Machine learning classification for gravitational-wave triggers in single-detector periods

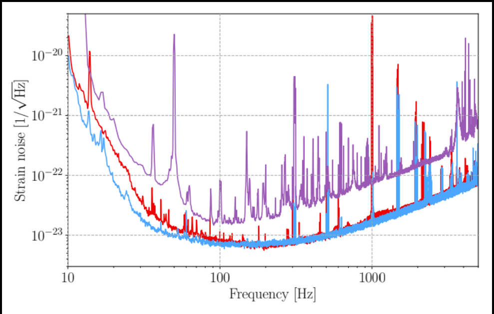
Michał Bejger, Éric Chassande-Mottin & Agata Trovato
(APC Paris)

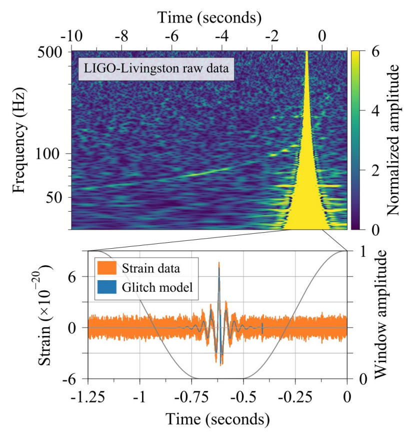
26.3.19

Outline

- ⋆ Introduction
- ⋆ Detectors' data and detection principles
- ⋆ Time-domain event classifier (DL CNN)
- ⋆ Ongoing work

LIGO-Virgo GW astronomy: 2015 - present

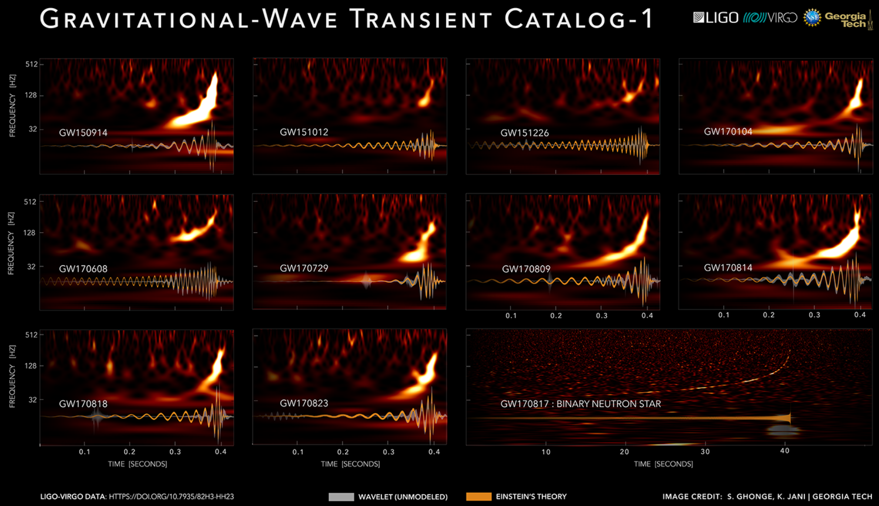




10 binary black hole mergers and 1 binary neutron star merger detected so far! (`arXiv:1811.12907`)

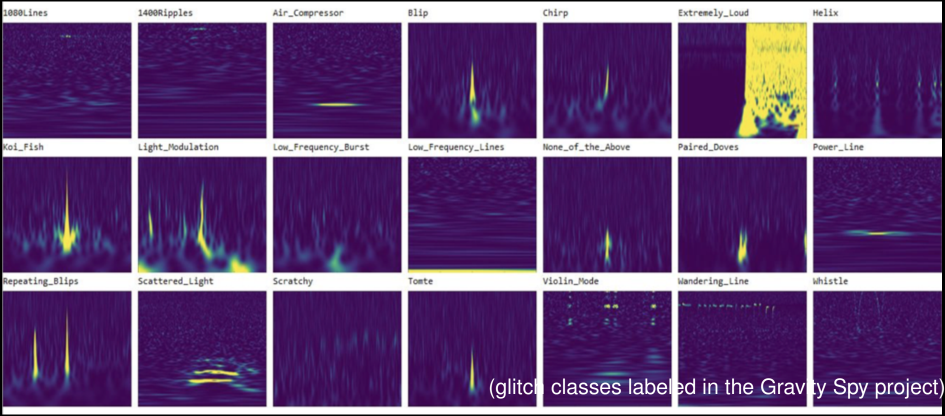
One year of O3: expected tens of binary black hole mergers and a few binary neutron star mergers.

GWTC-1



(arXiv:1811.12907)

Instrumental glitches as time-frequency maps



(glitch classes labeled in the Gravity Spy project)

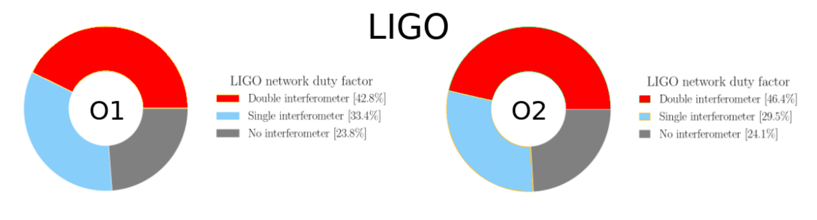
Deep learning on image data:

- ⋆ Spectrogram parameters/choice dependent,
- ⋆ Extra preparation time and large data volume.

Time series representation:

- ⋆ As close as possible to raw data (minimal manipulation),
- ⋆ Reduced volume of data.

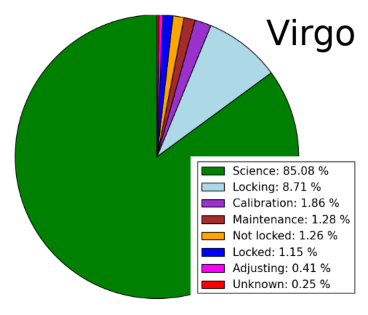
Network-of-detectors paradigm and beyond



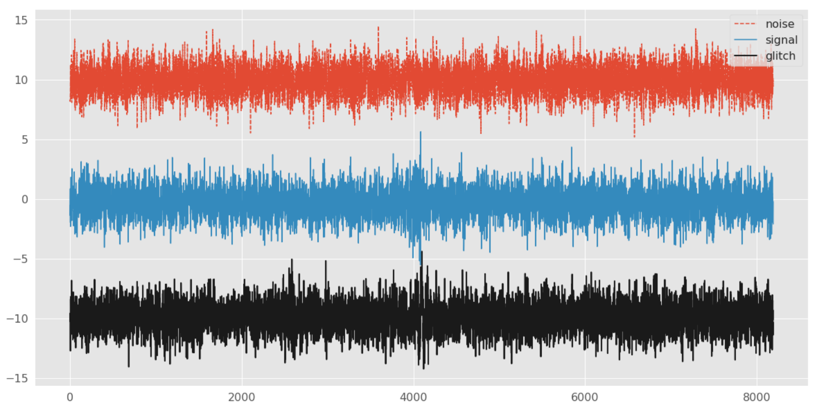
- ⋆ Transient noise may mimic the GW signal

→ current pipelines use coincidences in two or more detectors,

- ⋆ Single-detector time marginally exploited (2.7 months in O1+O2 → could contain 3 events)



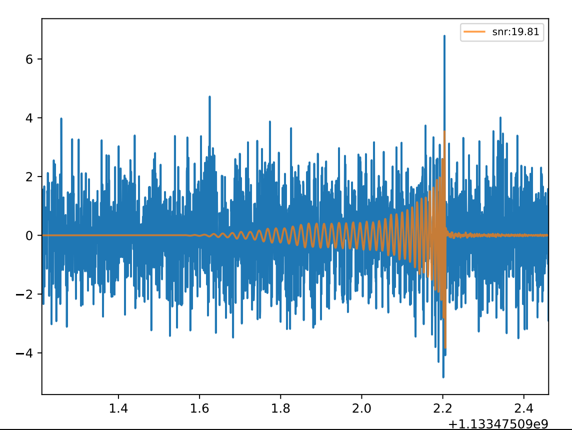
Training data: noise and glitches



Three instances of training data: noise, noise+signal, glitch

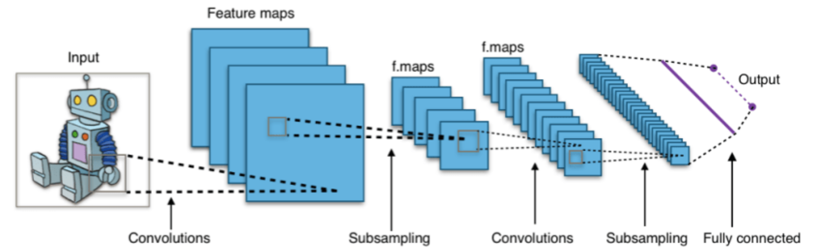
Glitches and ”clean” noise data samples from the last month of LIGO O1 run (downsampled to 2048 Hz, duration: 4s $\rightarrow$ 8192 points), whitened by the amplitude spectral density of the noise.

Training data: ”chirp” signals



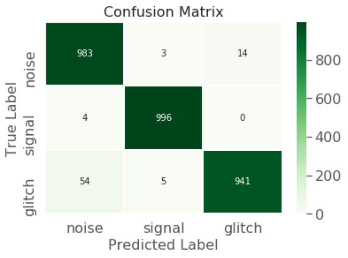
Randomly selected binary black holes’ system merger waveforms: $m_1, m_2 \in (8, 16)$, signal-to-noise $\rho \in (15, 45)$, added to ”clean” noise samples, whitened.

Convolutional Neural Network



Glitches, noises and signals - 1D classification results

- ⋆ Training data: 1000 instances $\times$ 3 classes,
- ⋆ Training time: $\simeq$10 minutes for 20 epochs @ Nvidia Tesla K40XL,
- ⋆ Accuracy on test data: 0.97



1D Convolutional Neural Network

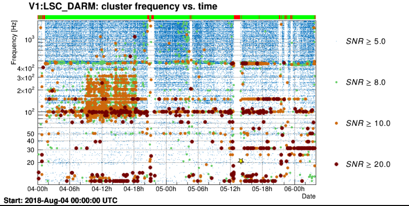
```
Layer (type)                 Output Shape              Param #
=================================================================
reshape_1 (Reshape)          (None, 8192, 1)           0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 8188, 500)         3000
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 2729, 500)         0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 2725, 250)         625250
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 2721, 250)         312750
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 907, 250)          0
_________________________________________________________________
conv1d_4 (Conv1D)            (None, 903, 150)          187650
_________________________________________________________________
global_average_pooling1d_1 ( (None, 150)               0
_________________________________________________________________
dropout_1 (Dropout)          (None, 150)               0
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 453
=================================================================
Total params: 1,129,103
Trainable params: 1,129,103
Non-trainable params: 0
```

Ongoing work

Having the training data set, we are

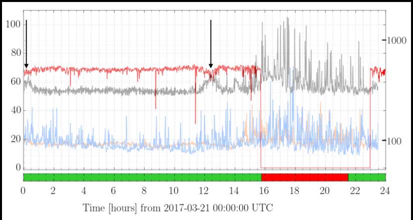
- adding features:
  - environmental channels (multi-instance learning),
  - specific classification for glitches (e.g. using labeled data from Gravity Spy),
  - study causality between time-series (e.g. main GW channel vs environmental channels),
  - DNN compression to decrease the size and latency.
- implementing other ideas for 1D DNN:
  - Recurrent Neural Networks (RNN), Long-Short Term Memory (LSTM) for classification,
  - Convolutional Denoising Autoencoders (to denoise signals/glitches to study their morphology),
  - Generative Adversarial Networks for anomaly detection.

Causality studies



Example Virgo time-frequency glitchgram

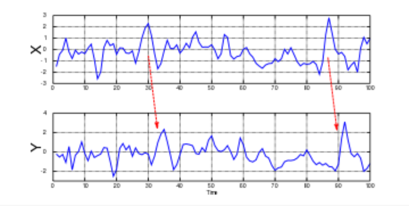
(Virgo logbook)



LIGO Hanford horizon drops due to trucks

(Berger 2018)

Predictive (Granger) causality: adding a new time-series improves prediction of the next data-point.



In the context of GW data characterization and quality:

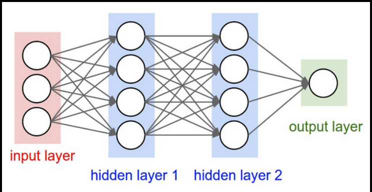
- ⋆ Detect cause-effect relation between auxiliary and main GW channels,
- → Investigate and remove noise sources and/or characterize and remove glitches from the main GW channel.

DNN compression

DNN as a parametric model:

$$p(\mathcal{D}|\mathbf{w}) = \Pi_{i=1}^{N} p(x_i|y_i, \mathbf{w})$$

$\mathcal{D}$ - data, N x_i - input y_i - output pairs, $\mathbf{w}$ - parameters weights with prior $p(\mathbf{w})$



- ⋆ Minimum description length (MDL) principle ("best model is the most compressed"), related to Bayesian inference
- ⋆ Approximating the posterior $p(\mathbf{w}|\mathcal{D} = p(\mathcal{D}|\mathbf{w})p(\mathbf{w})/p(\mathcal{D})$ to minimize a function

$$\mathcal{L}(\phi) = \underbrace{\mathcal{L}^C}_{\text{complexity cost}} + \underbrace{\mathcal{L}^E}_{\text{error cost}}$$

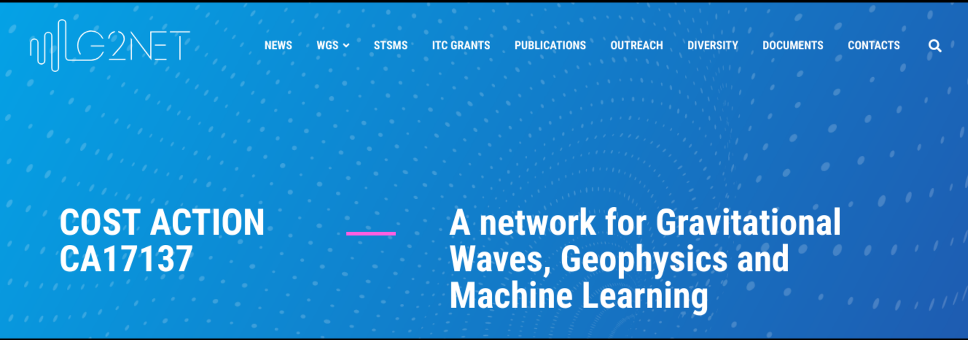
by optimizing the variational parameters ϕ with sparsity-inducing priors to prune nodes/weights.

→ "variational dropout"

→ motivation e.g. Louizos et al. 2017 (`arXiv:1705.08665`).

+

COST action G2Net (g2net.eu)

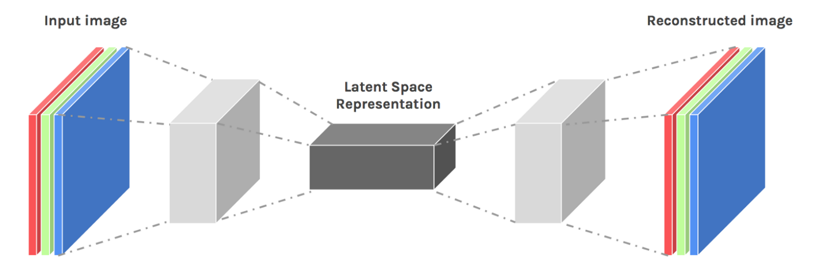


Three working groups:

- ⋆ WG1: ML for GW astronomy,
- ⋆ WG2: ML for low-frequency seismic measurement,
- ⋆ WG3: ML for Advanced Control techniques.

You are cordially invited to join! ⌣̈

Convolutional Denoising AutoEncoder



noise+signal

signal

Generative Adversarial Network