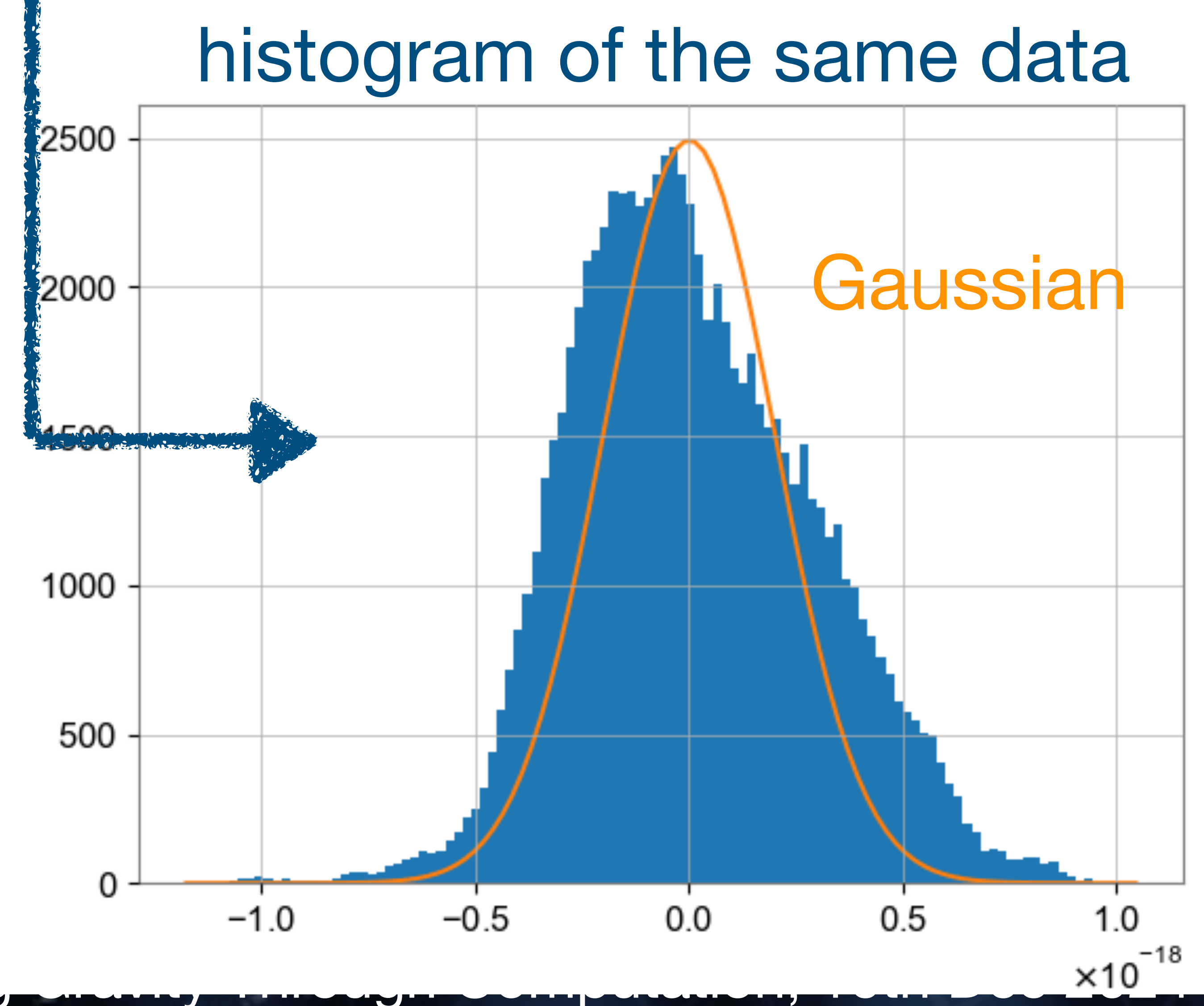




- Rare and weak signals in complex background: non-Gaussian non-stationary
- Rate of expected detections increase with the sensitivity improvement of the detectors

Non-Gaussian data



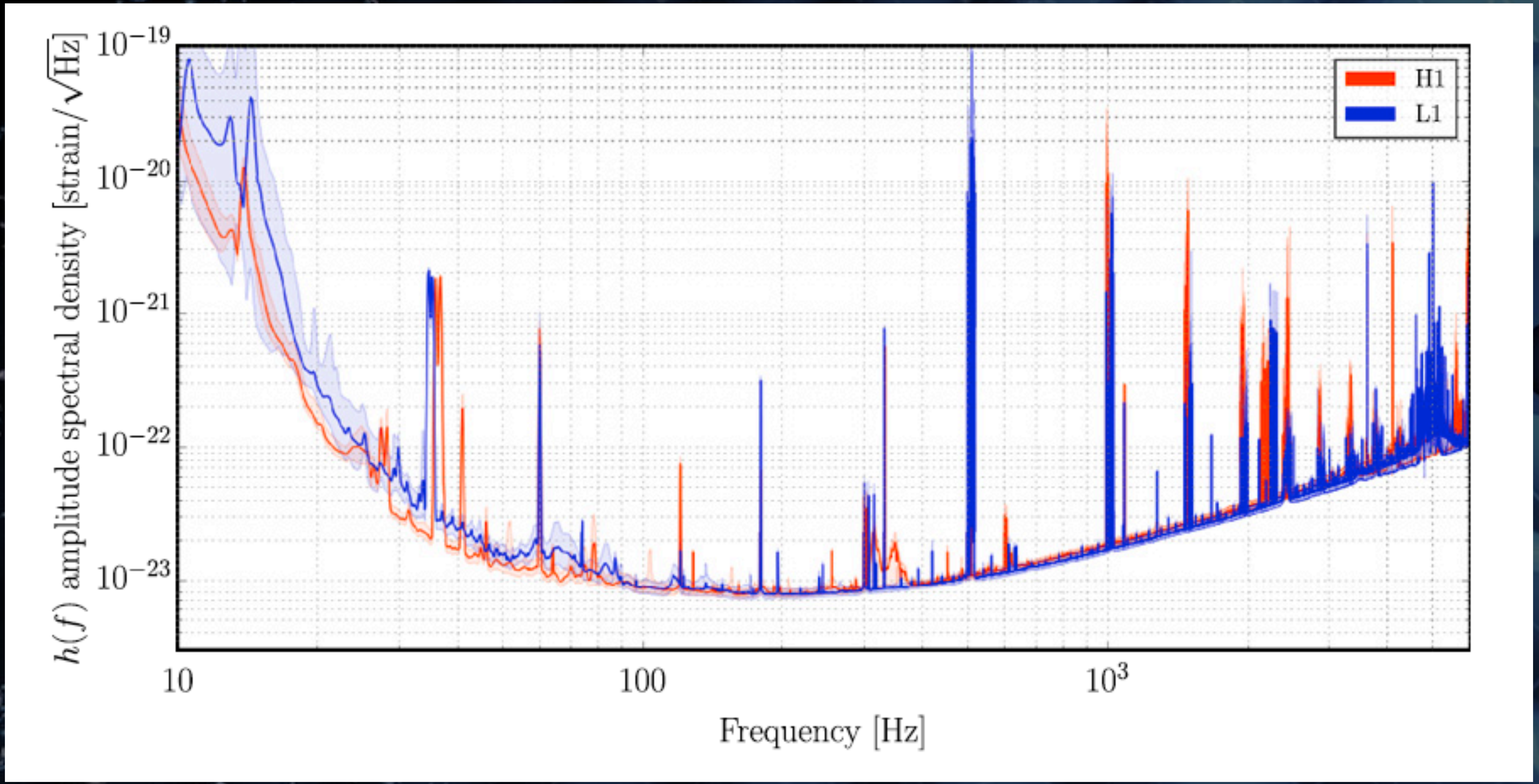


<u>The data are far from being Gaussian and stationary:</u>

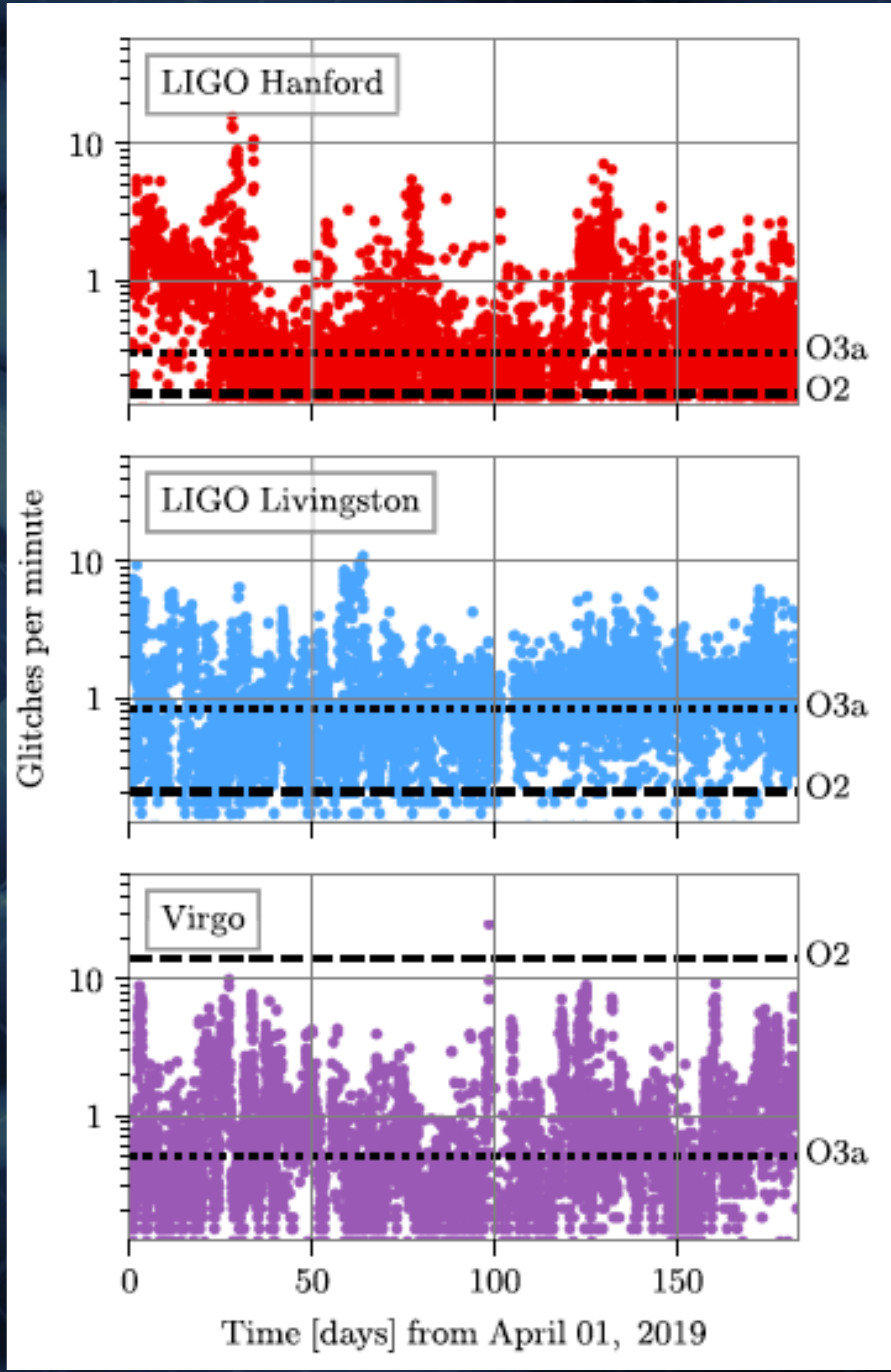
- Standard match-filter approach assume Gaussian data

Typical noise of GW detectors

Median sensitivity during O1
(shaded regions indicate the 5th and 95th percentile)



B P Abbott et al 2016 CQG 33 134001



R. Abbott et al. PHYS. REV. X 11, 021053 (2021)

ML in GW data analyses

- ML applied in all sorts of data analyses
- Impossible to summarise everything!
- On line page to collect papers about this subject: https://iphysresearch.github.io/Survey4GWML/
  - ✓ Not official repository but good representation
  - ✓ About **350 papers** (great part of the last 5 years)
- ML in GW data analysis also topic of EU COST actions (e. g. https://www.g2net.eu/)
- Kaggle competitions
  - ✓ https://www.kaggle.com/c/g2net-gravitational-wave-detection/
  - ✓ https://www.kaggle.com/competitions/g2net-detecting-continuous-gravitational-waves

1. Conferences & Workshops

2. General Reports & Reviews

3. Improving Data Quality

Glitch Classification

Glitch cancellation / GW denosing

4. Compact Binary Coalesces (CBC)

Waveform Modelling

Signal Detection (BBHs)

Parameter Estimation (PE)

Population Studies

5. Continuous Wave Search

6. Gravitational Wave Bursts

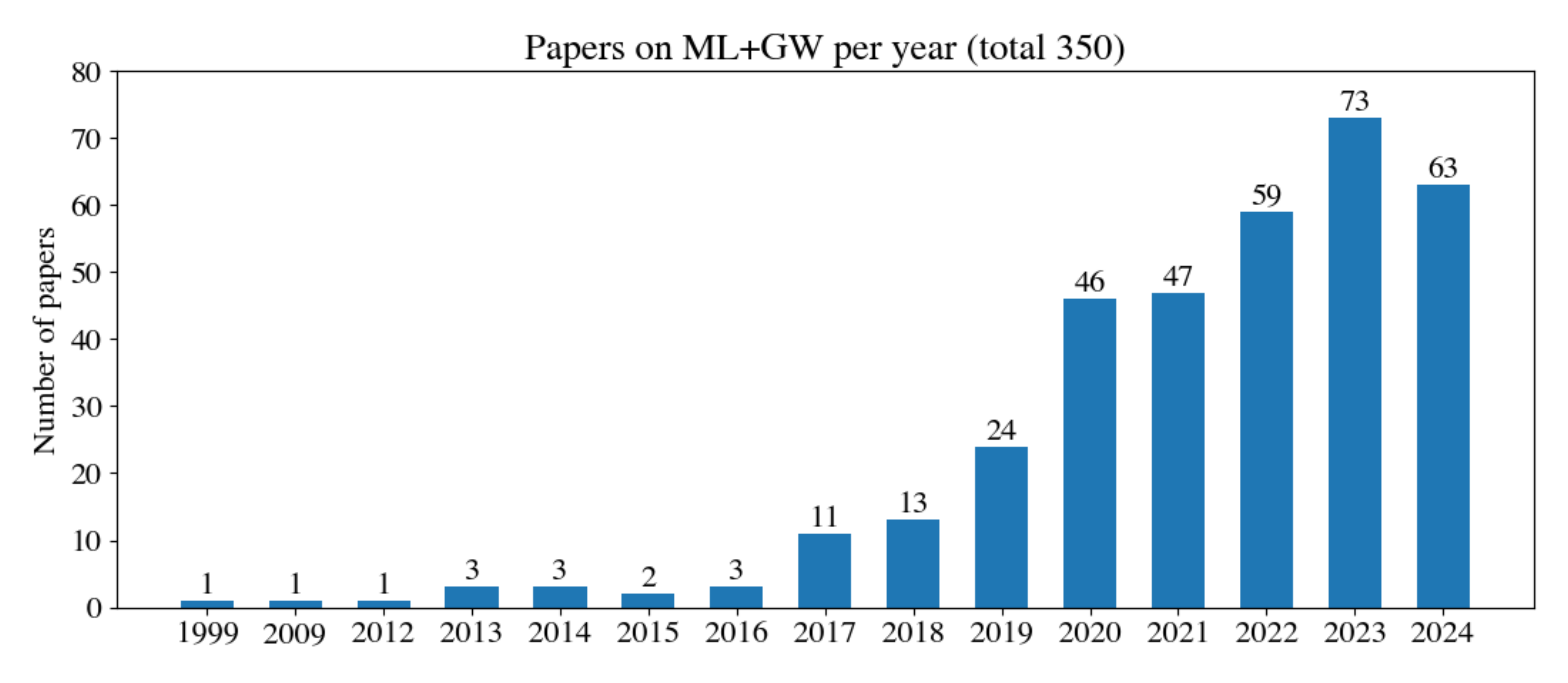
7. Stochastic Gravitational Wave Background

8. GW / Cosmology

9. Physics related

License

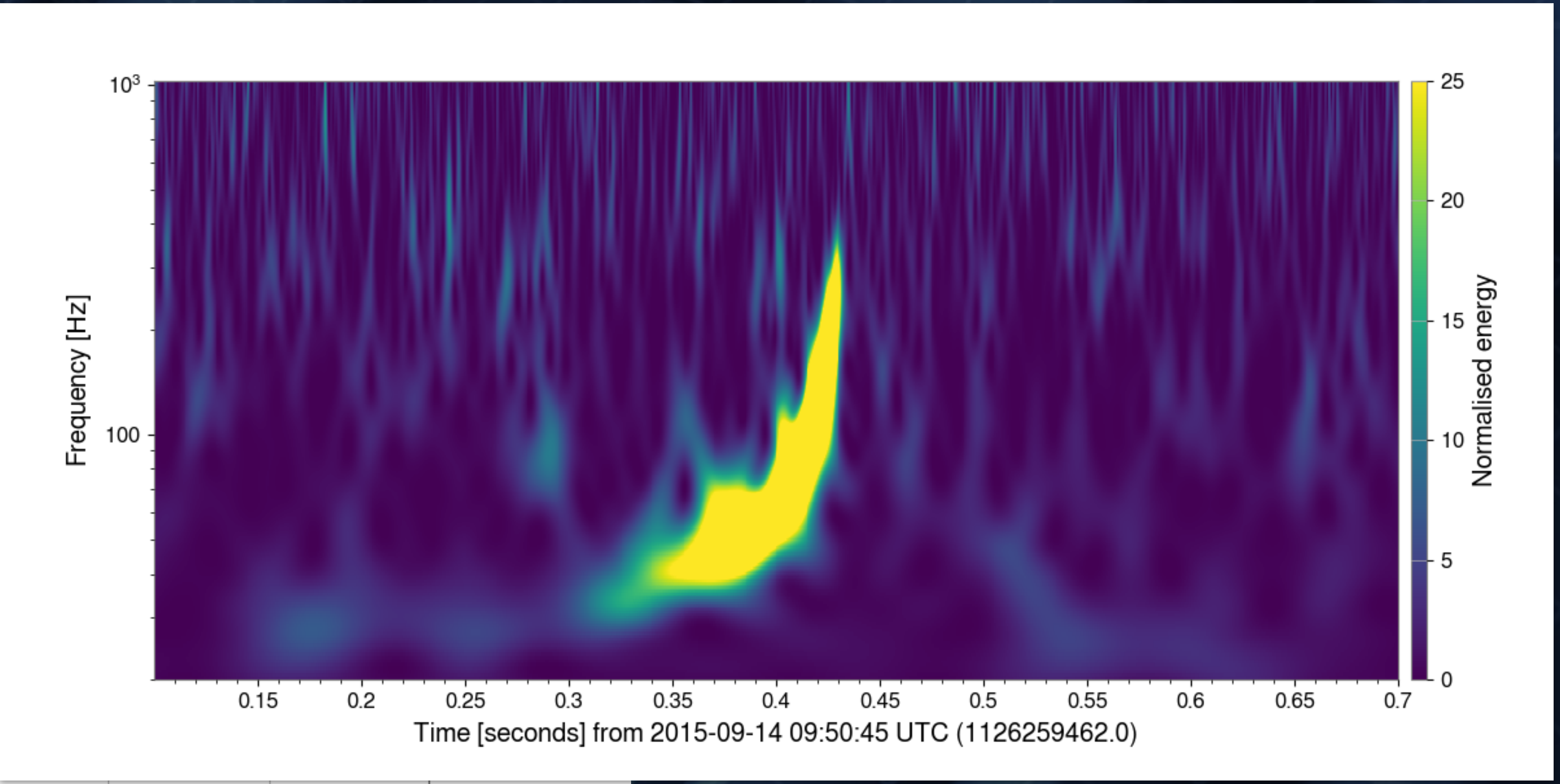
Papers per year

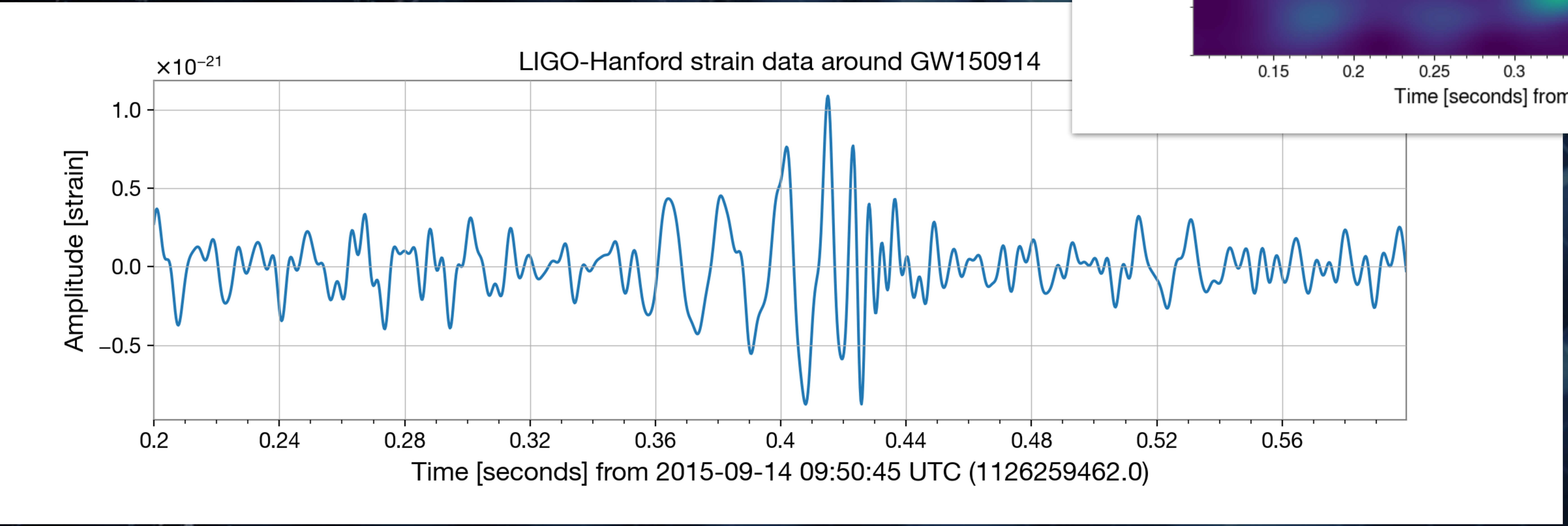


Source: https://inspirehep.net/

Data representation

- **Data representation**
  - ✓ Spectrogram vs Time series
  - ✓ Choice to make for Machine learning application



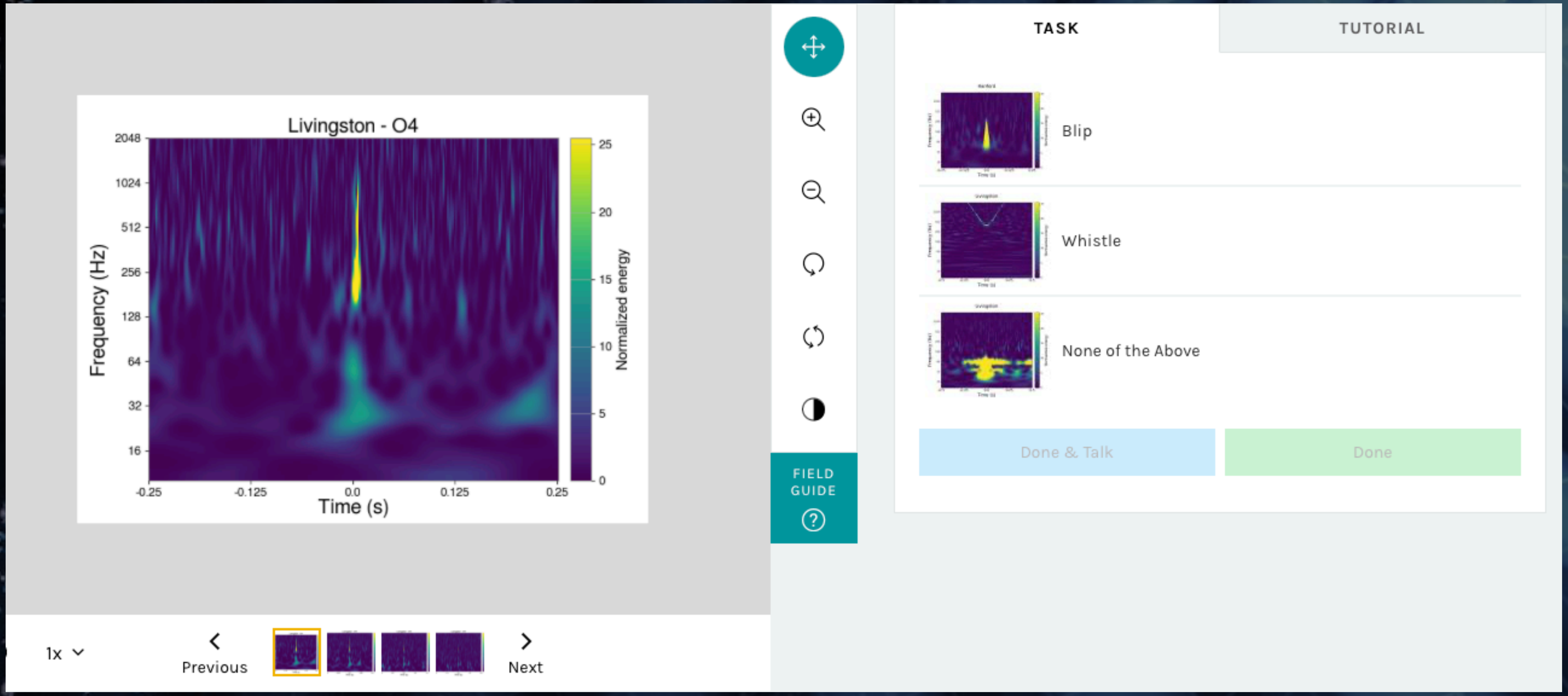


Glitch classification

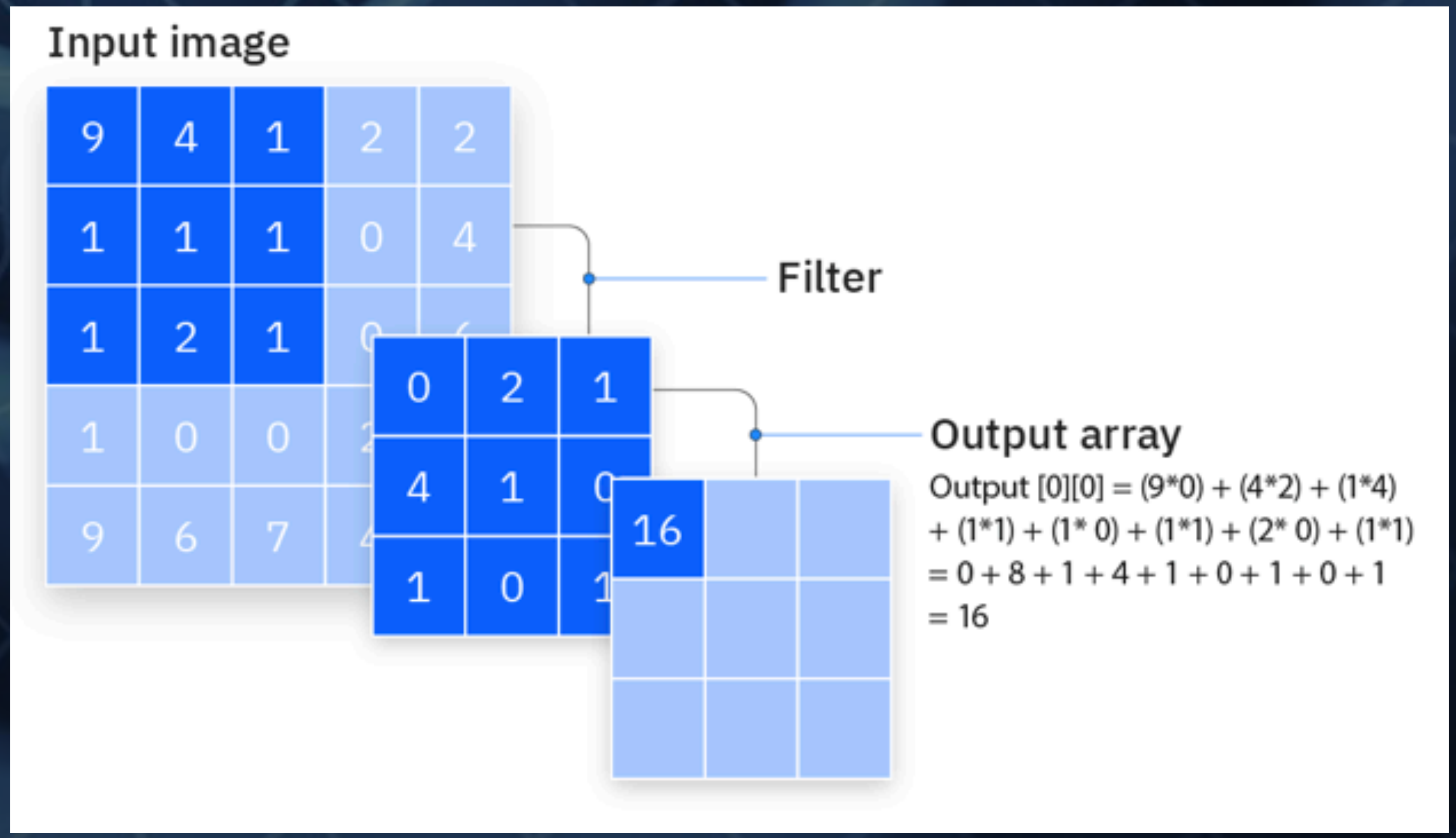
Gravity Spy

Goal: classify glitches by combining human and machine-learning classification schemes

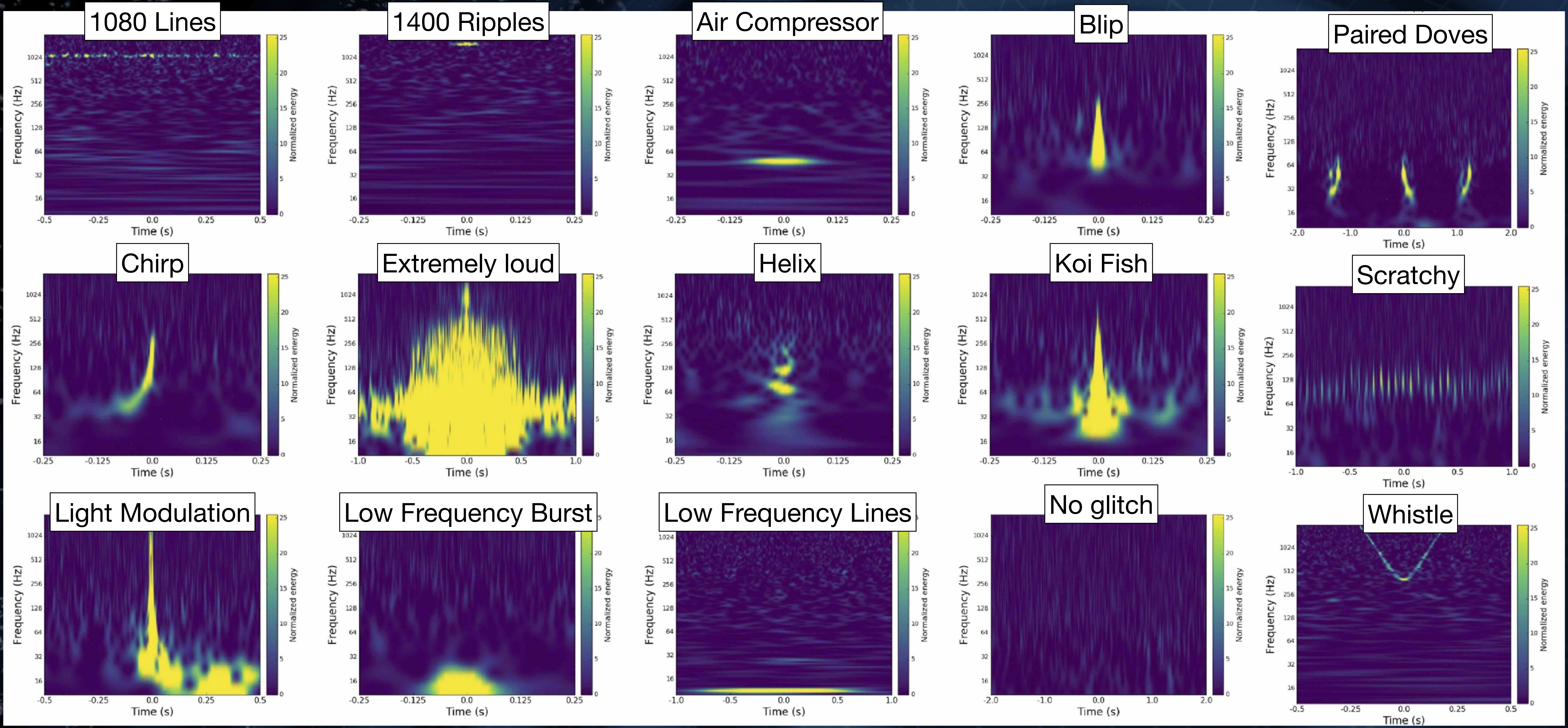
https://www.zooniverse.org/projects/zooniverse/gravity-spy



Gravity Spy uses Convolutional Neural network N, a deep-learning algorithm used primarily for image classification, to analyse data as time-frequency maps



Glitches zoo



⋆Credits: S. Bahaadini, et al., Information Sciences 444 (2018) 172

See also zenodo: https://zenodo.org/records/5649212

Explainable artificial intelligence

- Reference: N. Koyama et al. 2024 Mach. Learn.: Sci. Technol. 5 035028
- Convolutional neural network model to classify glitches using spectrogram images from the Gravity Spy O1 dataset.
- Class activation mapping for visualising influential regions in input images that contribute to specific predictions.

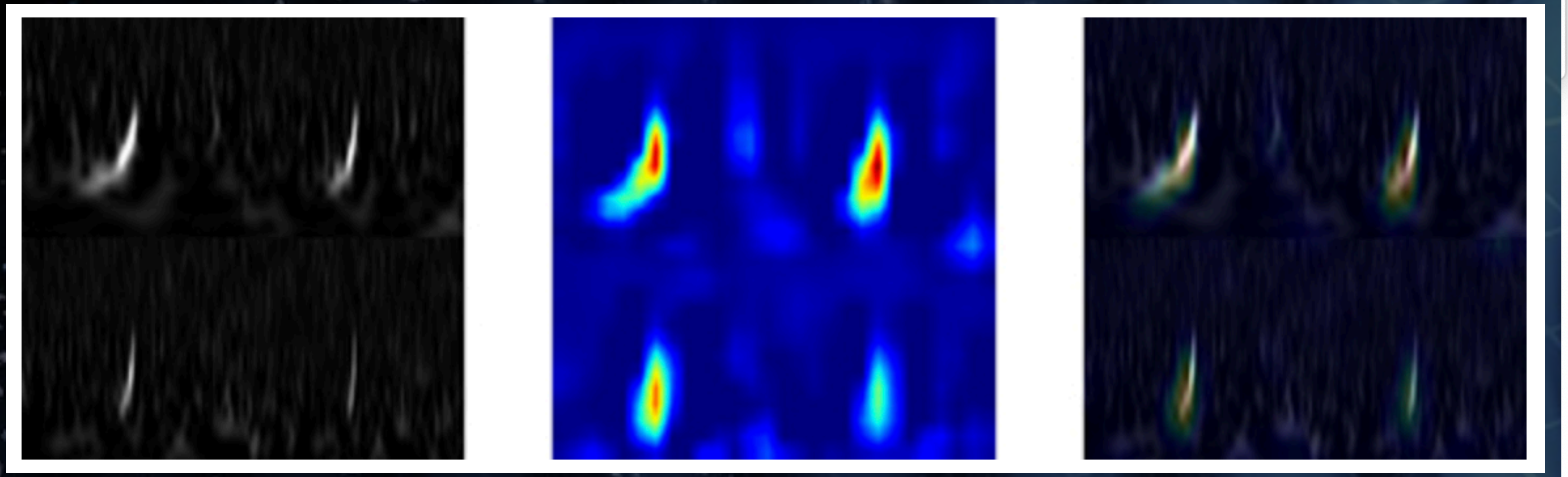


Figure 3. Estimation rationale of a correctly classified 'Chirp' sample. The figure comprises an input image (left); an estimation rationale heatmap (centre) is obtained from Score-CAM using the input image and backpropagated to the 'Chirp' Softmax output; the overlapping picture (right) highlights the coincident region between the input and the heatmap.

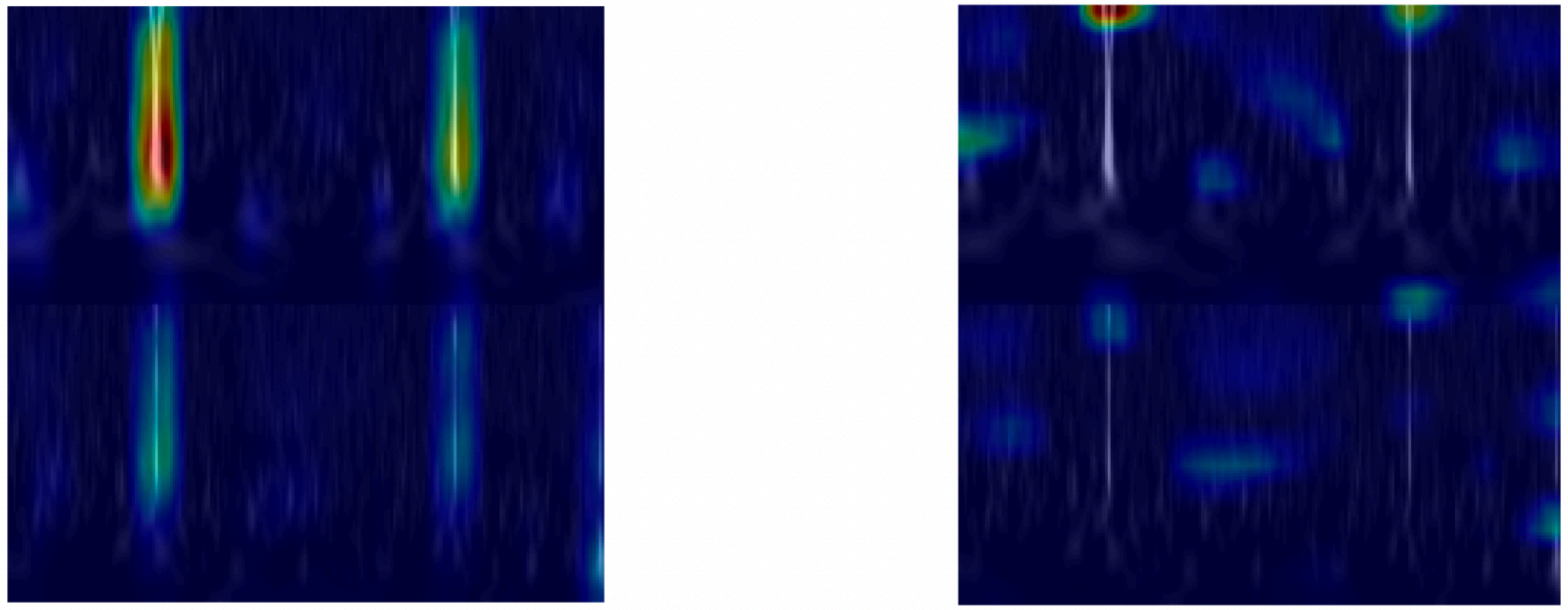
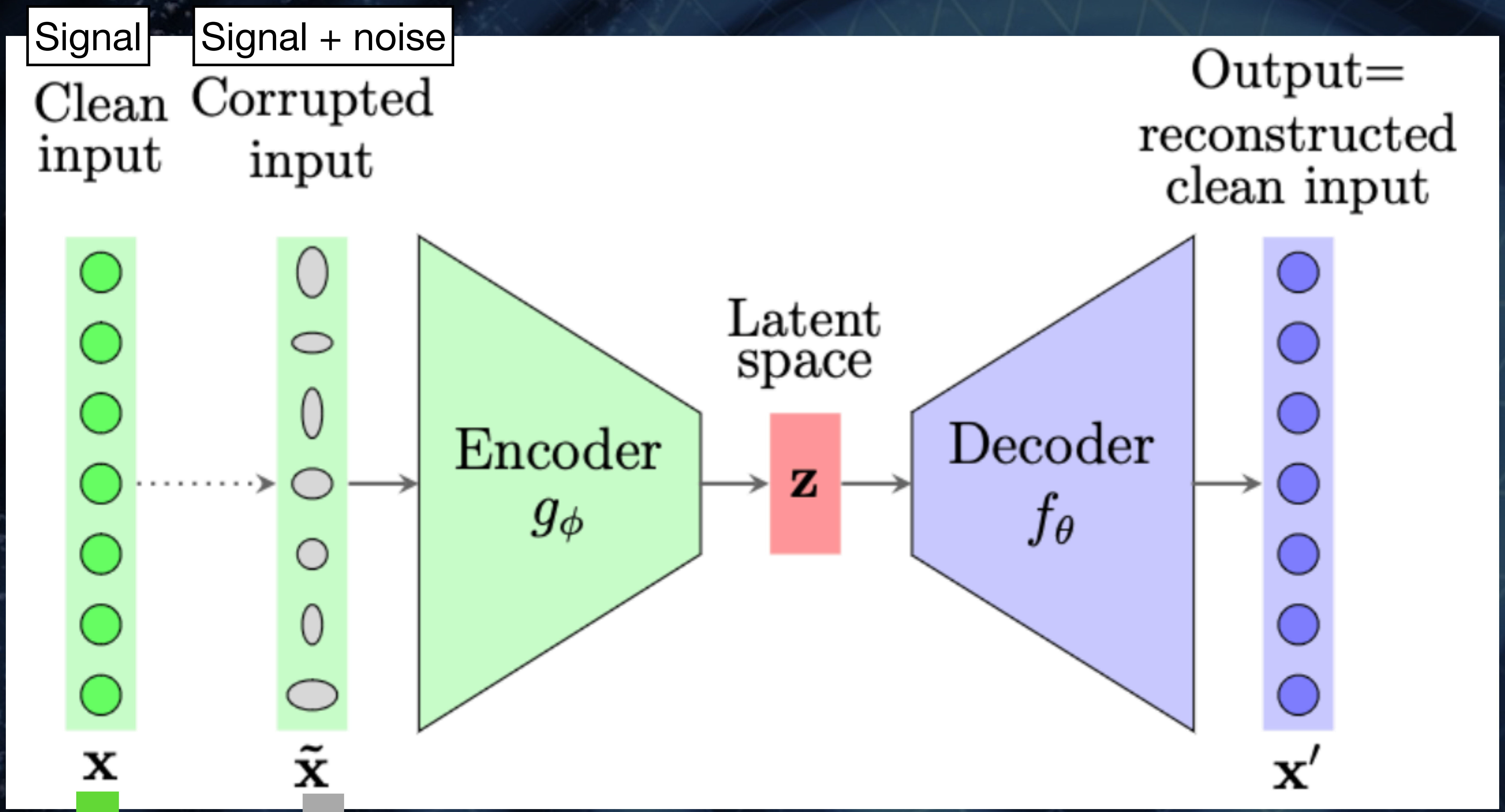


Figure 7. Left: Heatmap of the overlap image (input image and the estimation rationale) when "Whistle" is misclassified as "Blip". Right: Heatmap of the overlap image (input image and the estimation rationale) when "Whistle" is correctly classified as "Whistle".

Data denoising

Denoising autoencoder based on CNN

- Denoising: model that take noisy signals and return clean signals

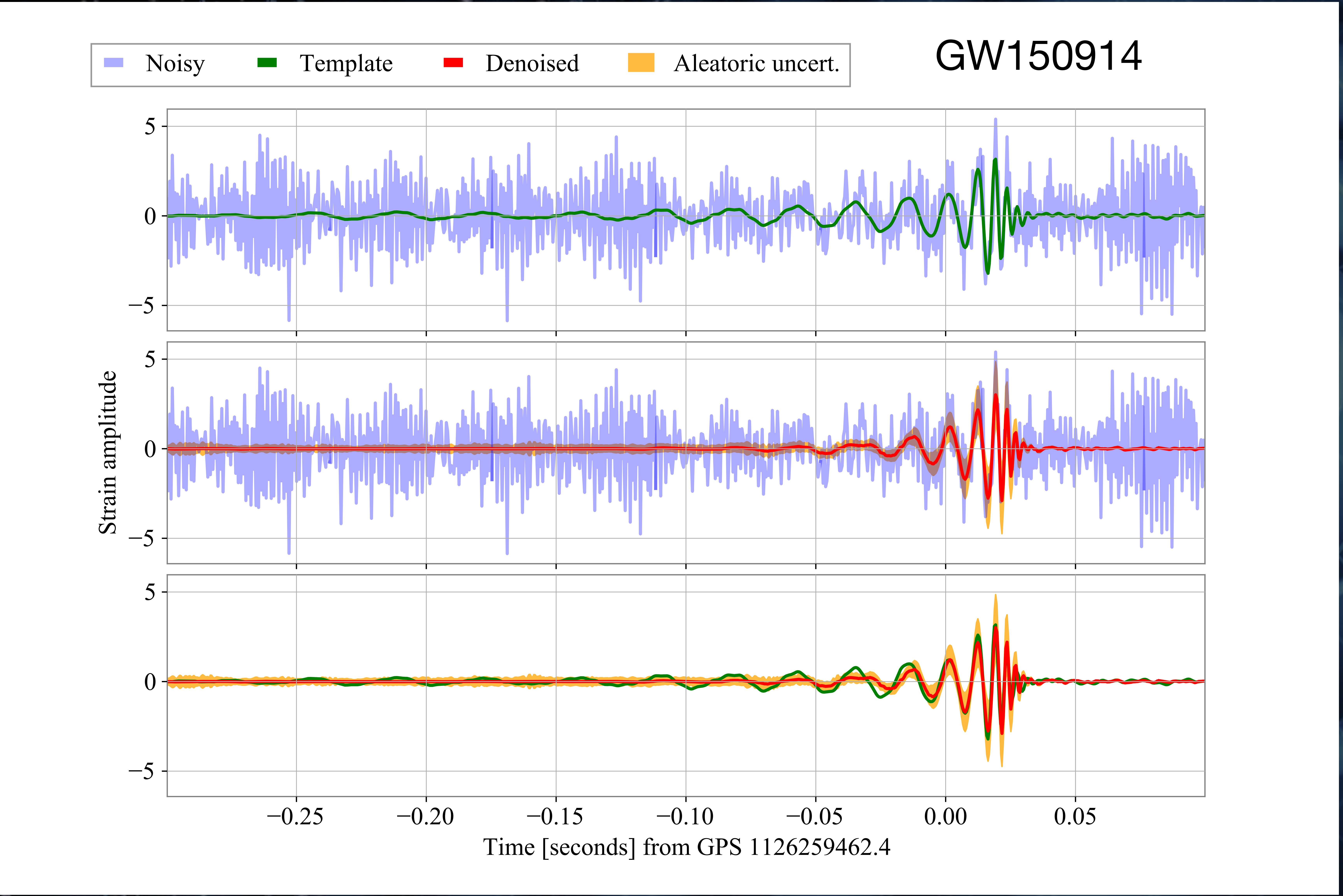


$$L_{DAE}(\theta, \phi) = \sum_{i=1}^{N} (x_i - f_\theta(g_\phi(\tilde{x}_i)))^2$$

Econder and decoder are CNNs

Reference: P. Bacon et al. MLST 4 (2023) 035024

Denoising real events



- Denoising works quite well for events with SNR>8 and masses in the range used for training
- Training only on L1 data but works also on H1
- Works also for O2 events (not tested for O3)

Reference: [P. Bacon et al. MLST 4 (2023) 035024]

Binary Black Hole signal detection

First example

- Reference: A Trovato et al 2024 Class. Quantum Grav. 41 125003
  - ✓ Classification of segments of data
  - ✓ Time-series representation
  - ✓ Training on real data
  - ✓ Focus on single detector periods
    - Glitch impact on sensitivity is larger during single-detector periods as coincidence with additional detector is impossible. Can machine learning help?
    - Single-detector time (~ 30% of the time when only the two LIGO take data or ~3% when also Virgo takes data): ~2.7 months in O1+O2; ~1.6 months in O3: ~ 2.4 months in O4a
  - ✓ Analysis of L1 single detector periods in O1

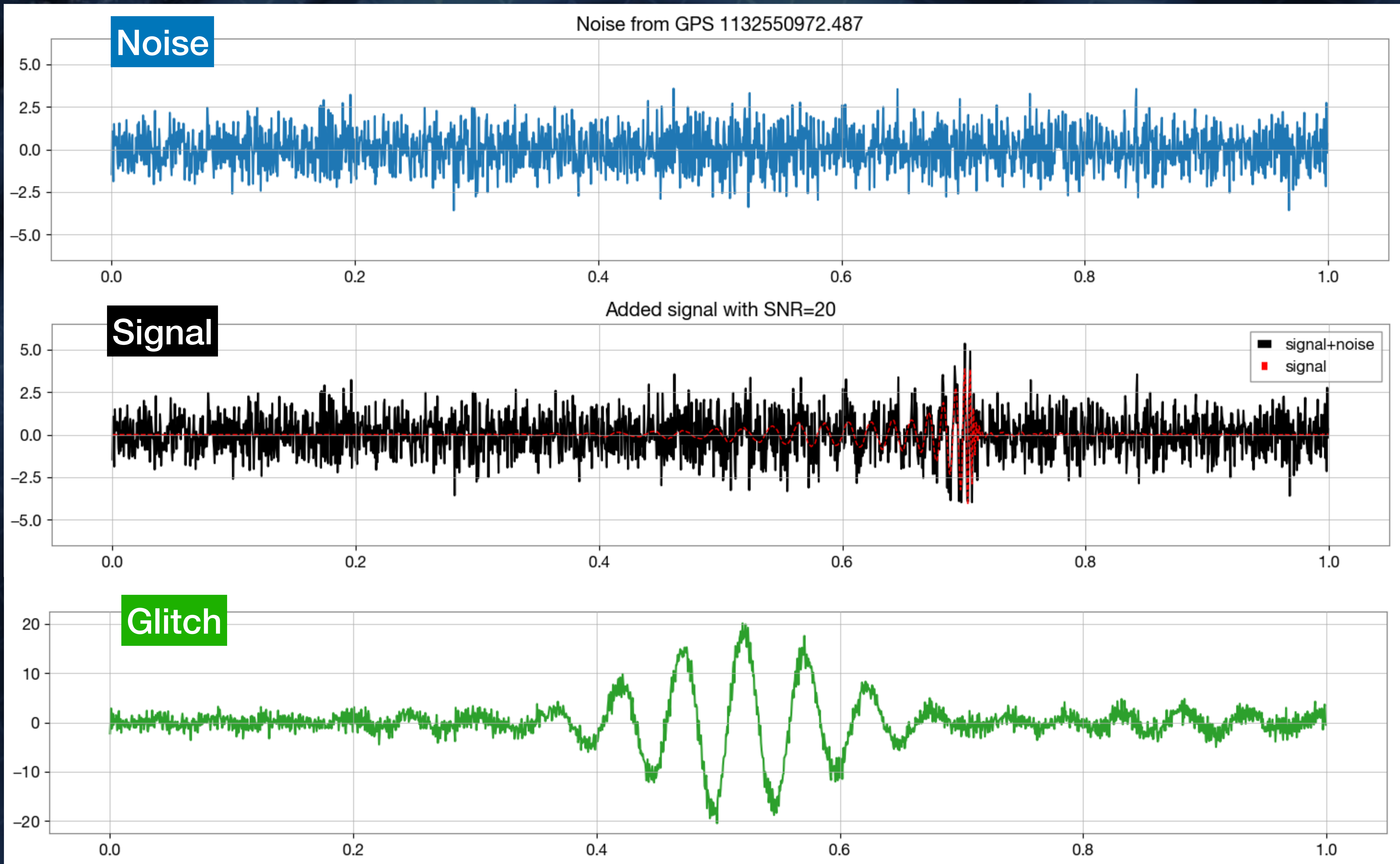
Training data: 3 classes

Segments of glitches and "almost Gaussian" noise data samples from the one month of LIGO O1 run (downsampled to 2048 Hz), whitened by the amplitude spectral density of the noise.

Real detector noise from real data when nor glitches nor signals nor injections are present

Real detector noise (selected as noise class) + BBH injections

Data containing glitches (glitches inferred from 2+ detector periods with gravity spy and cWB)



Summary of this paper

Input time series data

↓

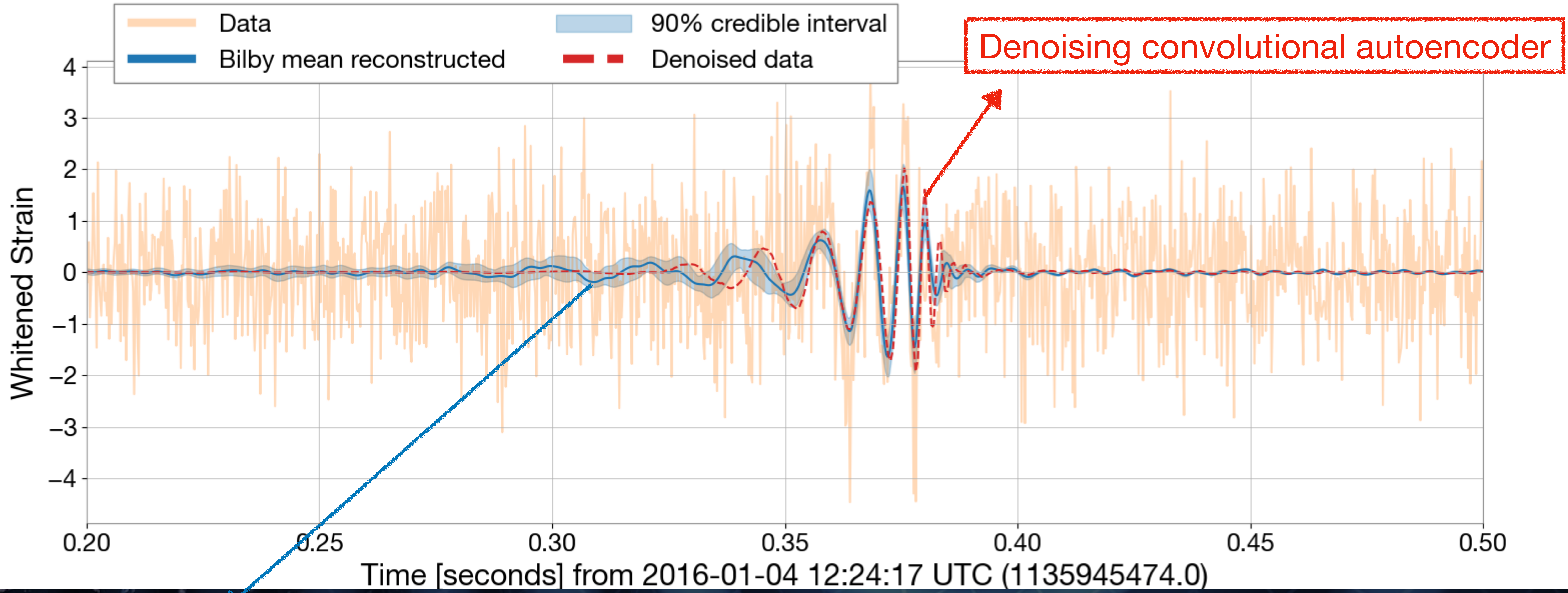
Neural network

↓

Probability for each of the three classes

- 3 NN architectures:
  - CNN : Convolutional Neural Network
  - TCN : Temporal Convolutional Network
  - IT : Inception Time
- Focus on the stricter cut possible: **P_s=1** at machine precision (single-precision floating-point format)
- Applied the 3 networks to the remaining 3 months of L1 in O1 excluding the 1 month period already used for training and testing and know injections
- Found one event common to the three analyses: L1-only at GPS=1135945474.0 (2016-01-04 12:24:17 UTC)

Bilby reconstruction



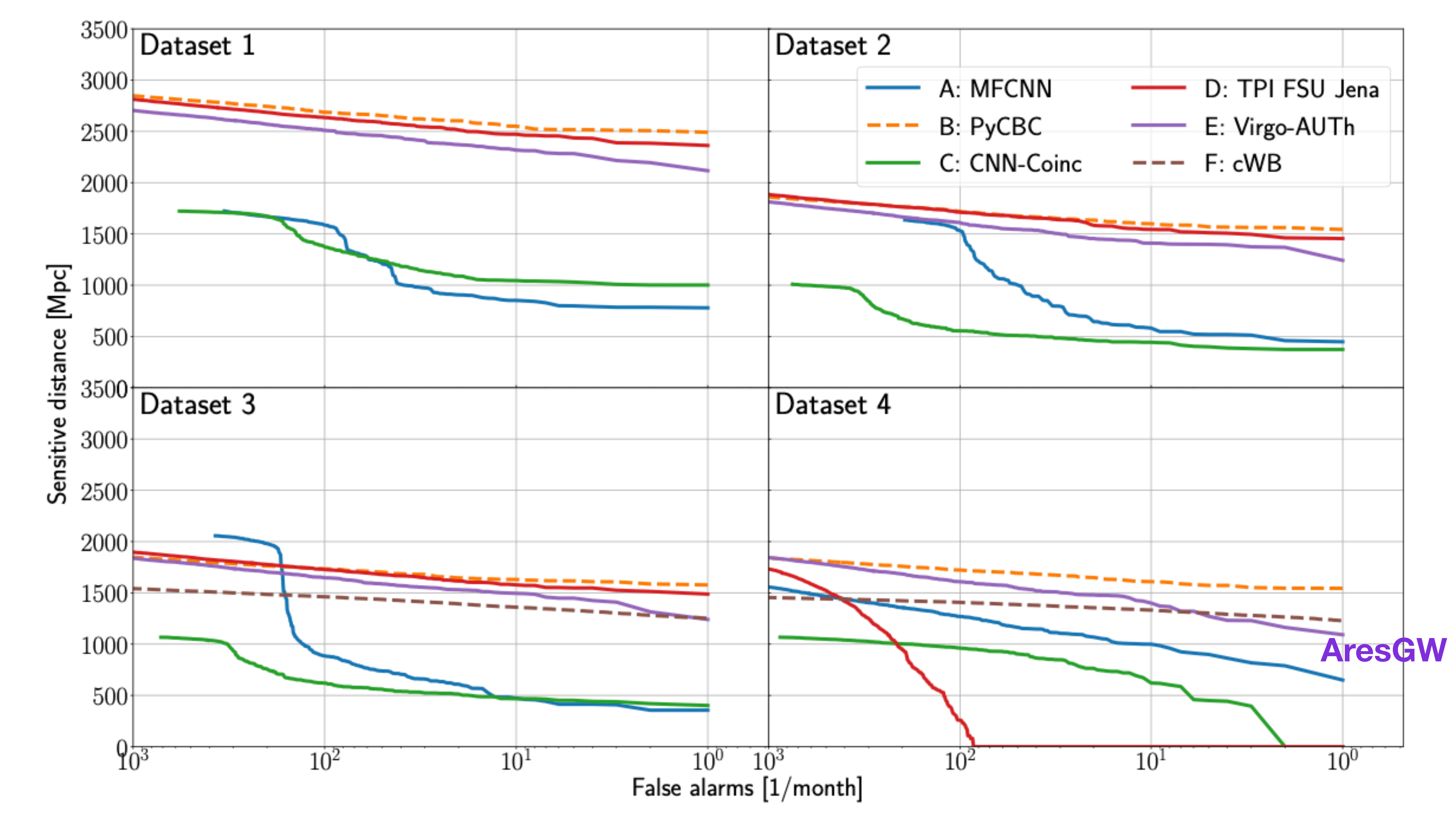
- Parameters consistent with BBH population observed so far:
  $SNR = 11.34^{+1.8}_{-1.6}, \mathscr{M} = 30.18^{+12.3}_{-7.3} M_\odot, m_1 = 50.7^{+10.4}_{-8.9} M_\odot, m_2 = 24.4^{+20.2}_{-9.3} M_\odot$

Other example: Kaggle competition

- Lots of literature on ML for BBH signal detection but results hard to compare

➡ Reference: [M. B. Schäfer et al. Phys. Rev. D 107 (2023) 023021]()

✓ Multi-detector search



AresGW improvements

- Reference: A. E. Koloniari et al.
- ResNet-based deep learning code
- hierarchical classification of triggers, based on different noise and frequency filters
- logarithmic ranking statistic $R_s = -\log_{10}(1 - R + 10^{-16})$
- eight new GW candidates in the O3 data, with pastro > 0.5

TABLE VI: New candidate events identified by AresGW.

#	Event Name	GPS Time (s)	$p_{\rm astro}$	FAR (1/yr)	$\mathcal{R}_s$	Time delay (s)	χ^2_L	χ^2_H	Class
1	GW190511_135545	1241614563.77	1.00	0.27	9.54	0.0027	1.16	1.46	Selective Passband
2	GW190614_144749	1244555287.93	0.99	4.6	5.80	0.0012	0.65	0.80	Selective Passband
3	GW190607_093827	1243931925.99	0.99	6.5	8.95	0.0056	1.03	0.37	Selective Noise Rejection
4	GW190904_114631	1251629209.01	0.72	14	4.35	0.0002	0.38	0.71	Selective Passband
5	GW190523_095933	1242637191.44	0.68	20	6.60	0.0054	0.75	1.39	Selective Noise Rejection
6	GW200208_211609	1265231787.68	0.55	18	4.0	0.0063	0.69	0.98	Selective Passband
7	GW190705_174632	1246380410.88	0.51	49	5.82	0.0103	1.05	0.98	Default Low-Pass*
8	GW190426_092124	1240302101.93	0.50	20	3.91	0.0007	1.48	0.53	Selective Passband

* This event also classified as Selective Noise Rejection, but it has the best $p_{\rm astro}$ as Default Low-Pass.

Parameter Estimation

"Standard" PE

Bayes theorem

$$p(x\,|\,y) = \frac{p(y\,|\,x)\,p(x)}{p(y)}$$

Parameters **Data**

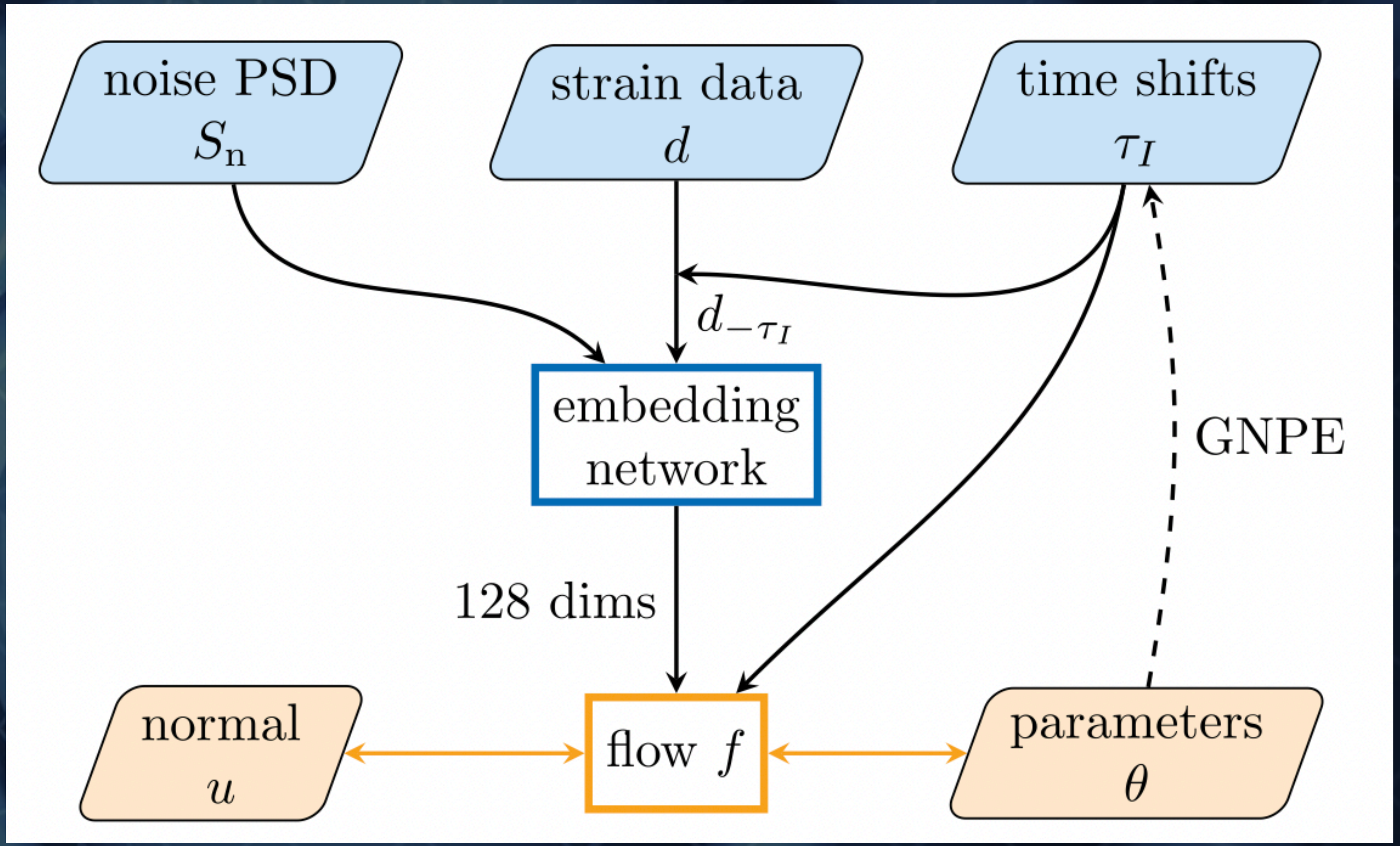
- p(y|x) = likelihood model for strain data y conditioned on system parameters x
- p(x) = prior distribution
- p(y) = evidence
- p(x|y) = posterior distribution

- Task of inference is to characterize the posterior by drawing samples from it using stochastic algorithms like Markov chain Monte Carlo (MCMC) methods
- These algorithms are computationally expensive as they require many likelihood evaluations for each independent posterior sample, and each likelihood requires a waveform simulation.
- Total inference time of hours to months, depending on the signal duration and waveform model

DINGO: Deep inference for gravitational-wave observations

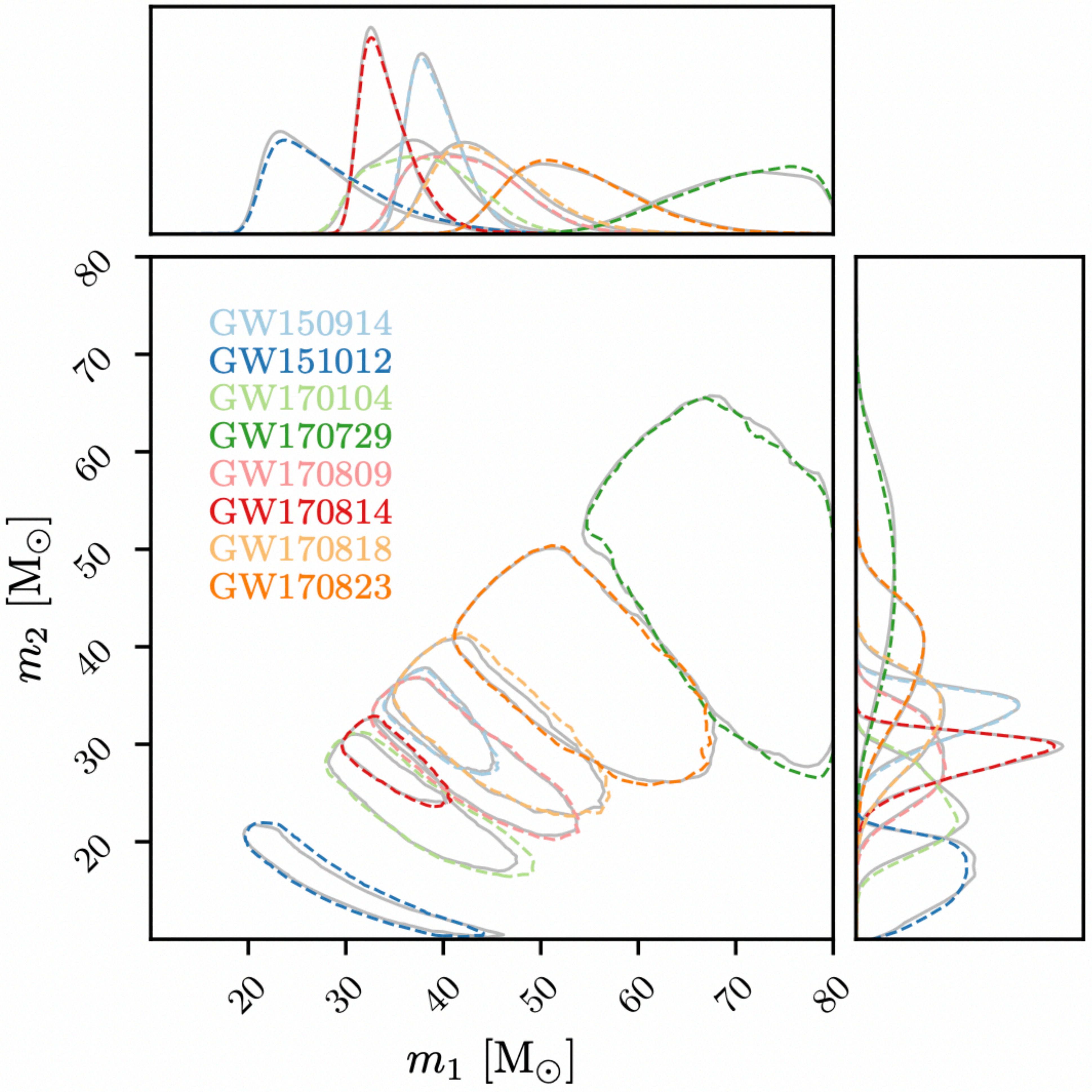
- Basic idea: produce a large number of simulated datasets (with associated parameters) and use these to train a type of neural network known as a "**normalizing flow**" to approximate the posterior
- Likelihood used to simulate the data (while for conventional methods, its density is evaluated)
- *Normalizing Flow: A technique to build up representations of complex probability distributions by learning the necessary transformations from a simpler base distribution (e.g. a Gaussian)*

M. Dax et al. PRL 127 (2021) 241103

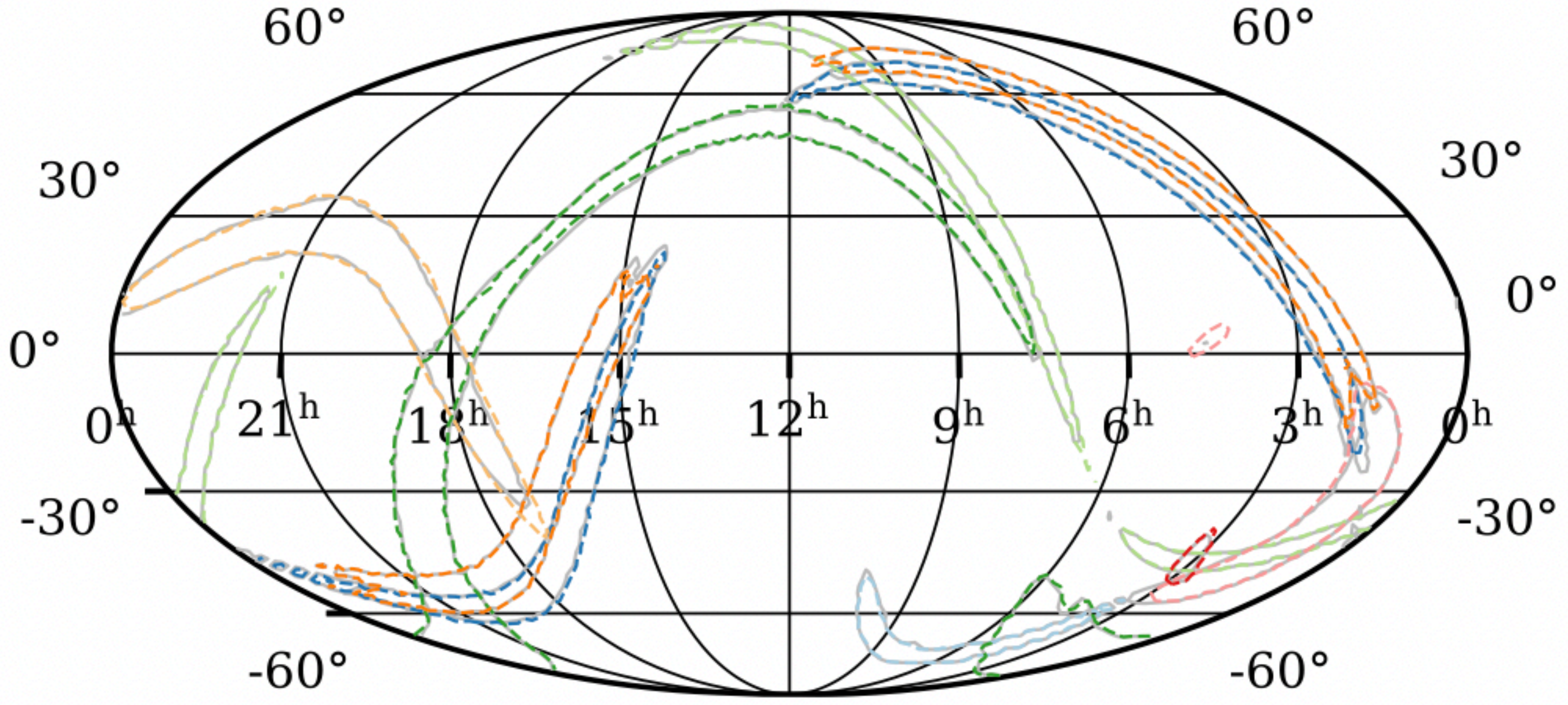


The flow itself depends on a (compressed) representation of the noise properties S_n and the data d, as well as an estimate τ_I of the coalescence time in each detector I

DINGO results



(a)



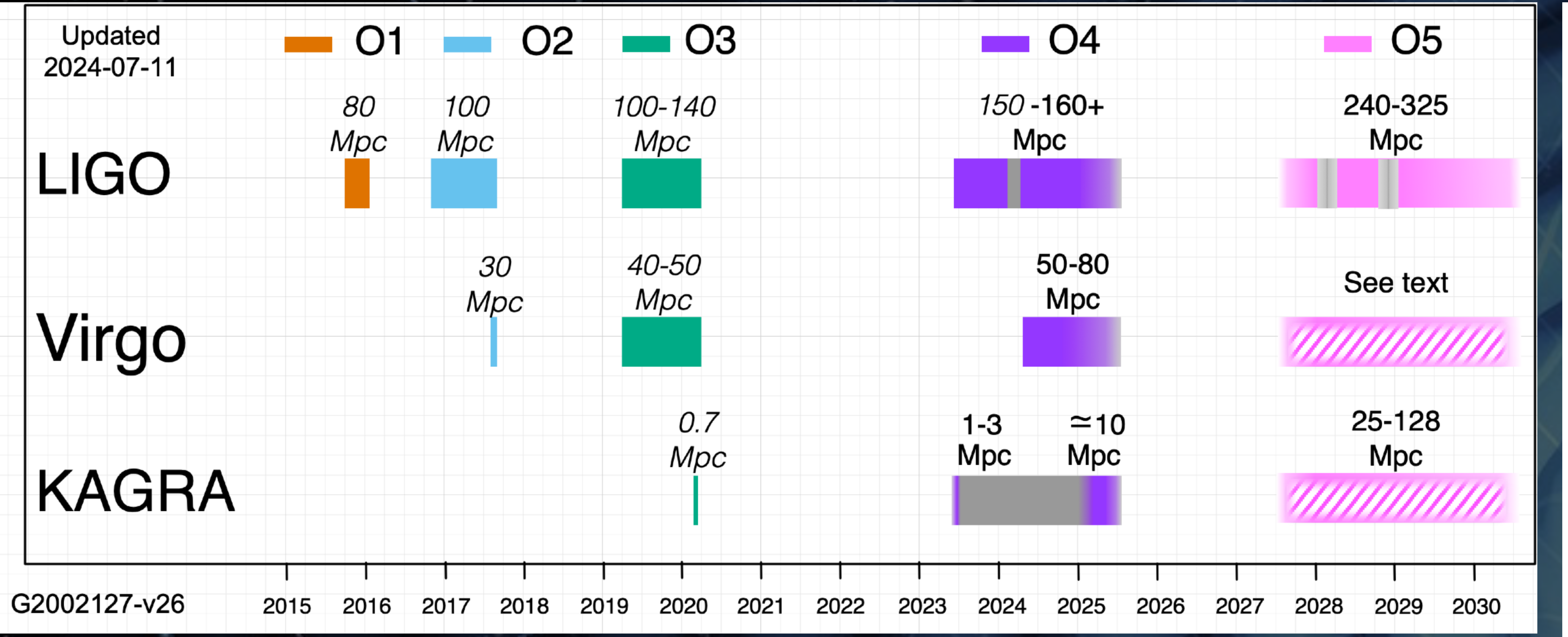
(b)

FIG. 3. Comparison of (a) detector-frame component mass and (b) sky position posteriors from DINGO (colored) and LALINFERENCE (gray) for eight GWTC-1 events. 90% credible regions shown.

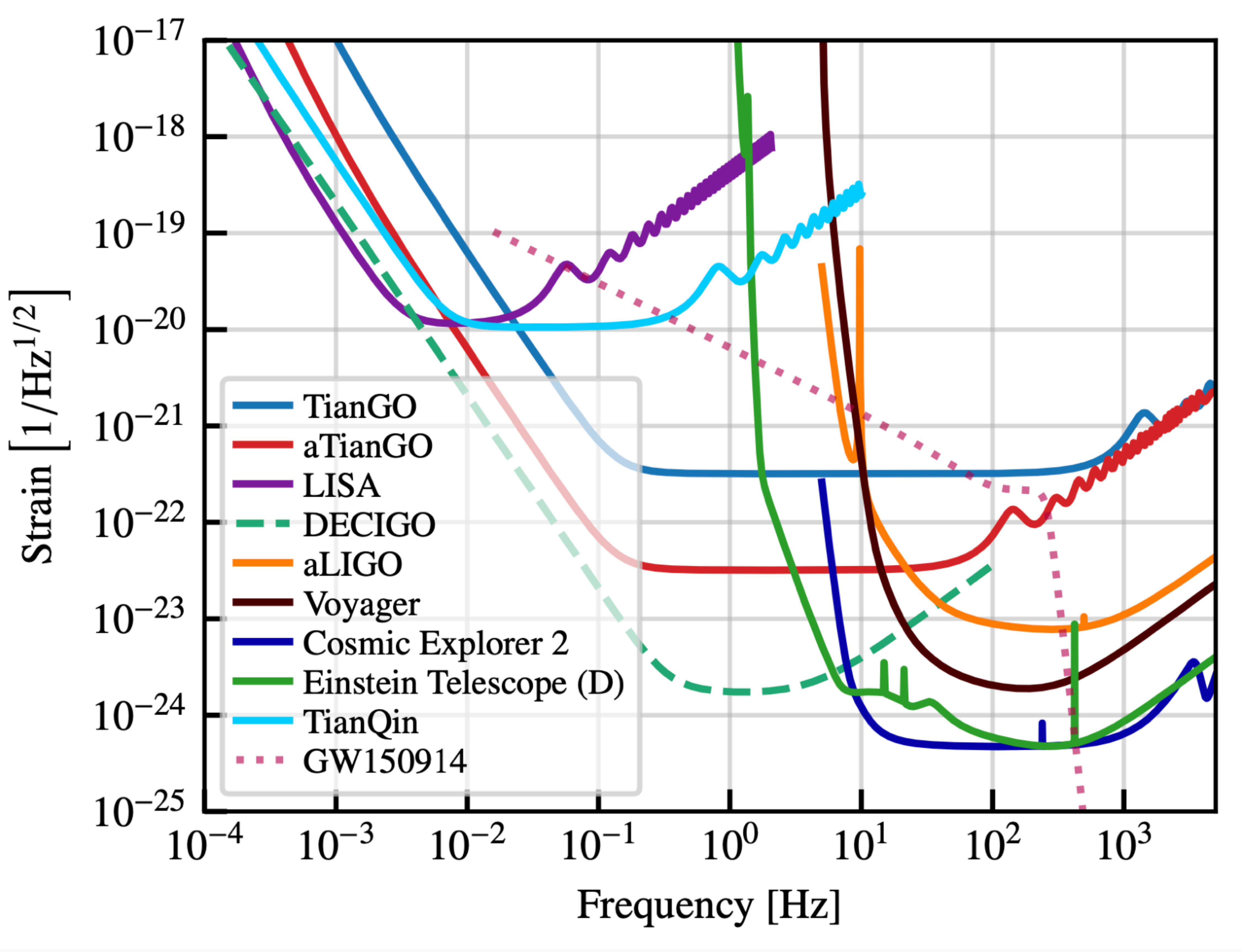
M. Dax et al. PRL 127 (2021) 241103

What we can expect in the future

https://dcc.ligo.org/LIGO-G2002127/public



K. A. Kuns, Phys. Rev. D 102, 043001 (2020)



- Great improvements in sensitivity
- New challenges in data analysis!
  - ✓ huge event rates (superposition of events)
  - ✓ longer in-band duration of CBC signals due to the lower minimum frequency
  - ✓ ML will become more prominent

Conclusion

- Lots of interest to use machine learning for GW data analysis
- Many ML models get stacked at the development stage
  - ✓ Excitement phase when you start developing but challenges in deploying, versioning, manage GPU libraries, etc.
  - ✓ This happens also outside academy, see e.g. this link
- Hard to join forces and progress on previous experience
  - ✓ Attempt to build general use frameworks exists: https://github.com/ML4GW
- Initiatives like the cost action https://www.g2net.eu/ rare
- In the future we will rely more on ML for GW data analysis!

Backup slides

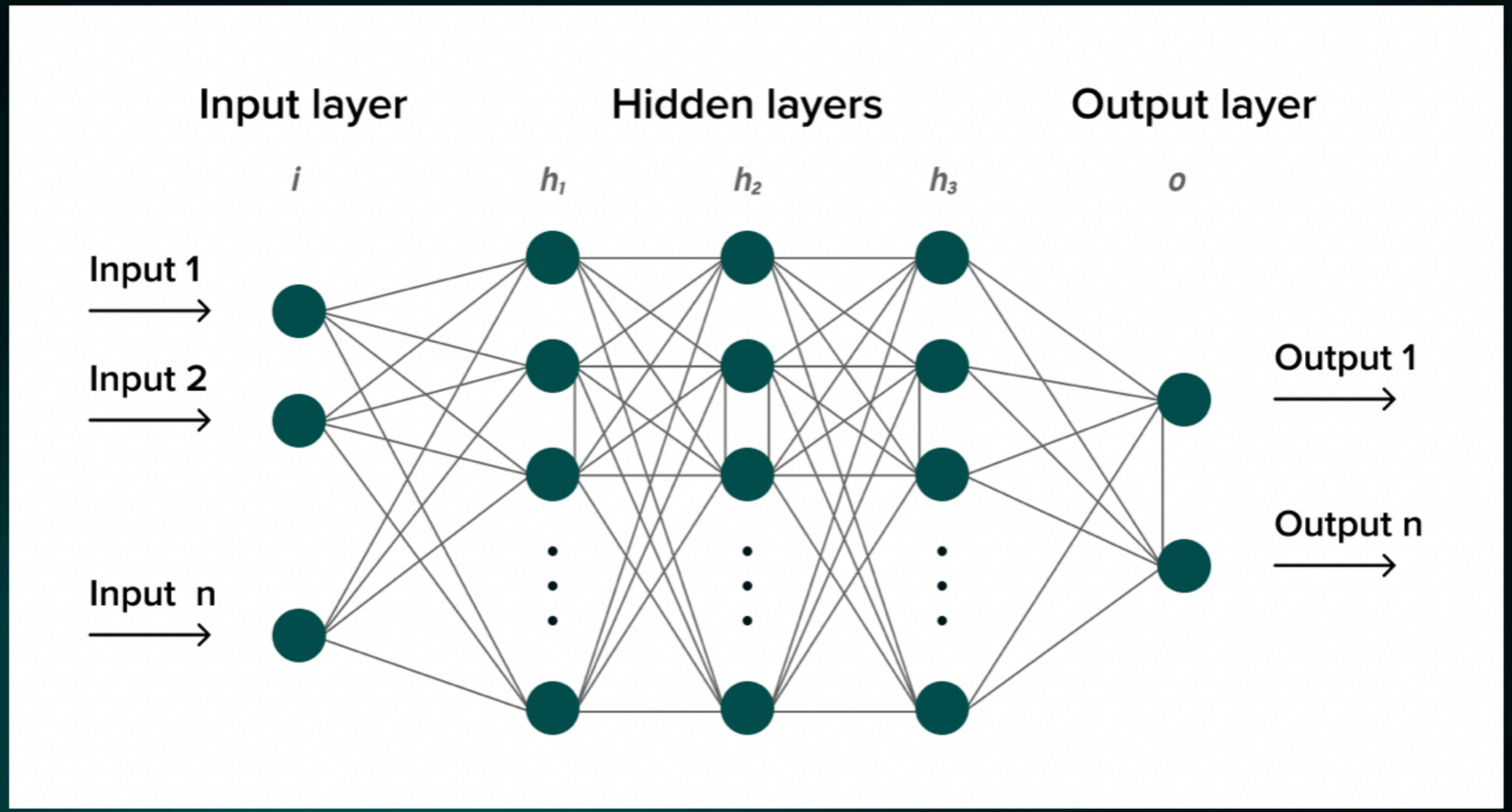
2020 state of enterprise machine learning

✓ see this link

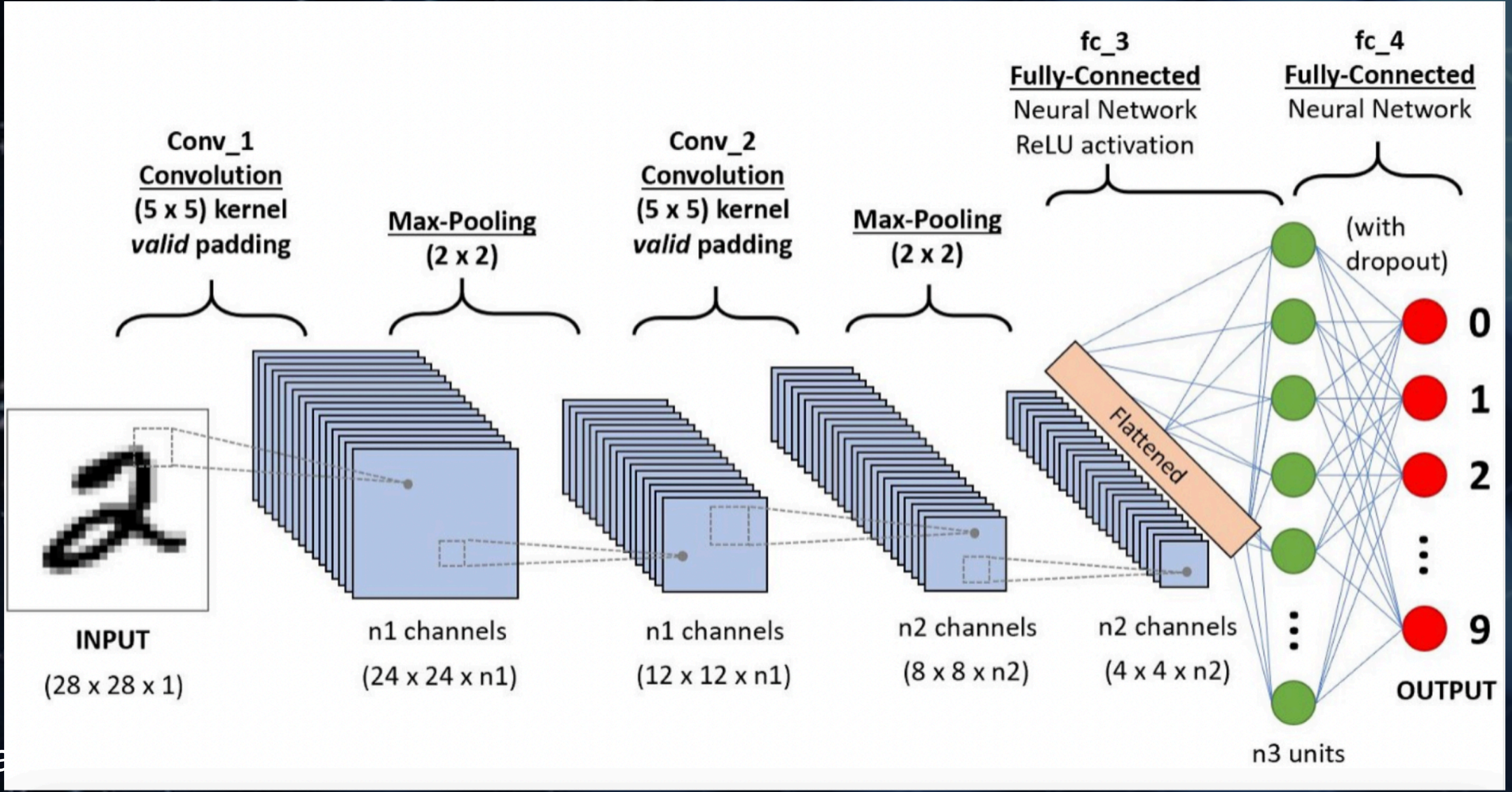
Survey at a glance

The main takeaway from the 2020 State of Enterprise Machine Learning survey is that a growing number of companies are entering the early stages of ML development, but challenges in deployment, scaling, versioning, and other sophistication efforts still hinder teams from extracting value from their ML investments. As a result, we will likely see a boom in the number of ML companies providing services to overcome these obstacles in the near term.

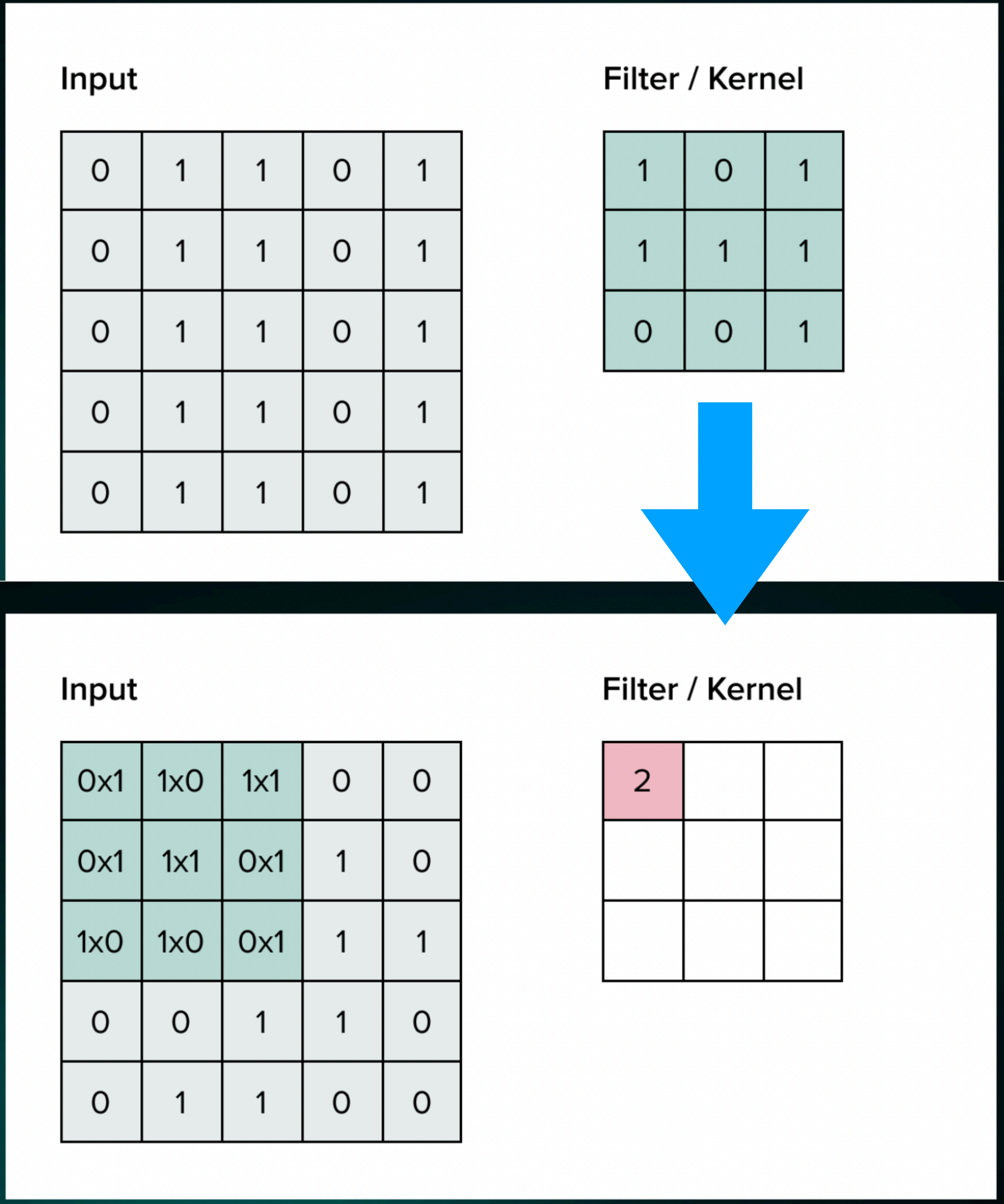
CNN



Input layer
Hidden layers
Output layer
i
h1
h2
h3
o
Input 1
Input 2
Input n
Output 1
Output n



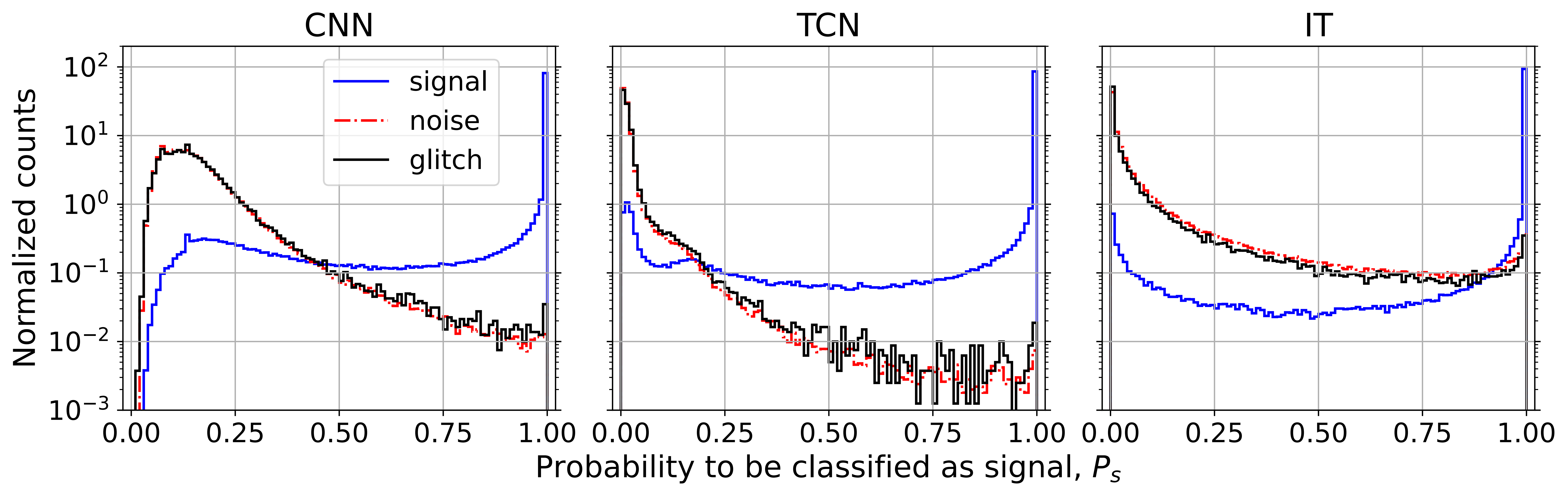
Conv_1
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
Conv_2
Convolution
(5 x 5) kernel
valid padding
Max-Pooling
(2 x 2)
fc_3
Fully-Connected
Neural Network
ReLU activation
fc_4
Fully-Connected
Neural Network
(with
dropout)
Flattened
INPUT
(28 x 28 x 1)
n1 channels
(24 x 24 x n1)
n1 channels
(12 x 12 x n1)
n2 channels
(8 x 8 x n2)
n2 channels
(4 x 4 x n2)
n3 units
0
1
2
9
OUTPUT



Input
Filter / Kernel
0 1 1 0 1
0 1 1 0 1
0 1 1 0 1
0 1 1 0 1
0 1 1 0 1
1 0 1
1 1 1
0 0 1
Input
Filter / Kernel
0x1 1x0 1x1 0 0
0x1 1x1 0x1 1 0
1x0 1x0 0x1 1 1
0 0 1 1 0
0 1 1 0 0
2

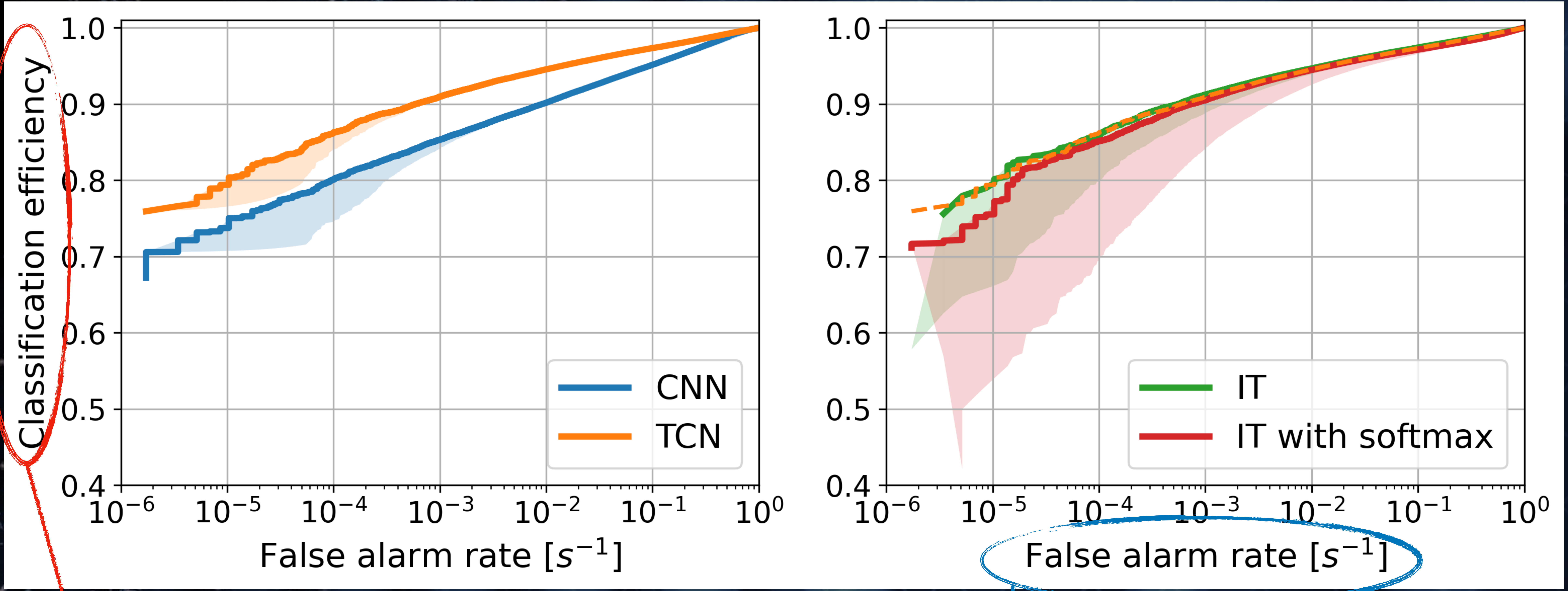
Probability to be classified as signal

Probability to be classified as signal can be used as test statistic



- Noise and glitch classes looks similar in all cases because in general the networks are not able to distinguish between glitch and noise (so they behave as only one class actually)
- We decided to focus on the signal identification and sum up noise + glitch

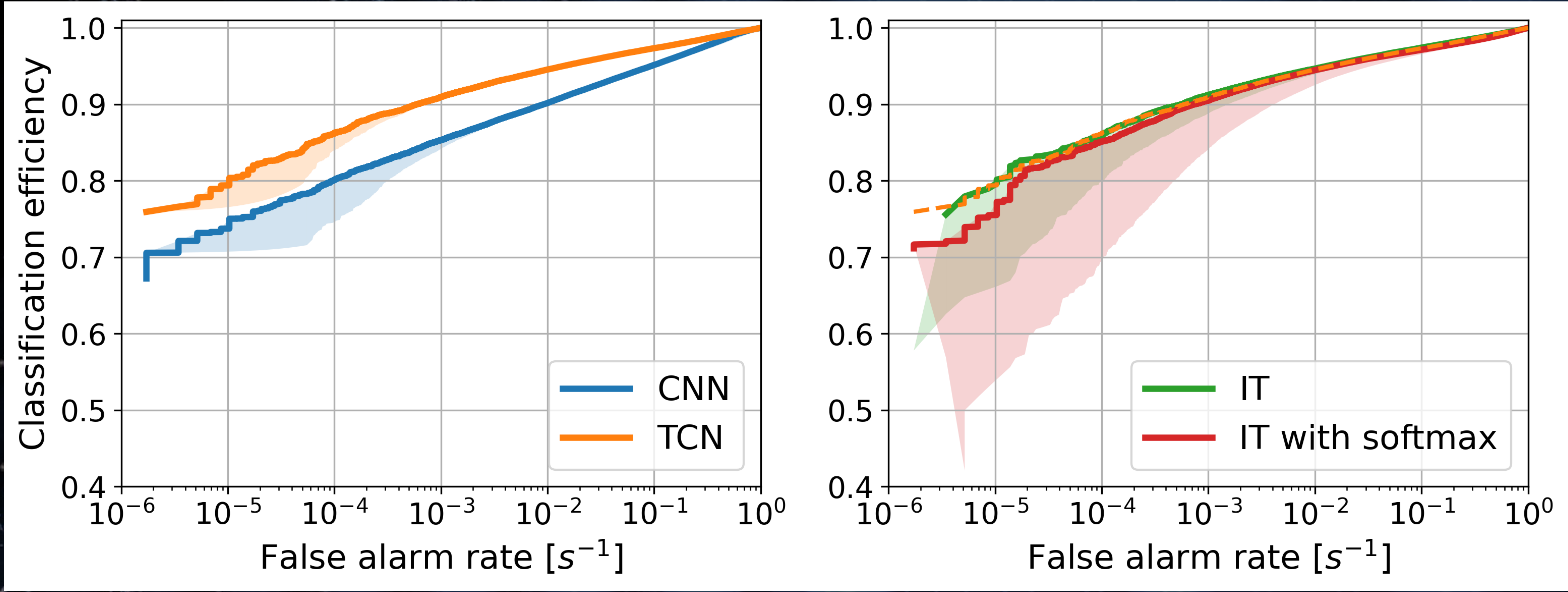
ROC curves



$$\frac{\#\ \textit{signal samples with } P_s \textit{ above some threshold}}{\textit{Tot signal samples}}$$

$$\frac{\#\ \textit{noise + glitch samples with } P_s \textit{ above some threshold}}{\textit{Tot duration}[s]\ \textit{noise + glitch samples}}$$

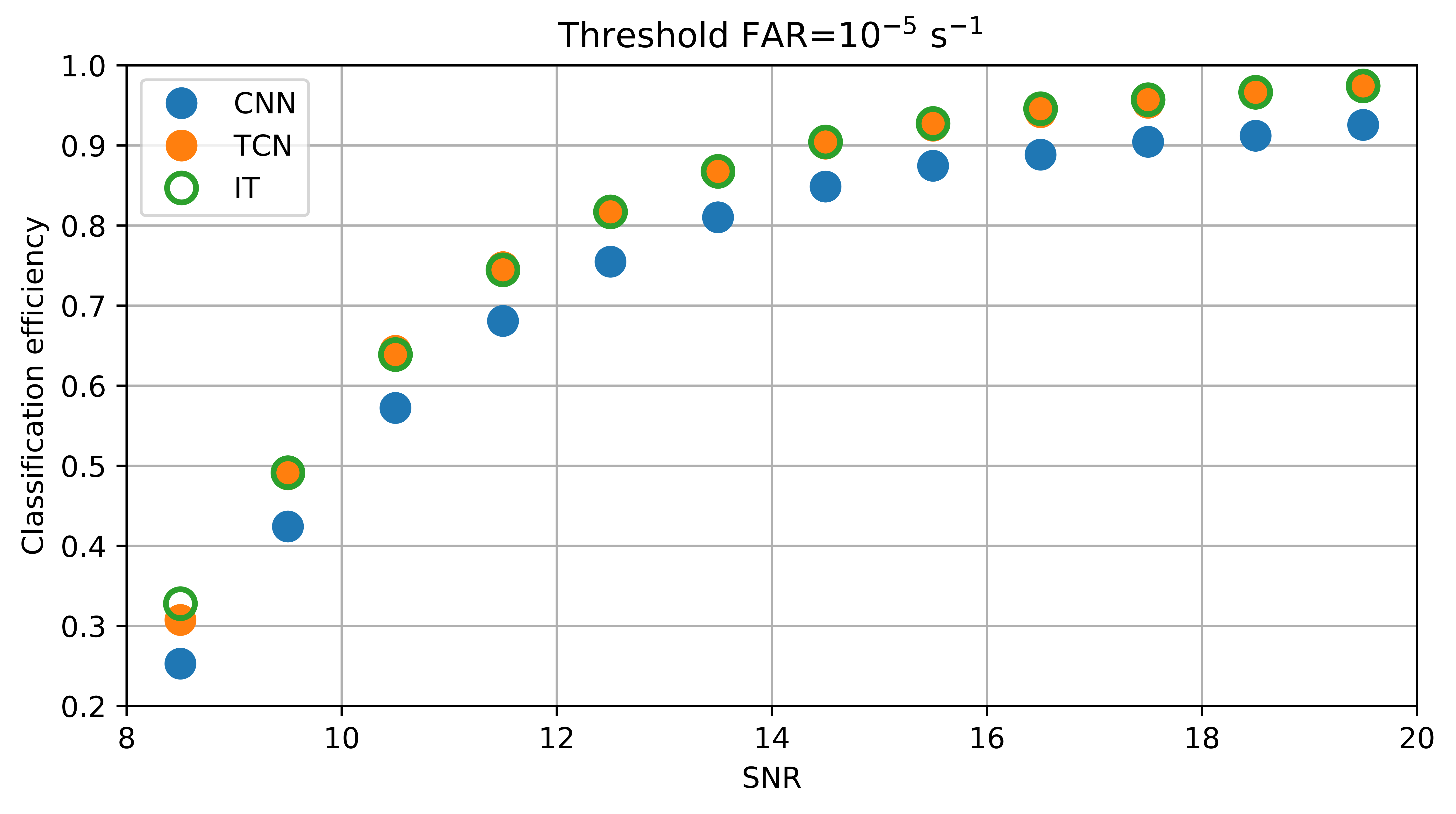
ROC curves



- Shaded area between the highest and the lowest ROC curves obtained for each model in the 10 repetitions of train and test
- “IT with softmax” refers to IT model with softmax activation function applied at the last fully connection layer during training.

Classification efficiency vs SNR for fixed FAR

Only the best model out of the 10 repetitions considered for each architecture



- TCN and IT perform similarly and outperform CNN
- Efficiency better than 0.5 for SNR>9 at this level of FAR
  - (1 alarm per 10^5 s = 0.864 alarms per day)

Trigger selection cut

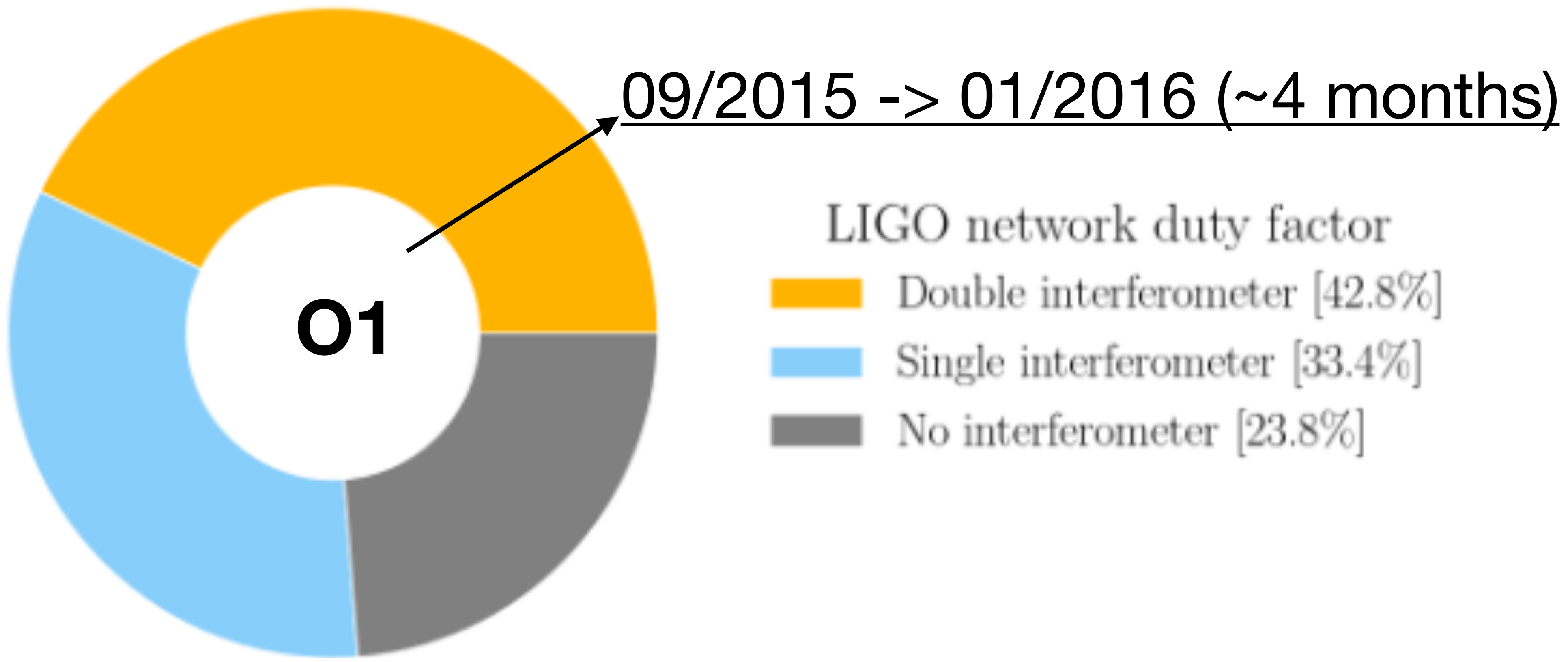
- We focus on the stricter cut that we can consider: **P_s=1** at machine precision (single-precision floating-point format)
- With this cut we have:

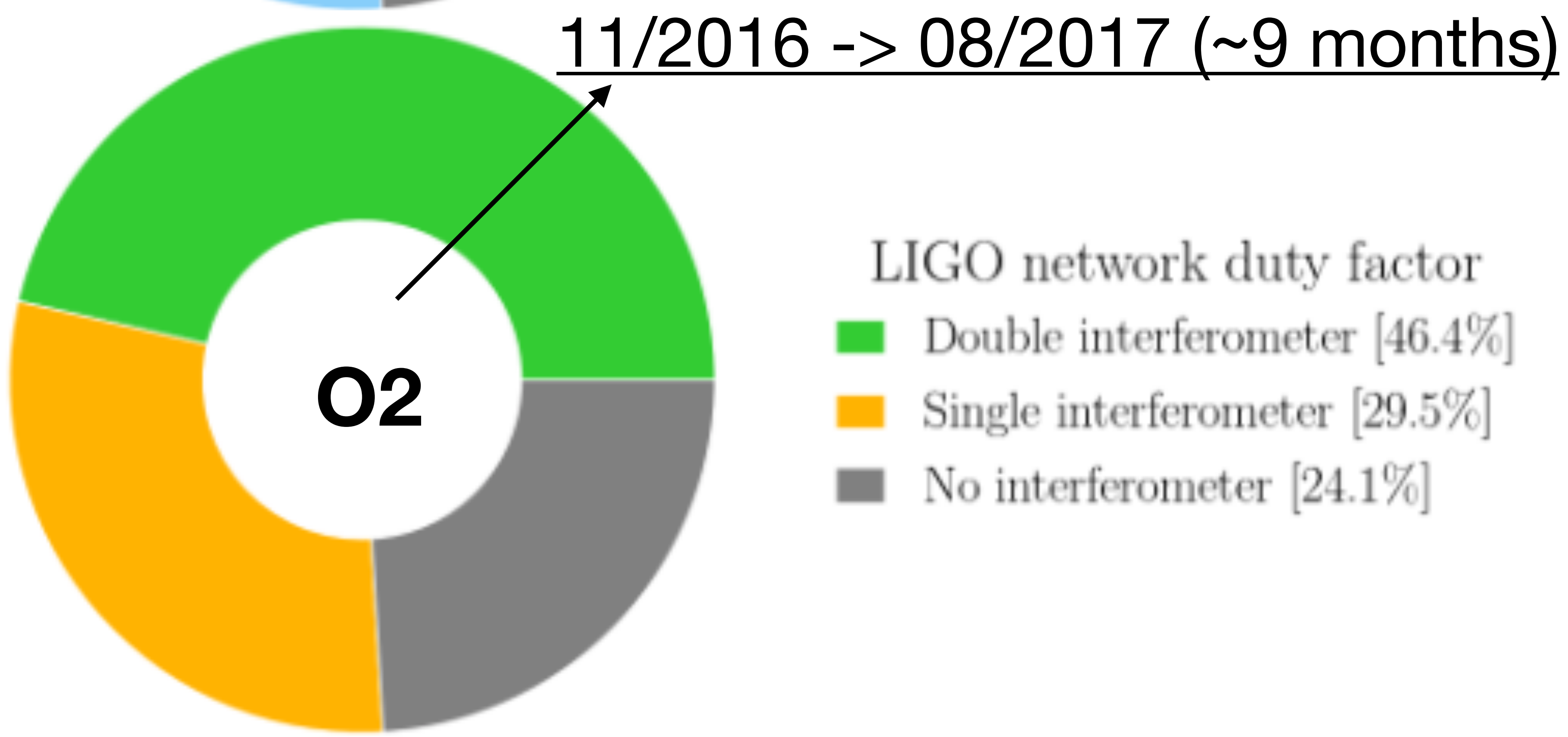
	CNN	TCN	IT
Noise+glitch samples with P_s=1	0	1	2
Equivalent FAR [s^{-1}]	$< 1.7 \times 10^{-6}$	1.7×10^{-6}	3.4×10^{-6}
Equivalent FAR in days	< 1/(7 days)	1/(7 days)	1/(3 days)
Signal classification efficiency	65%	76%	76%

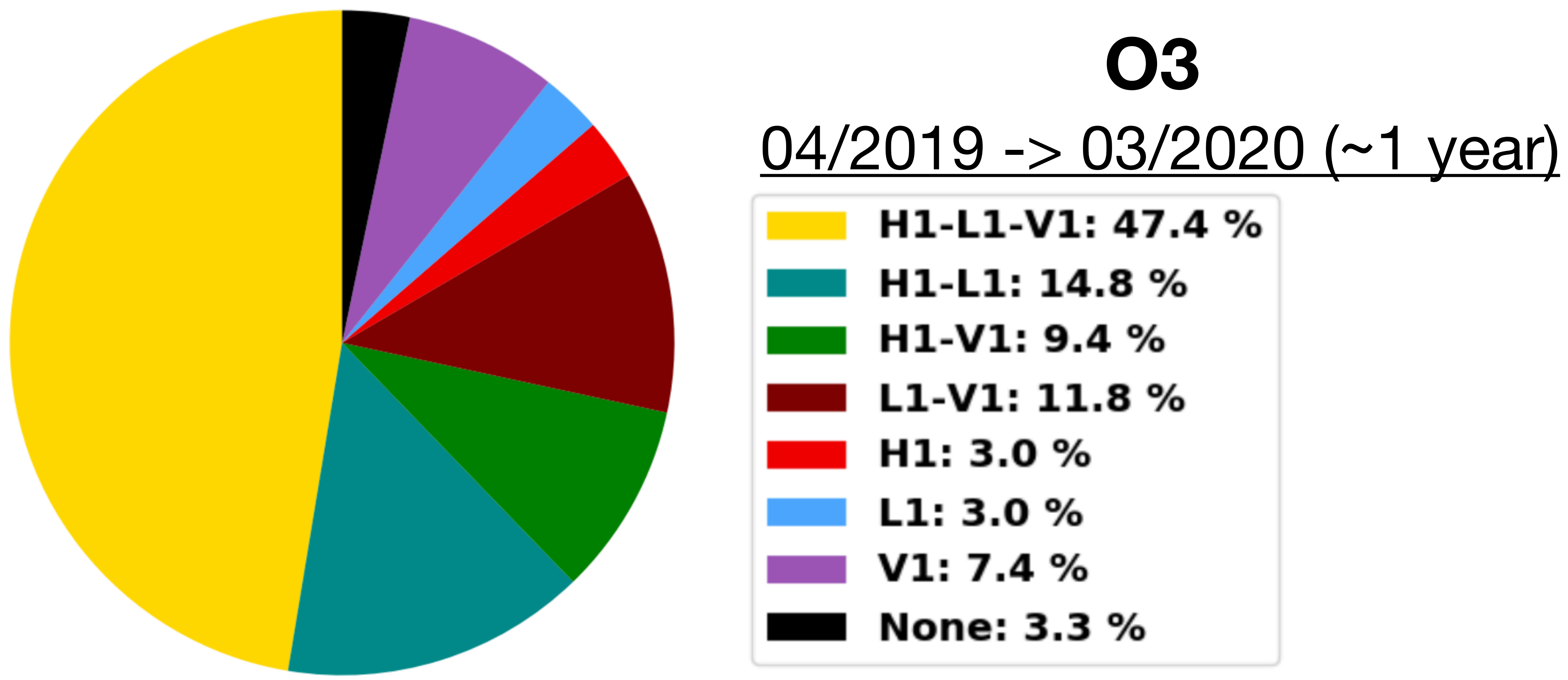
- The FAR level reached is compatible with our initial goal: 2 false alarms per day => FAR = $2.3 \times 10^{-5}\ s^{-1}$

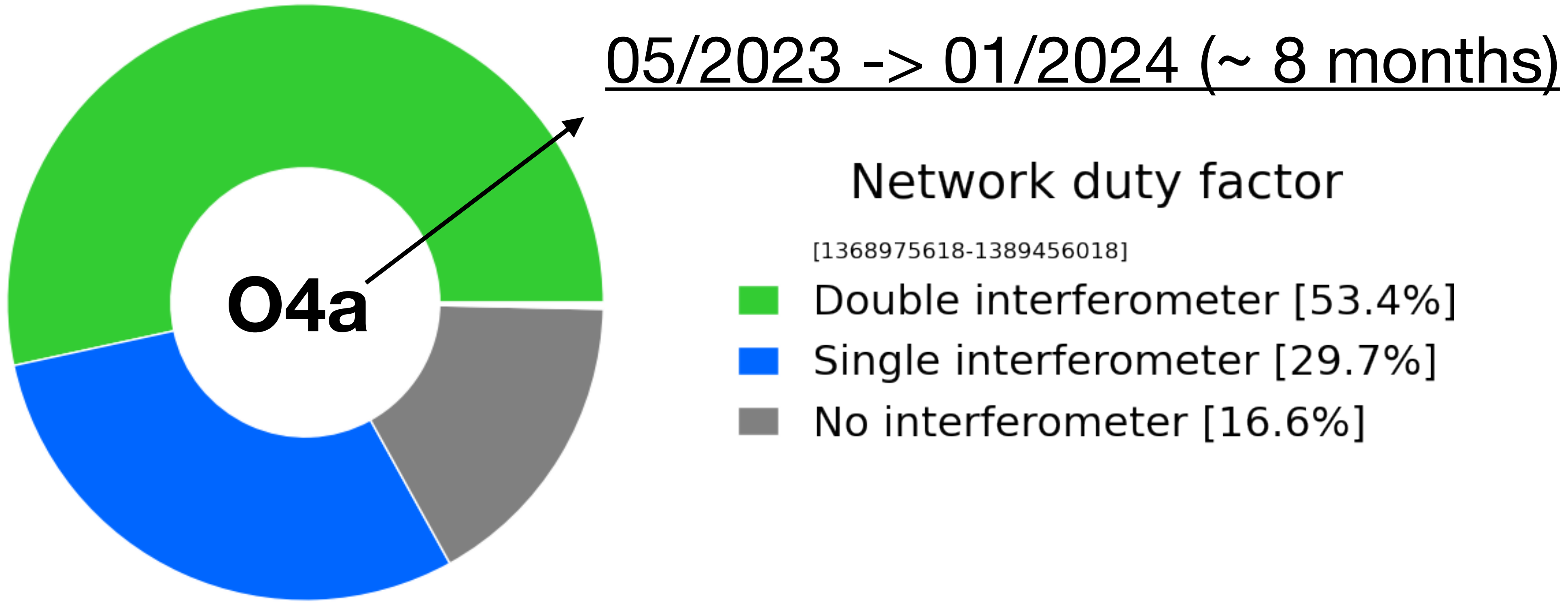
Single-detector time

- Glitch impact on sensitivity is larger during single-detector periods as coincidence with additional detector is impossible. Can machine learning help?
- Single-detector time:
  - ✓ ~2.7 months in O1+O2; ~1.6 months in O3: ~ 2.4 months in O4a



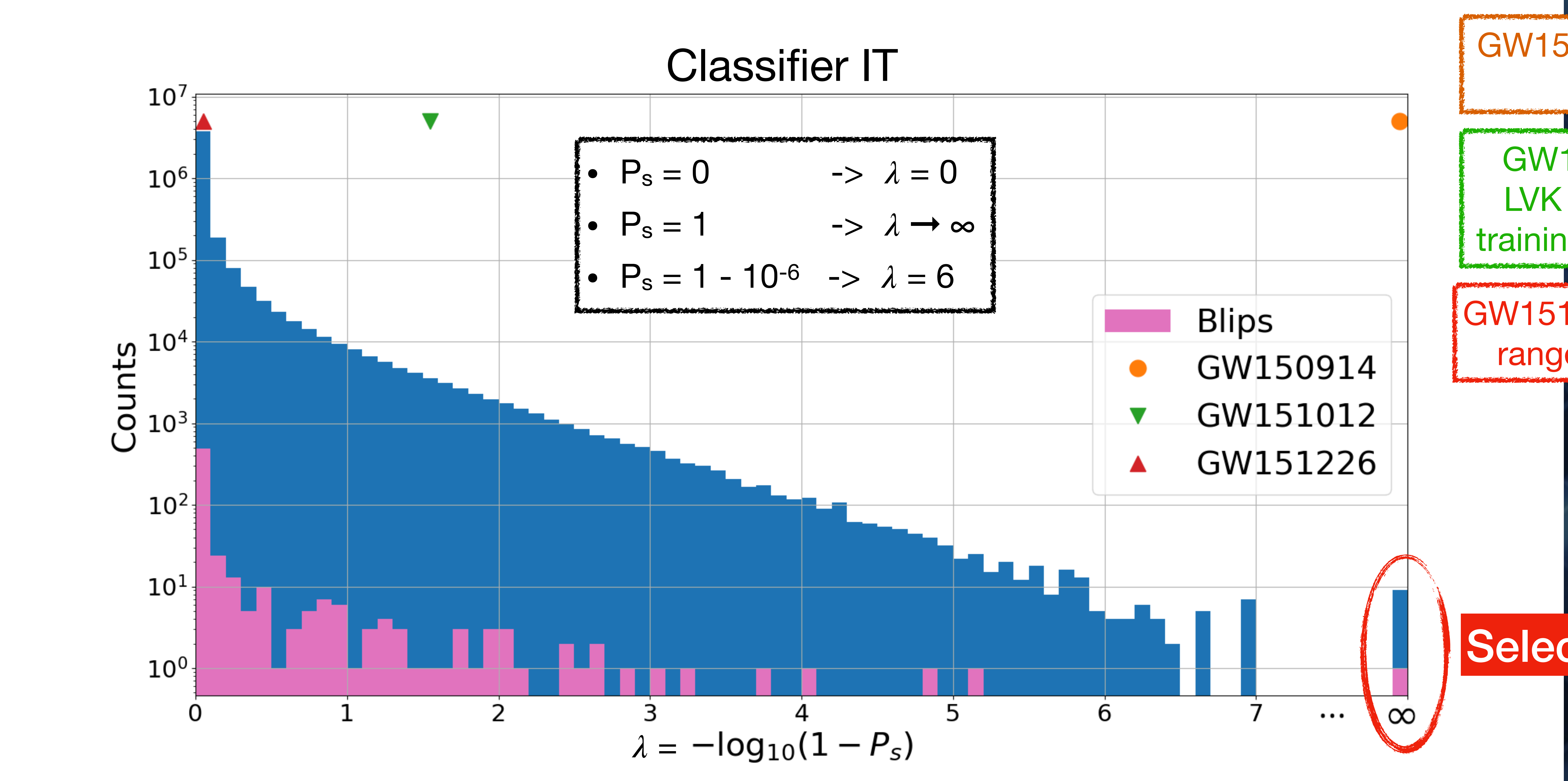






Analysis of the remaining 3 months of O1

- We applied the 3 networks to the remaining 3 months of L1 in O1 excluding the 1 month period already used for training and testing and know injections
- Periods around known GW detections have been examined separately



GW150914 identified with P_s =1 by all networks

GW151012 was detected by LVK in L1 with a SNR~6 (our training set has a minimum of 8)

GW151226 has masses not in the range used in our training set

Selected triggers

Triggers found in the remaining 3 months of O1

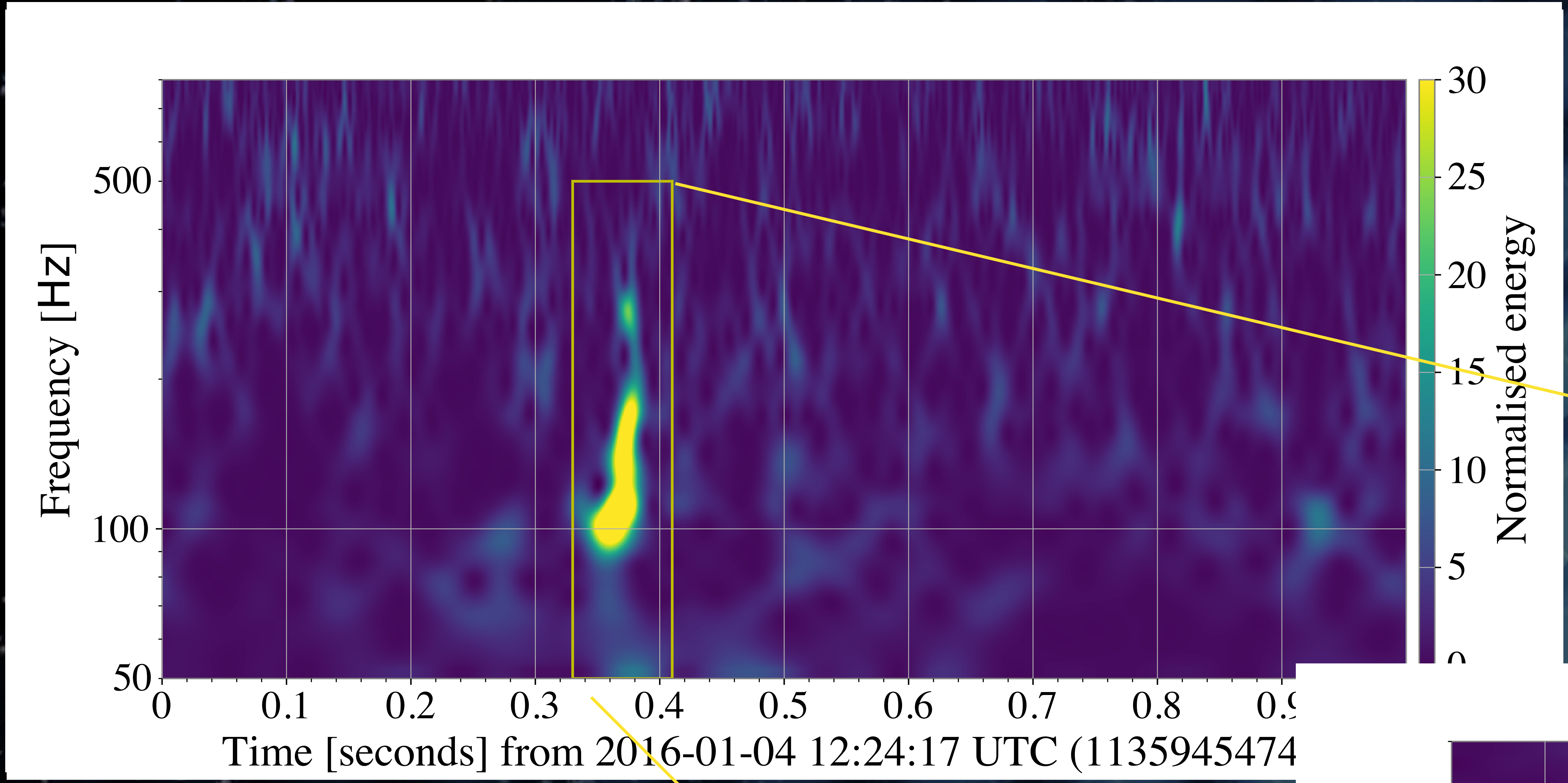
- Selection cut: P_S=1

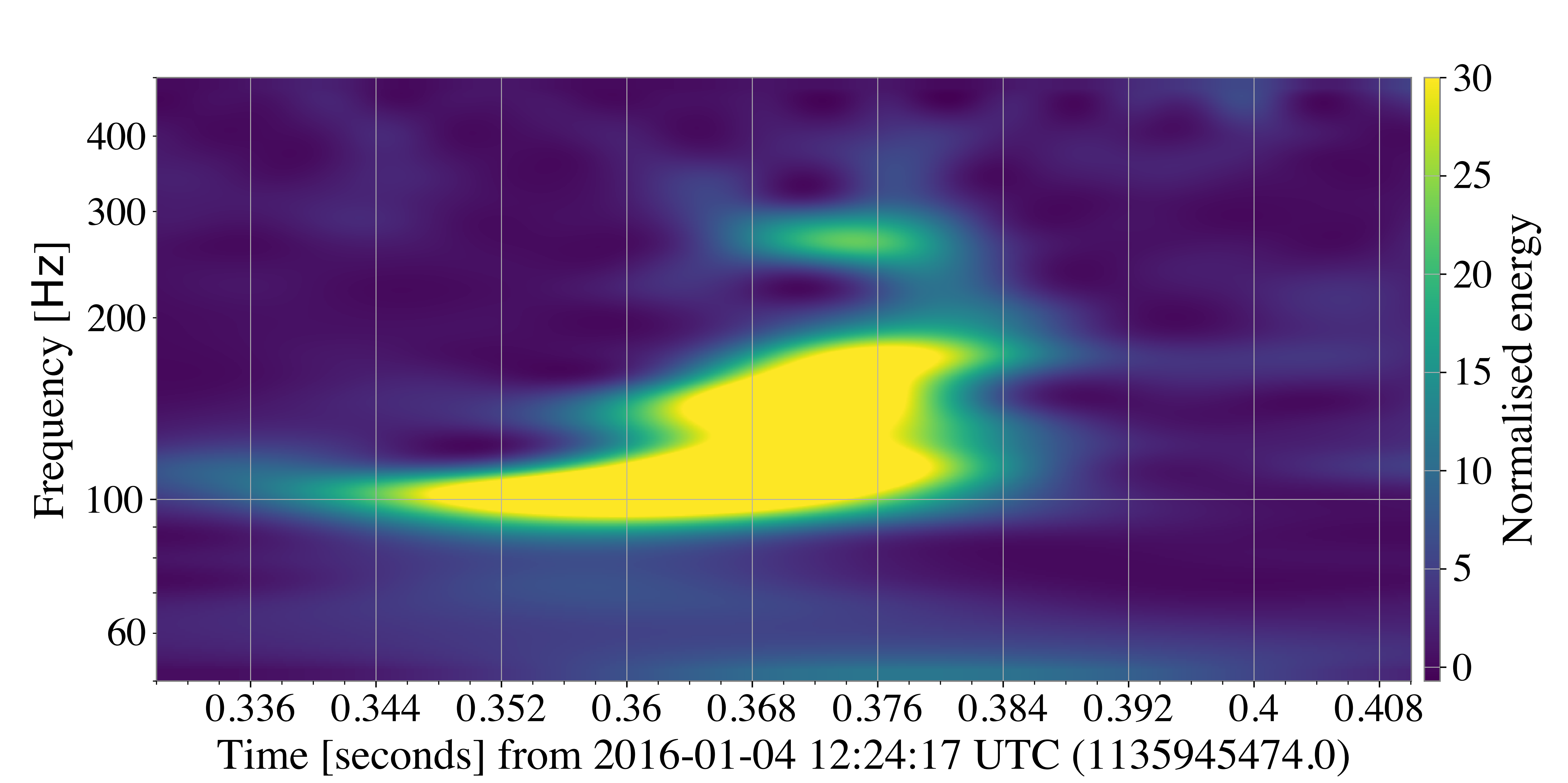
	CNN	TCN	IT
Samples with P_s=1 in single-det time	2	14	2
Samples with P_s=1 in double-det time	2	91*	7

- Only one event common to the three analyses: L1-only at **GPS=1135945474.0 (2016-01-04 12:24:17 UTC)**

* Trigger rate excess for TCN. At the limits of expected trigger count for single-detector times. Exceed expectation for multiple detector times (clusters of triggers observed during three periods of O1 -- under further investigations).

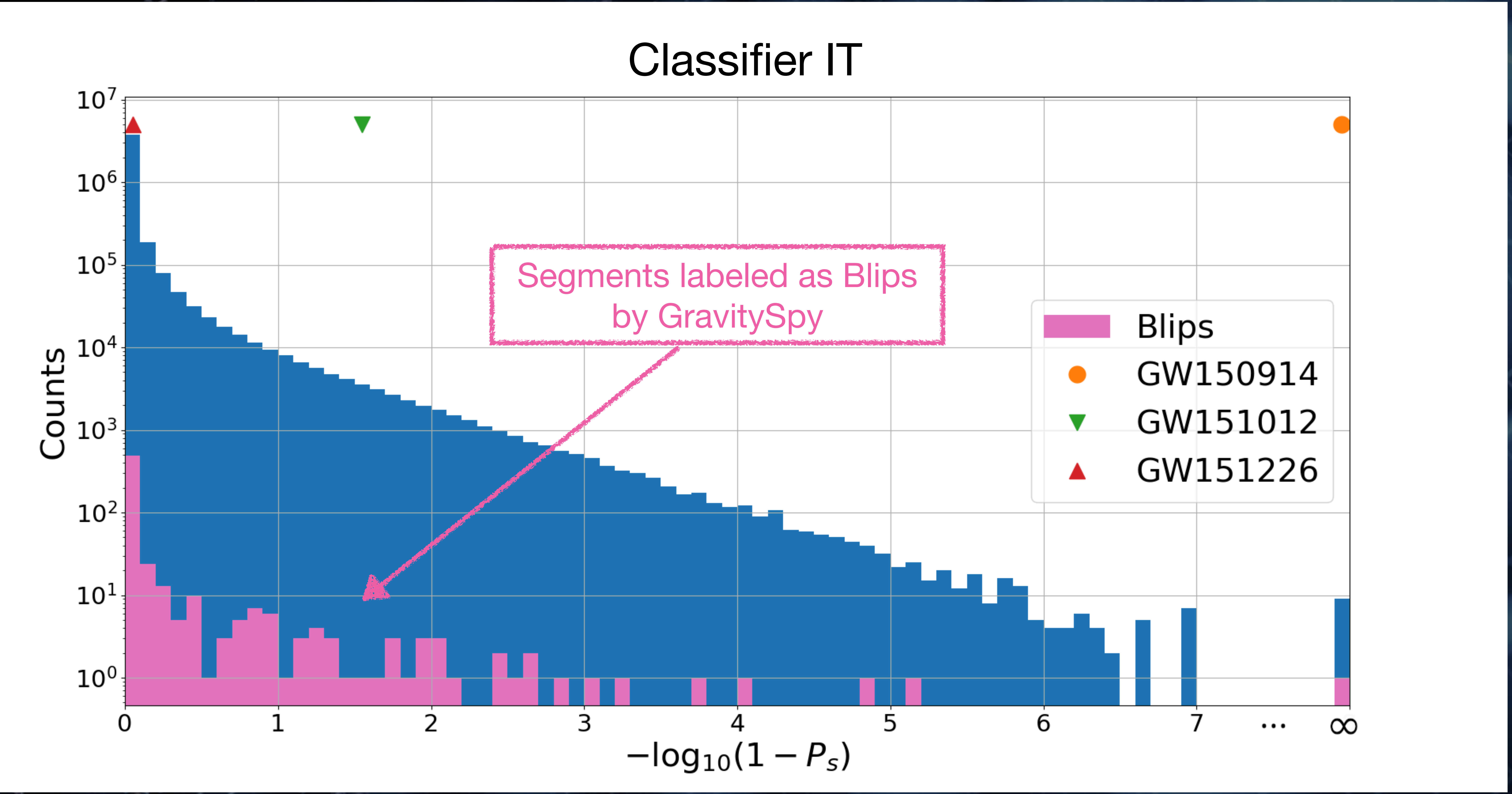
Q-scan segment 4th January 2016

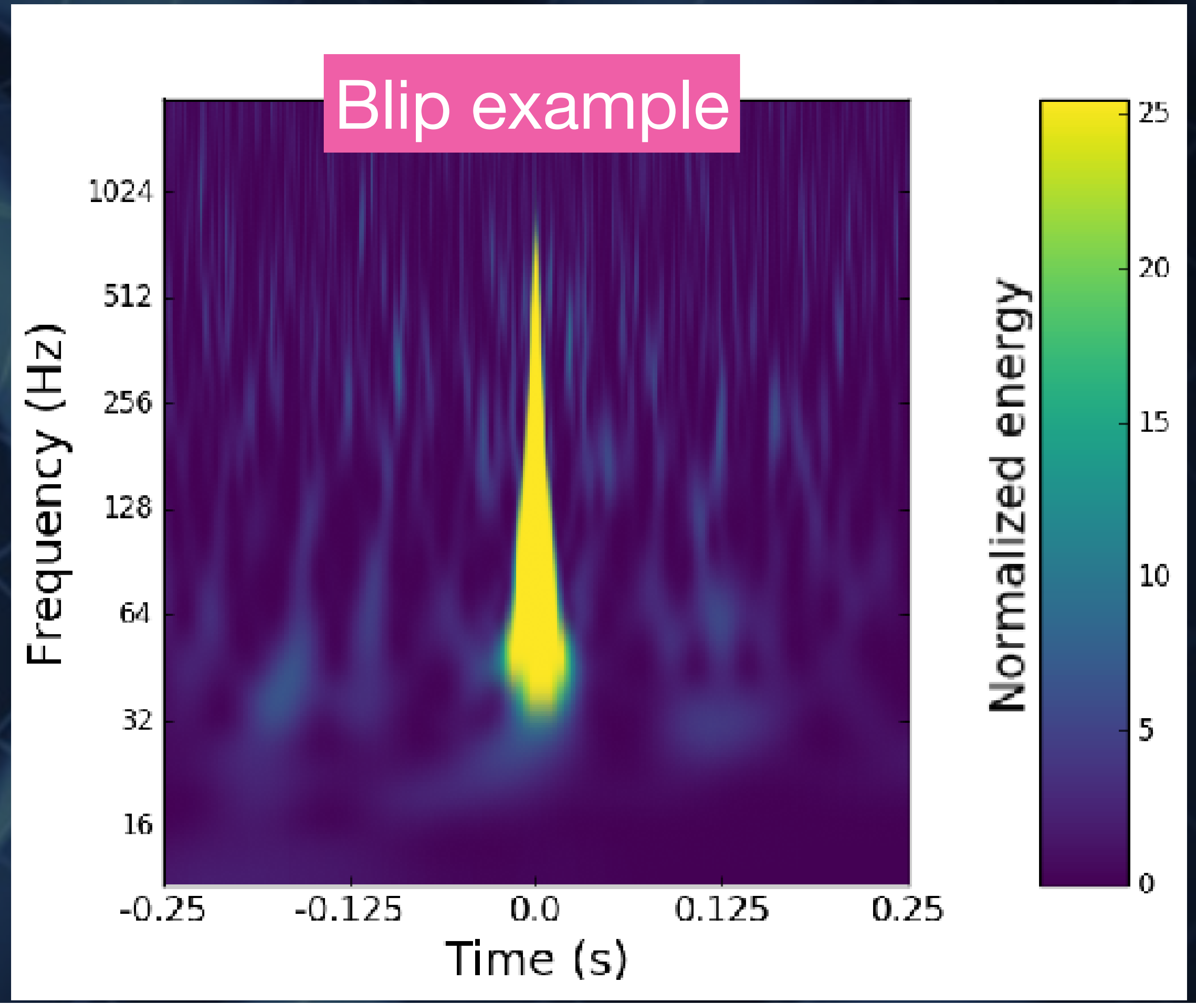


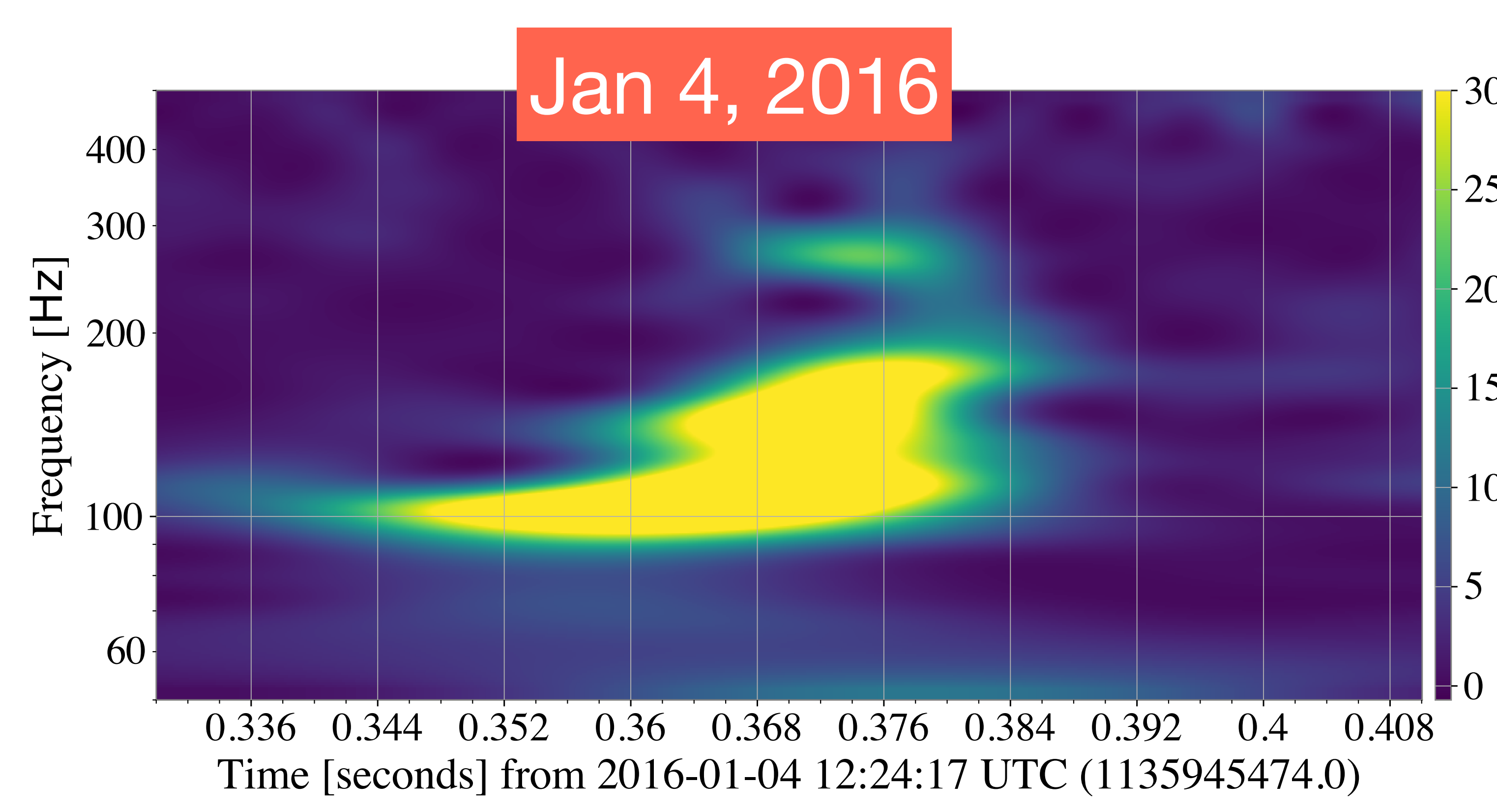


Is it a Blip?

- Gravity Spy finds a Blip at 1135945474.373
- In general the population of Blips compatible with background: Jan 4 outlier for this population

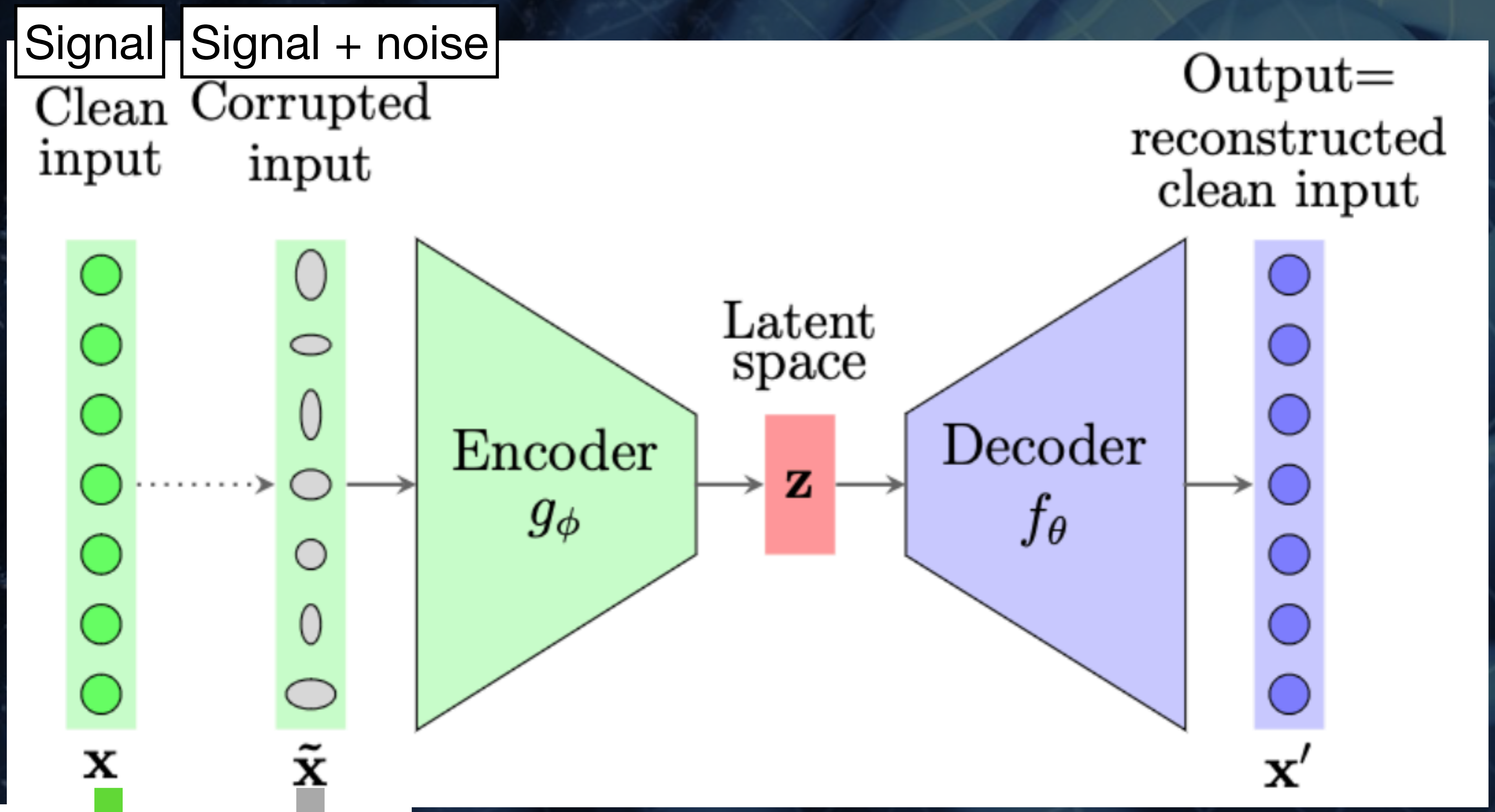






Has it an astrophysical origin?

- Checks that the transient signal is compatible with a GW waveform model
  - ✓ Bayesian parameter estimation: Bilby
  - ✓ Independent check: denoising convolutional neural network by Bacon et al 2023 Mach. Learn.: Sci. Technol. 4 035024

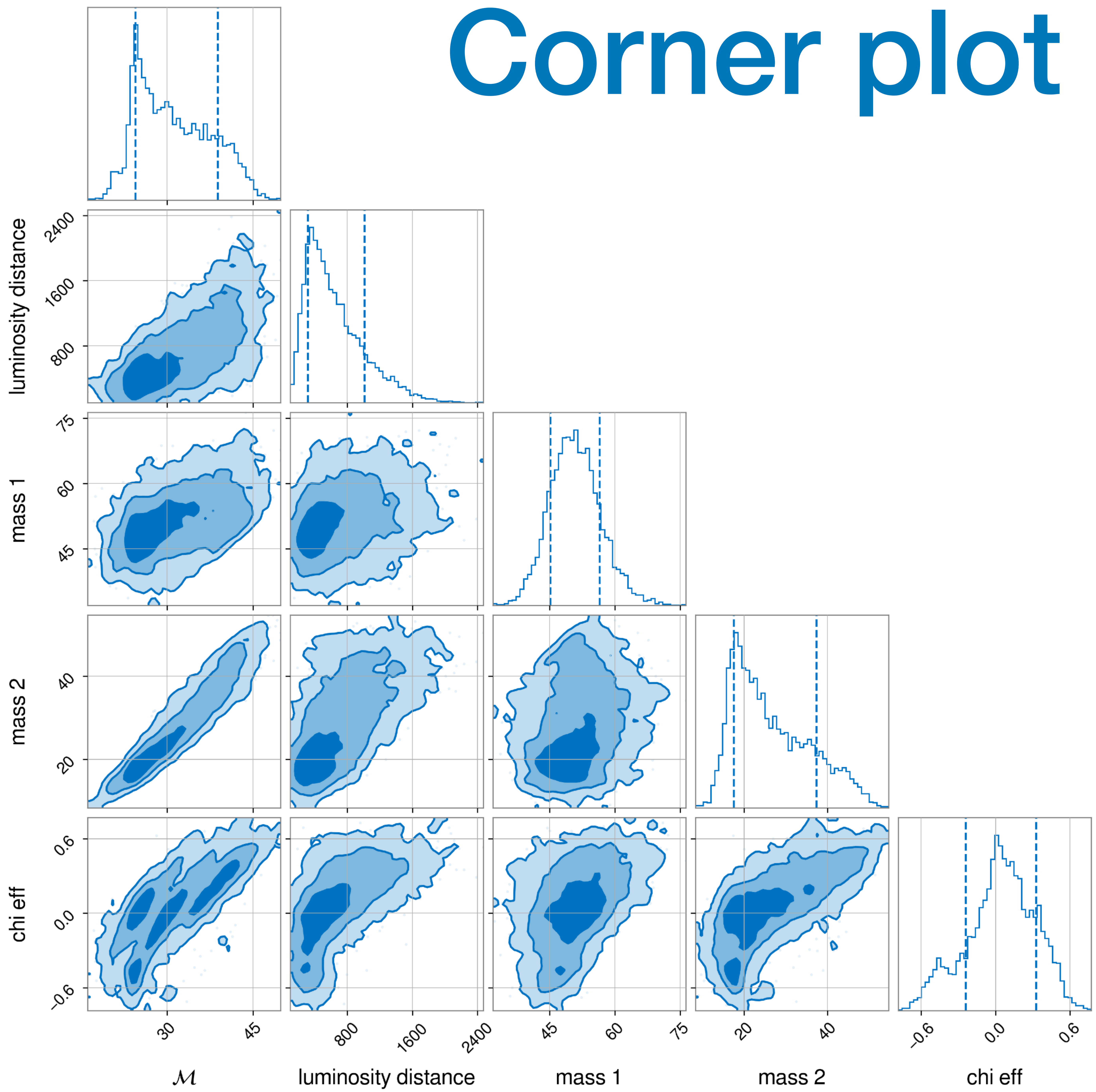


Denoising: model that takes noisy signals and returns clean signals

$$L_{DAE}(\theta, \phi) = \sum_{i=1}^{N} (x_i - f_\theta(g_\phi(\tilde{x}_i)))^2$$

Enconder and decoder are CNNs

Corner plot



$$GPS = 1135945474.373^{+0.076}_{-0.07}$$

$$SNR = 11.34^{+1.8}_{-1.6}$$

$$\mathscr{M} = 30.18^{+12.3}_{-7.3} M_\odot$$

$$m_1 = 50.7^{+10.4}_{-8.9}\ M_\odot$$

$$m_2 = 24.4^{+20.2}_{-9.3}\ M_\odot$$

$$\chi_{\rm eff} = 0.06^{+0.4}_{-0.5}$$

$$d_L = 564^{+812}_{-338}\ \text{Mpc}$$

Consistent with BBH population observed so far