Wavefier
Real time Gravitational Wave transient signal classifier

WaveFier: a prototype for a real time transient signal classifier

IPAM GW2019: Computational Challenges in Gravitational Wave Astronomy (January 28 - February 1, 2019)

Elena Cuoco, EGO and SNS
www.elenacuoco.com
Twitter: @elenacuoco

EGO EUROPEAN GRAVITATIONAL OBSERVATORY
SCUOLA NORMALE SUPERIORE

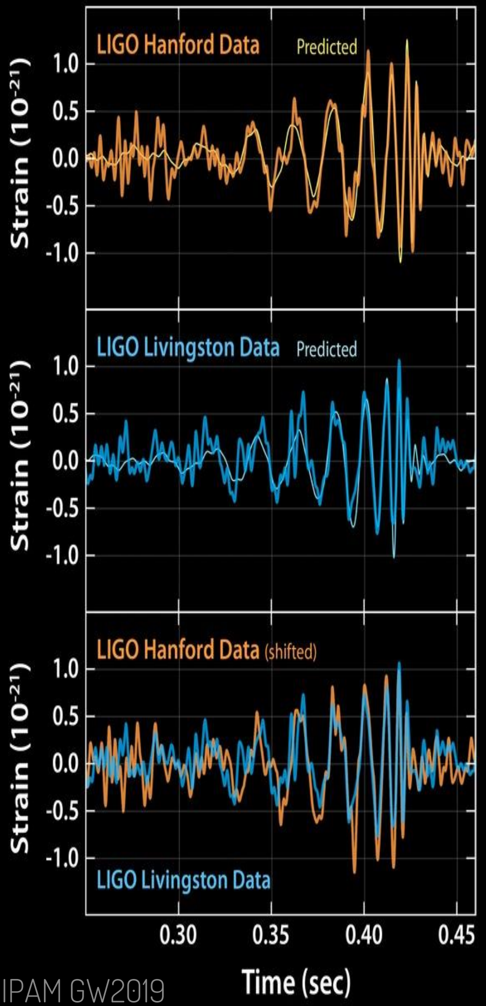
VIR-0922A-18

Outline

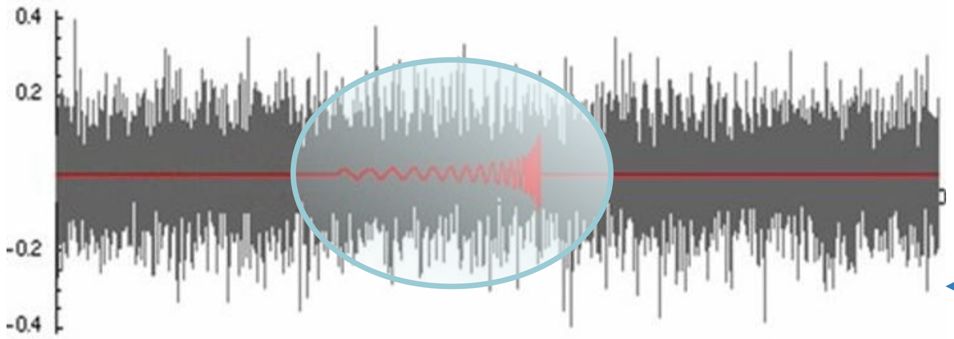
Wavefier

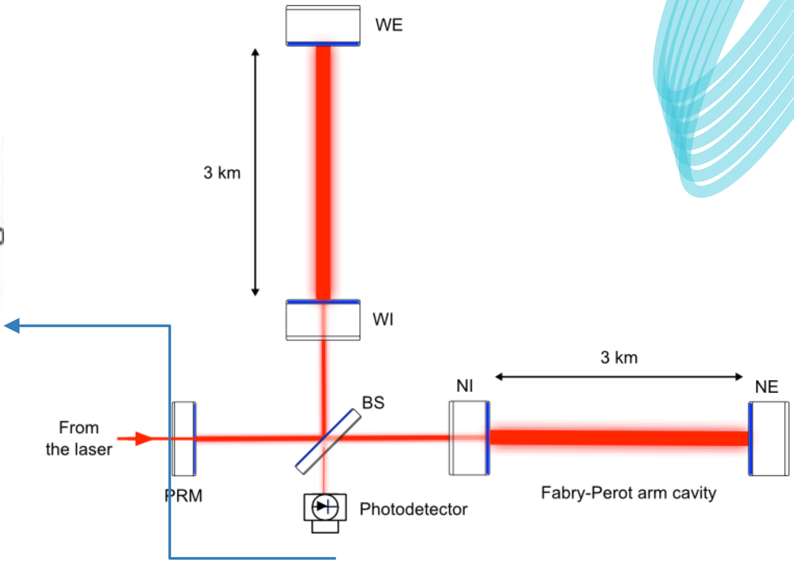
A prototype for a real time transient signal classifier

- Why we need to classify transient signals
- GW data characterization
- Whitening procedure
- Adaptive whitening
- Wavelet decomposition
- Wavelet de-noising
- Waveform reconstruction
- Triggers Detection
- Real time implementation
- Triggers classification
- Wavefier



Why Machine Learning in Gravitational Wave research





LIGO/Virgo data

are time series sequences… **noisy time series** with low amplitude GW signal buried in

Our "signals"

Astrophysical signals

Known GW signals

Compact coalescing binaries has known theoretical waveforms

Optimal filter: Matched filter

Too many templates to test

Unknown GW signals

Core collapse supernovae

No Optimal filter

Parameters estimation

Noise

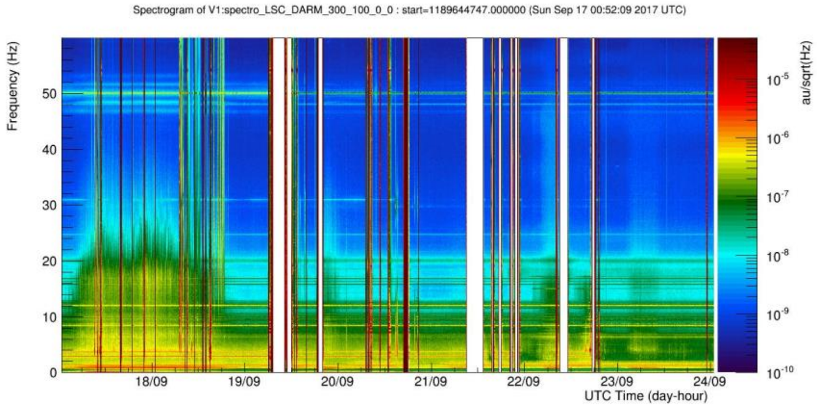
Moving lines

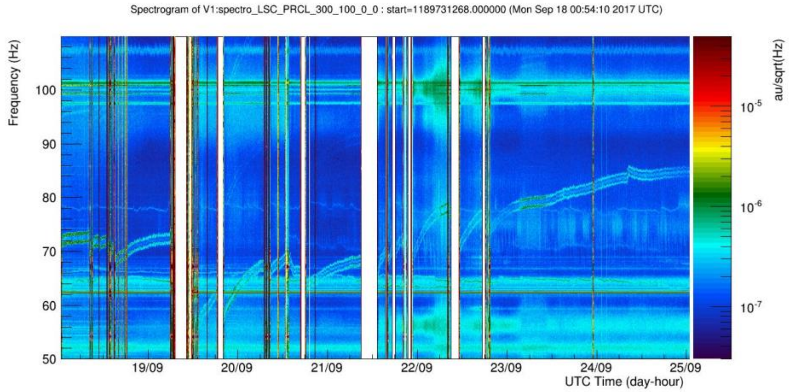
Broad band noise

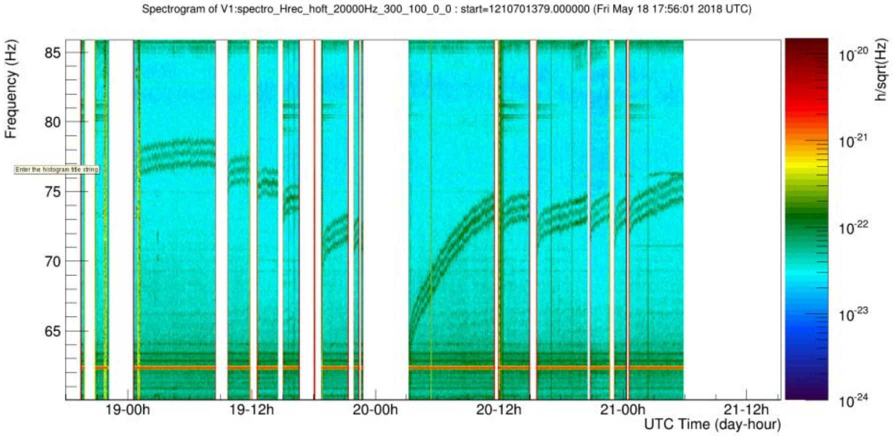
Glitch noise

"Pattern recognition" by visual inspection

Example of other noise signals

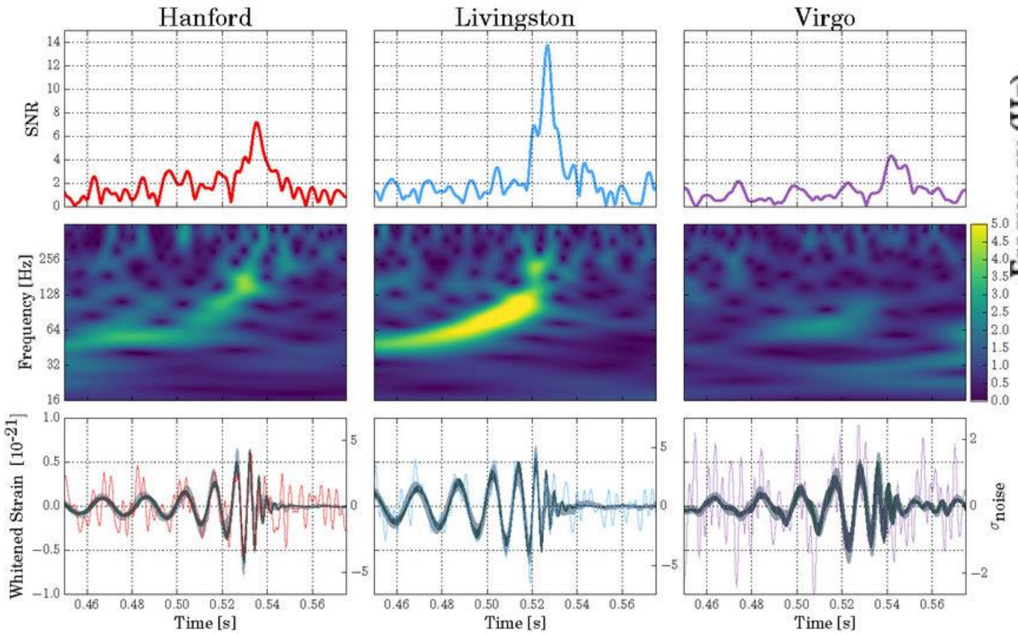




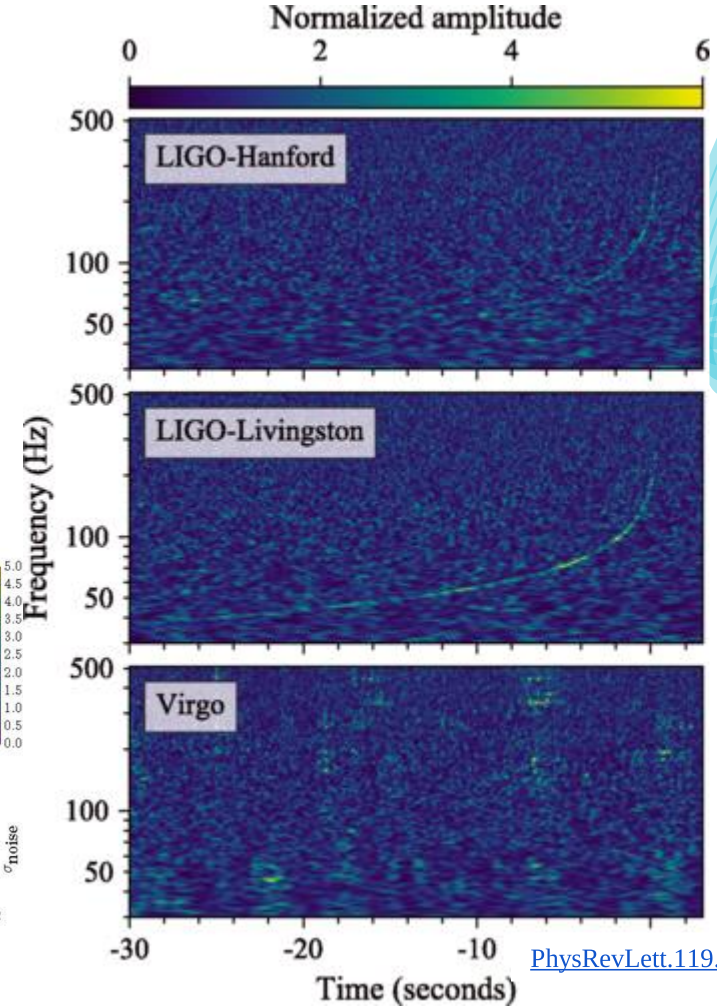


I. Fiori courtesy

Example of GW signals in Time-Frequency plots



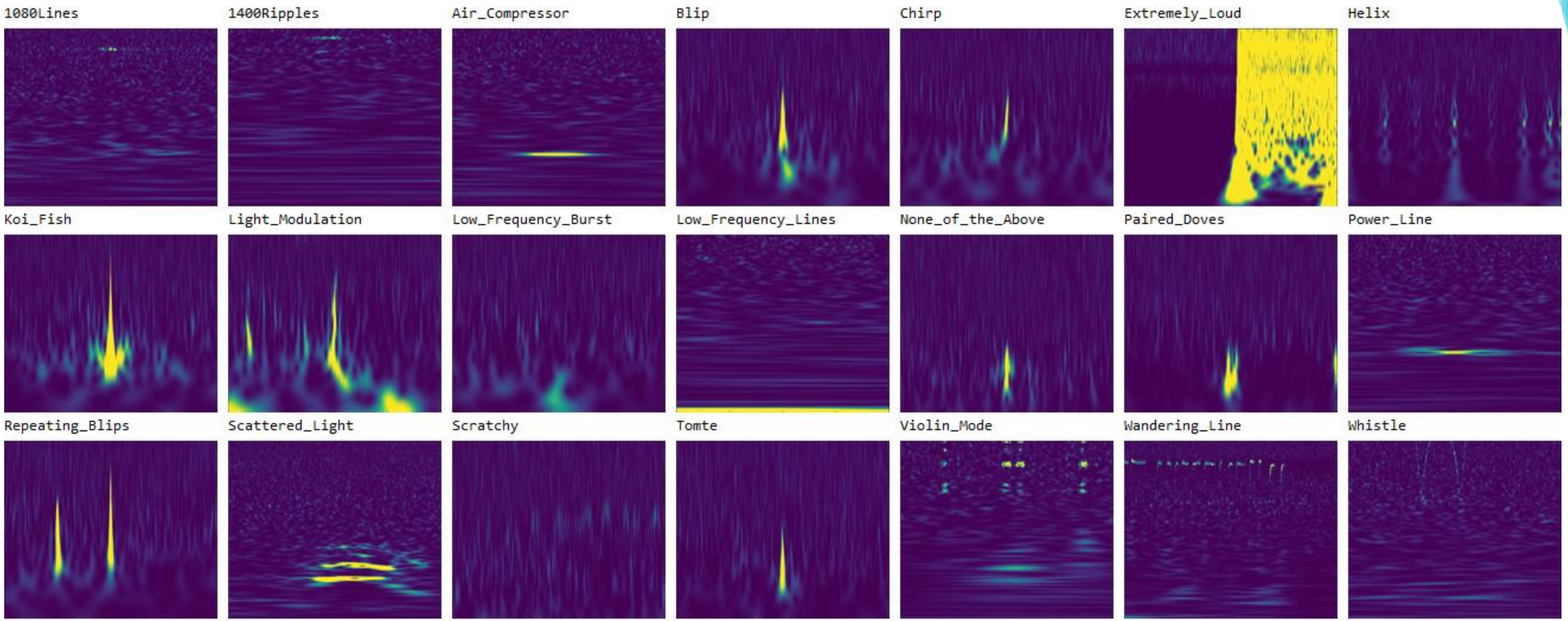
Phys. Rev. Lett., 119 (14), pp. 141101, 2017.



PhysRevLett.119.161101.

Example of Glitch signals

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



Gravity Spy, Zevin et al (2017)

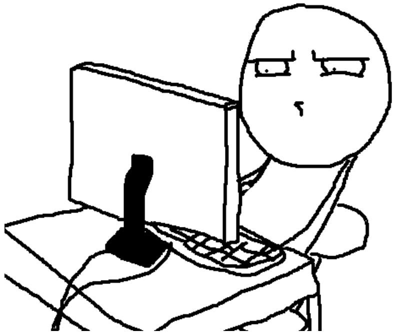
How Machine Learning can help

Data conditioning

- Identify Non linear noise coupling
- Use Deep Learning to remove noise
- Extract useful features to clean data

Signal Detection/Classification/PE

- A lot of fake signals due to noise
- Fast alert system
- Manage parameter estimation



Numbers about Virgo data

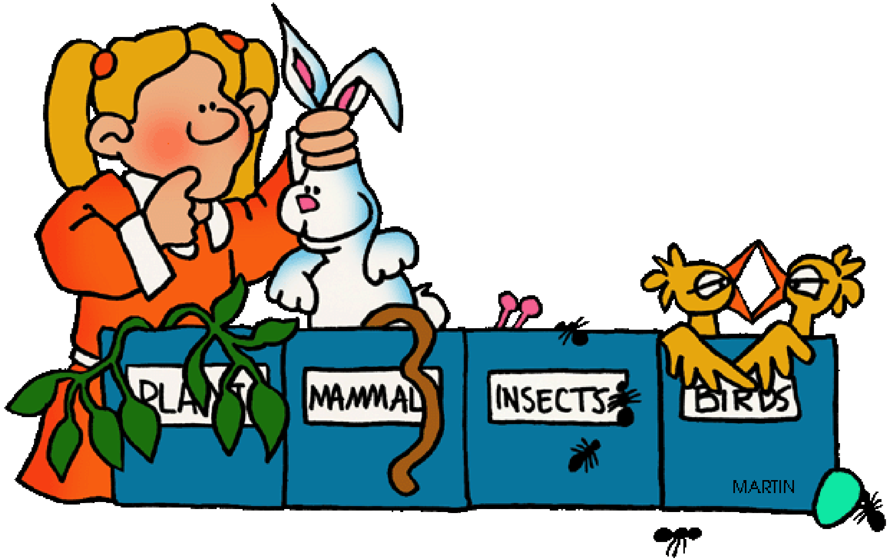
Data Stream Flux	Data on disk	Number of events	Number of glitches
• 50MB/s	• 1-3PB	• 1/week • 1/day?	• 1/sec • 0.1/sec?

Should be analysed in less than 1min

Elena Cuoco

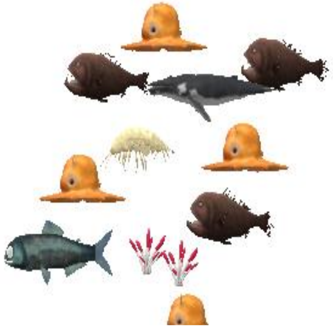
Why Signal Classification?

- If we are able to classify the noise events, we can clean the data in a fast and clear way
- We can help commissioners
- We can identify glitch families



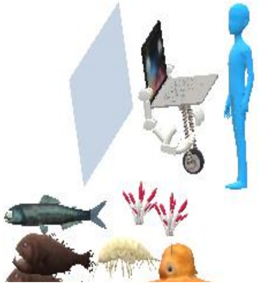
Machine learning models

Unsupervised



No label for the data

Semi-supervised

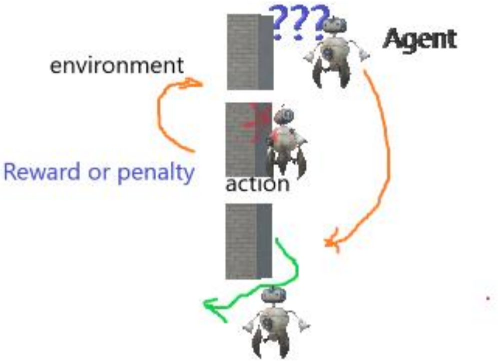


- Few labeled data
- A lot of not labeled data

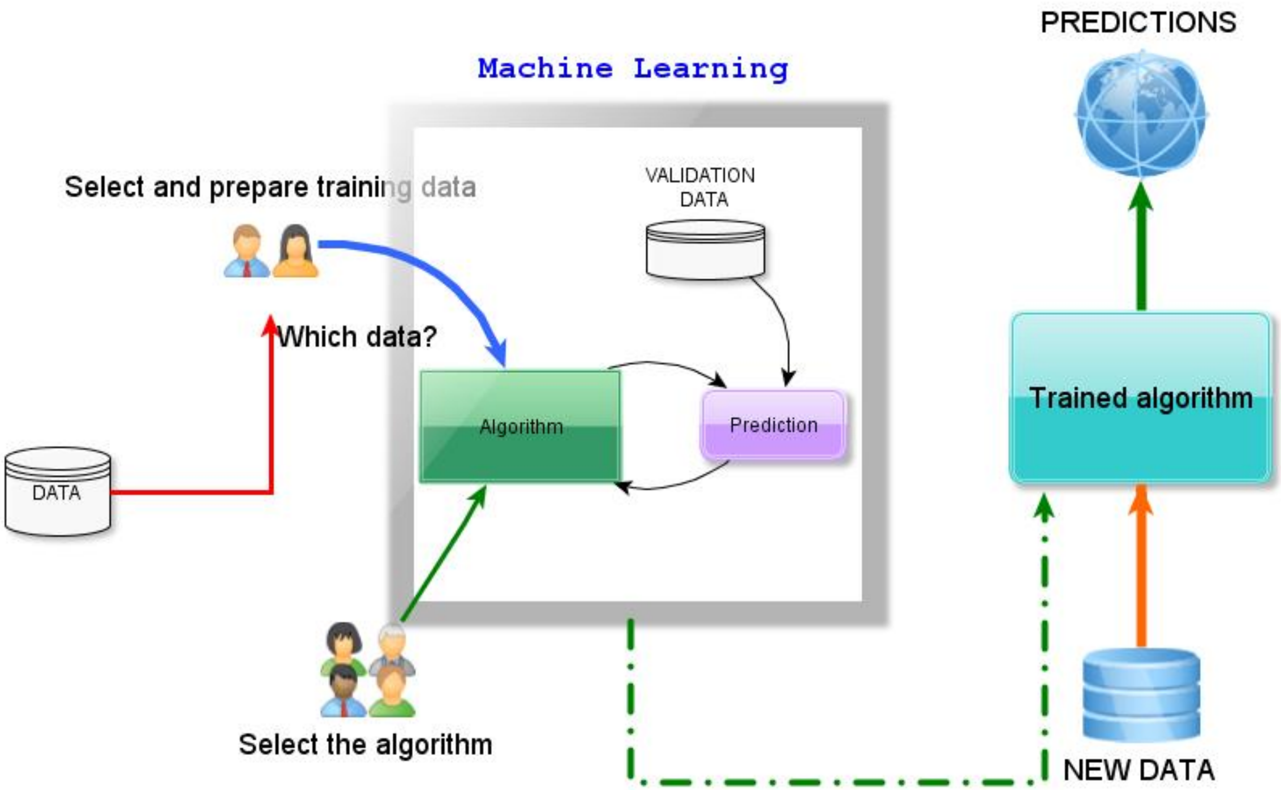
Supervised

Labeled training data

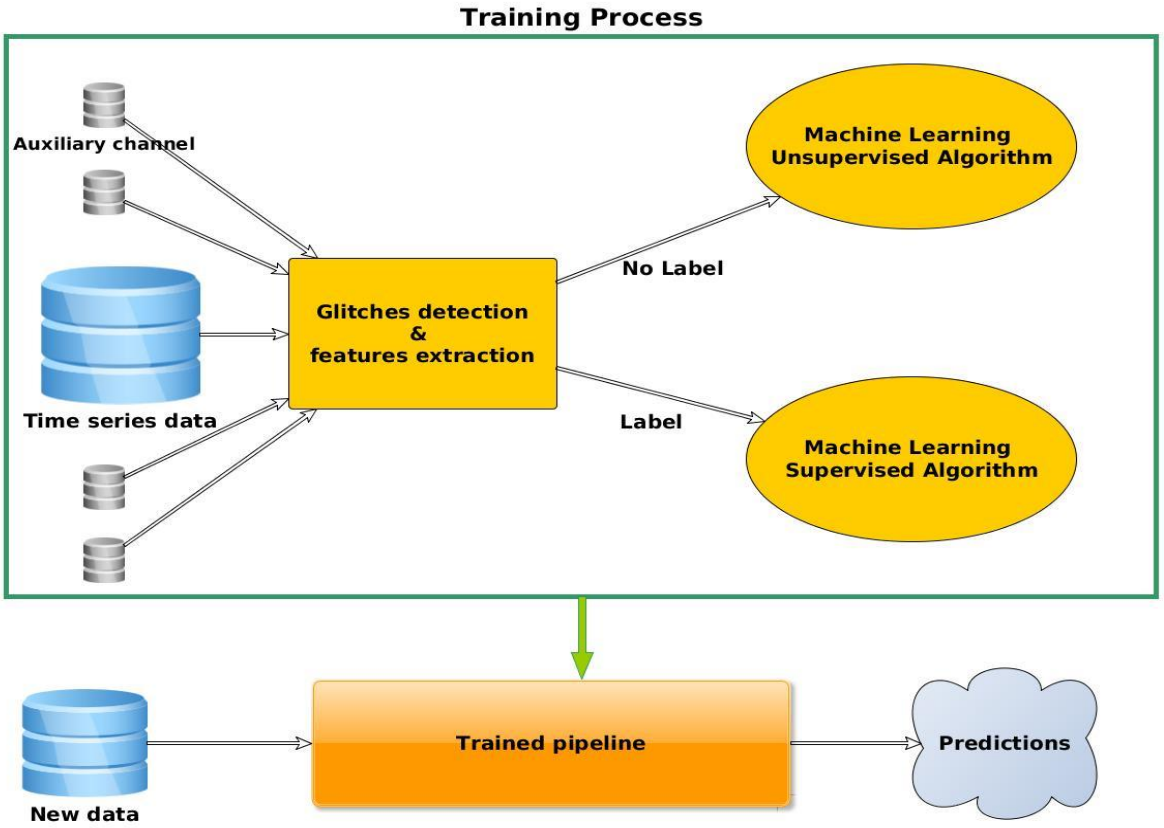
Reinforcement learning



Artificial Intelligence workflow



Glitch classification strategy for GW detectors



Elena Cuoco

Selecting data format for training set

- Images

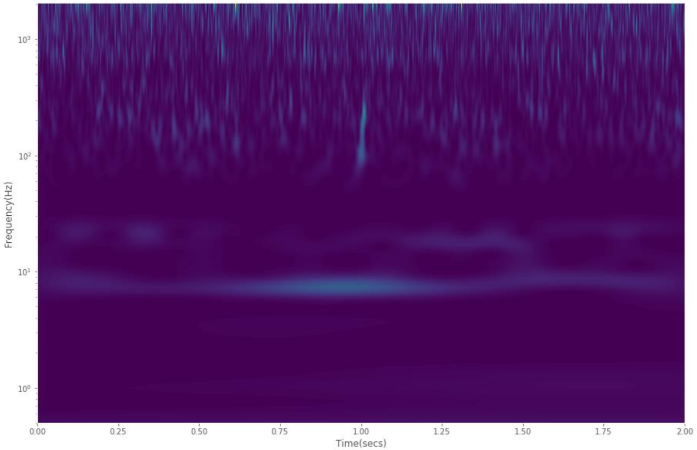
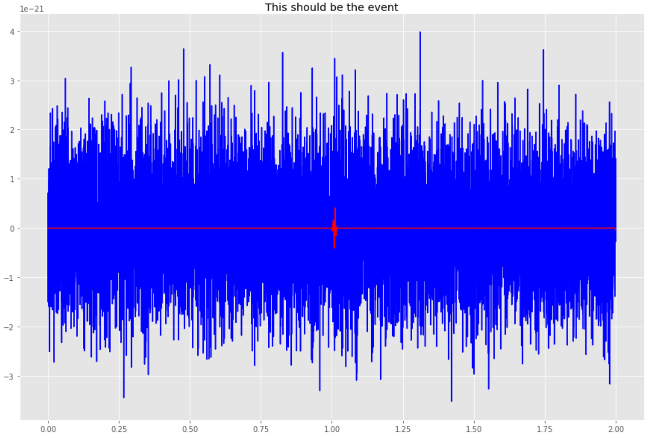


Image-based deep learning for classification of noise transients in gravitational wave detectors, Massimiliano Razzano, **Elena Cuoco**, Class.Quant.Grav. 35 (2018) no.9, 095016

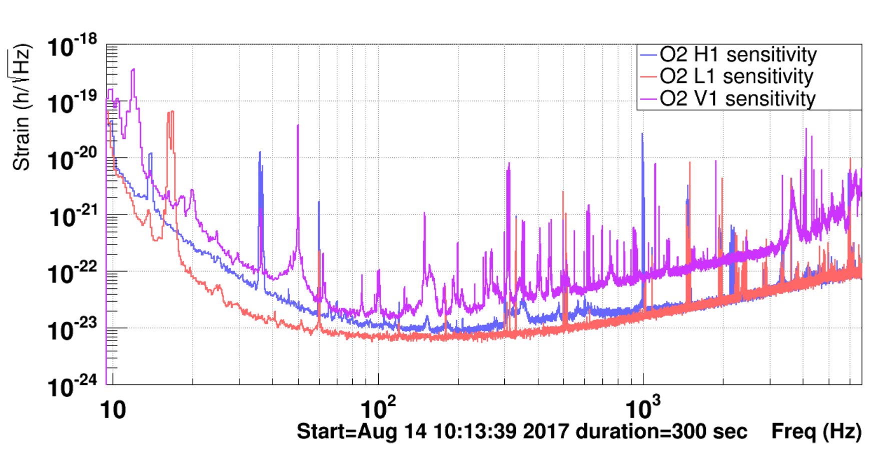
- Time series

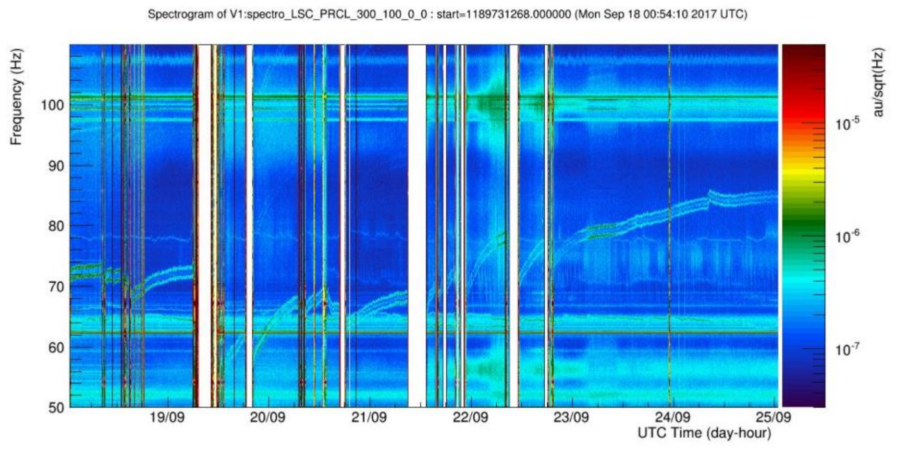


Wavelet-based Classification of Transient Signals for Gravitational Wave Detectors, **Elena Cuoco**, Massimiliano Razzano and Andrei Utina, #1570436751 accepted reviewed paper at EUSIPCO2018

Data preprocessing

- Many spectral features
- Non stationary and non linear noise





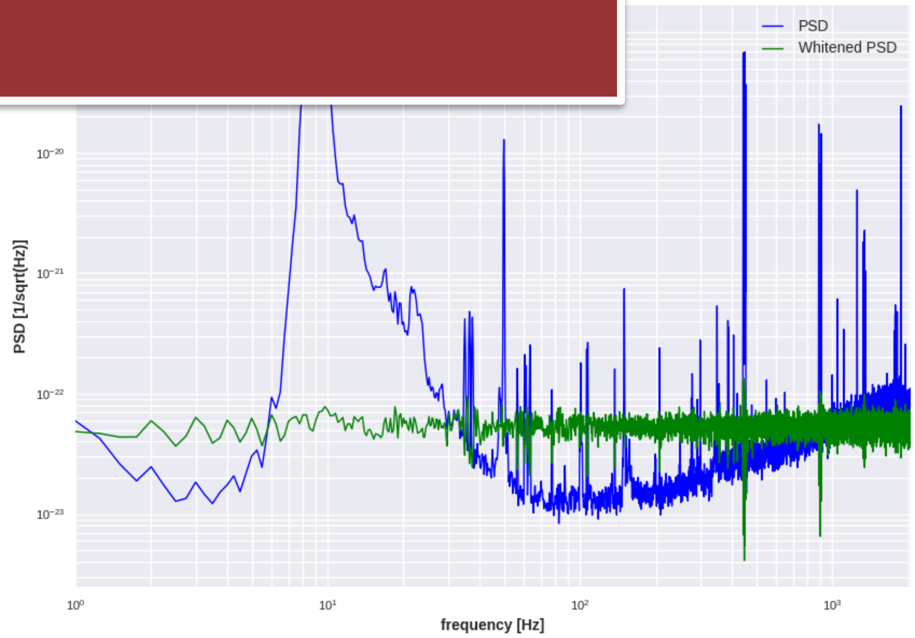
Whitening in time domain

We need parametric modeling

It can be use
line application

It can be implemented for non stationary noise

It can catch the autocorrelation function to larger lags



Elena Cuoco

ARMA parametric modeling

A general process described by a ARMA (Autoregressive Moving average) model satisfies the relation:

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + \sum_{k=0}^{q} b[k]w[n-k]$$

and its transfer function is given by $\mathscr{H}(z) = \frac{\mathscr{B}(z)}{\mathscr{A}(z)}$
where $\mathscr{A}(z) = \sum_{k=0}^{p} a[k]z^{-k}$ and $\mathscr{B}(z) = \sum_{k=0}^{q} b[k]z^{-k}$.

The PSD of the ARMA output process is

$$P_{ARMA}(f) = \sigma^2 \left|\frac{B(f)}{A(f)}\right|^2 ,$$

σ being the variance of driven white noise w, $A(f) = \mathscr{A}(2\pi i f)$
$B(f) = \mathscr{B}(2\pi i f)$.

AR parametric modeling

An AutoRegressive process is governed by this relation

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + w[n],$$

and its PSD for a process of order P is given by

$$P_{AR}(f) = \frac{\sigma^2}{|1 + \sum_{k=1}^{P} a_k \exp(-i2\pi k f)|^2}$$

Kay S 1988 Modern spectral estimation: Theory and Application Prentice Hall Englewood Cliffs

Advantages of AR modeling

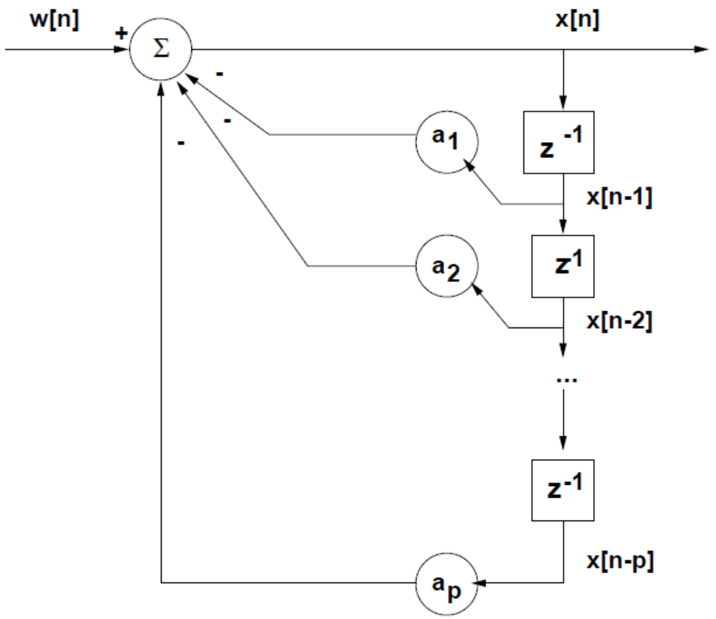
- Stable and causal filter: same solution of **linear predictor filter**

$$\hat{x}[n] = \sum_{k=1}^{P} w_k x[n-k] \,.$$

$$e[n] = x[n] - \hat{x}[n]$$

$$\varepsilon_{min} = r_{xx}[0] - \sum_{k=1}^{P} w_k r_{xx}[-k] \,,$$

$$\begin{aligned} w_k &= -a_k \\ \varepsilon_{min} &= \sigma^2 \end{aligned}$$



Wiener-Hopf equations

AR process: Burg Algorithm

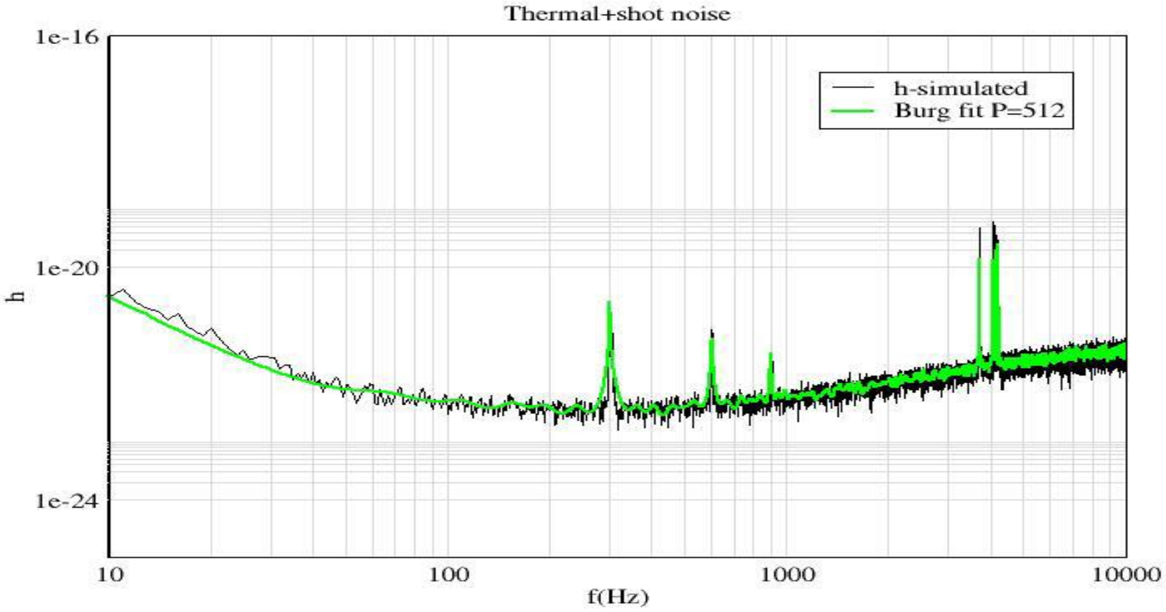
- We have to find the AR(P) parameters to fit our PSD
- An equivalent representation for an AR process is based on the value of autocorrelation function at lag 0 and a set of coefficient called reflection coefficient

  k_p for $p = 1, \ldots P$

  being P the AR order

- The kth reflection coefficient represents the partial correlation coefficient between x[n] and x[n-k]
- The Burg Algorithm estimates the reflection coefficients by autocorrelation data in a recursive way.
- This algorithm assures a minimum phase behaviour of the filter

PSD AR(P) Fit



Cuoco et al. Class.Quant.Grav. 18 (2001) 1727-1752 and
Cuoco et al.Phys.Rev.D64:122002,2001

Lattice Filter

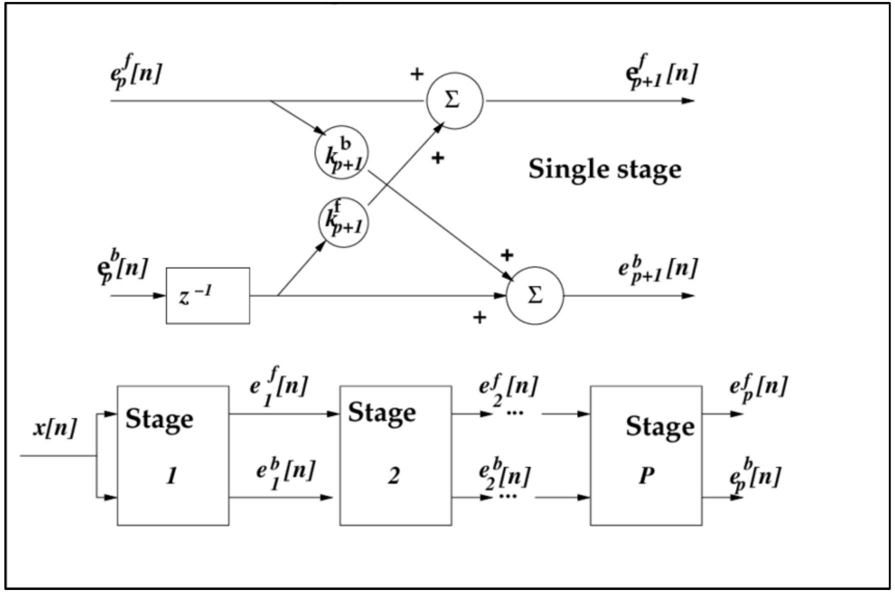
The Least Squares based methods build their cost function using all the information contained in the error function at each step, writing it as the sum of the error at each step up to the iteration n

Forgetting factor

$$\epsilon[n] = \sum_{1}^{n} \lambda^{n-1} e^2(i|n)$$

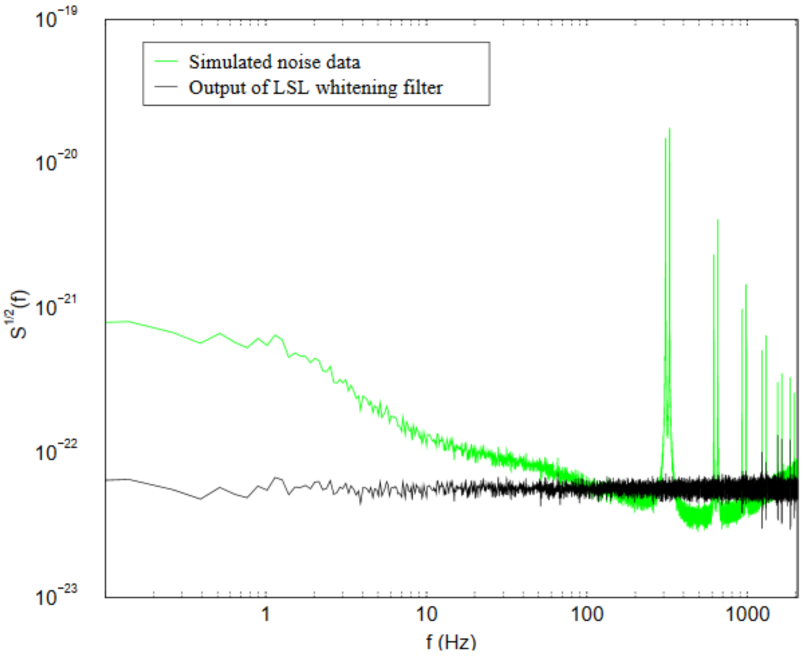
$$e(i|n) = d[i] - \sum_{k=1}^{N} x_{i-k} w_k[n],$$

Desired signal



Adaptive whitening using Lattice Filter

- If $\lambda = 1$ we are in the stationary data
- If $0 < \lambda < 1$ we can follow non stationary noise
- The Least Square Lattice filter is a modular filter with a computational cost proportional to the order P



Whitening in time domain

Static whitening

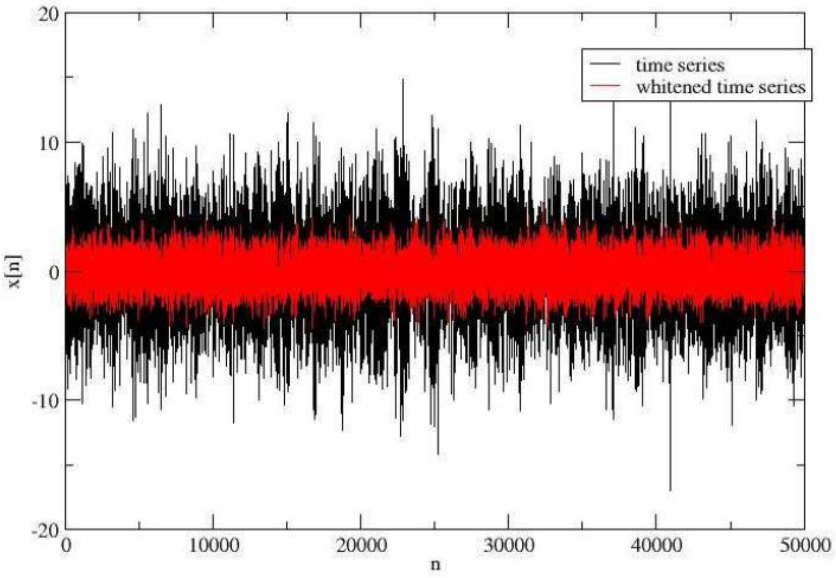
- We estimate the AR and reflection coefficients in a first part of the data
- We assume the data are stationary
- We setup a Lattice structure to run on line the whitening filter in time domain.

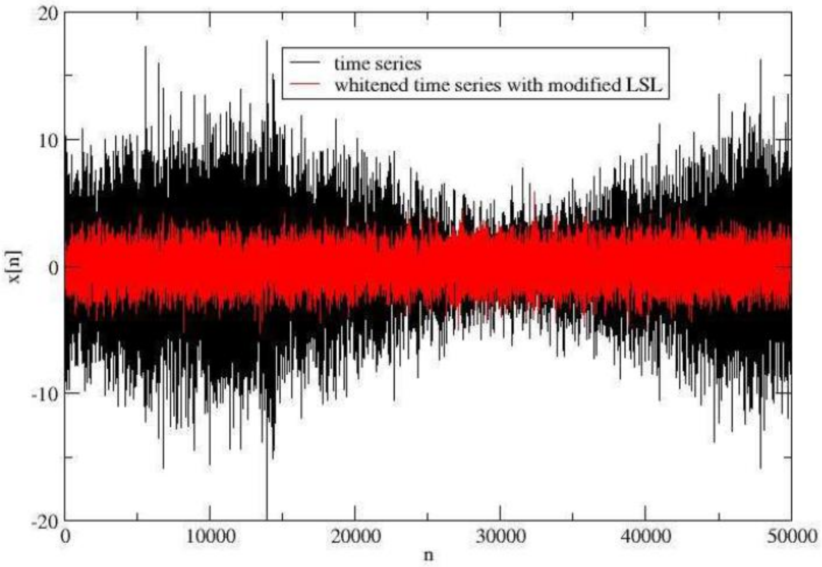
Adaptive whitening

- We make only a guess of the rmse
- We start estimating the reflection coefficients while acquiring data
- We use the forgetting factor to follow and remove the slow non stationary noise

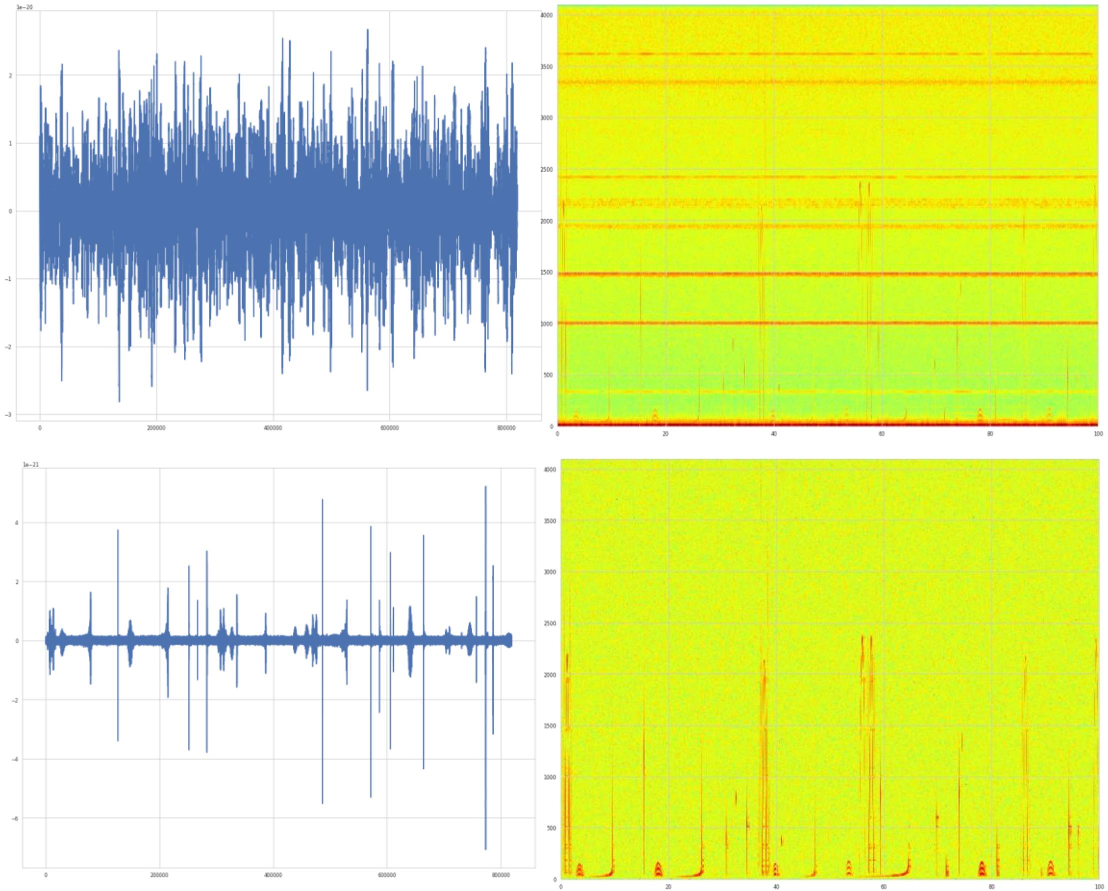
Whitened data in time domain

Example on simulated data



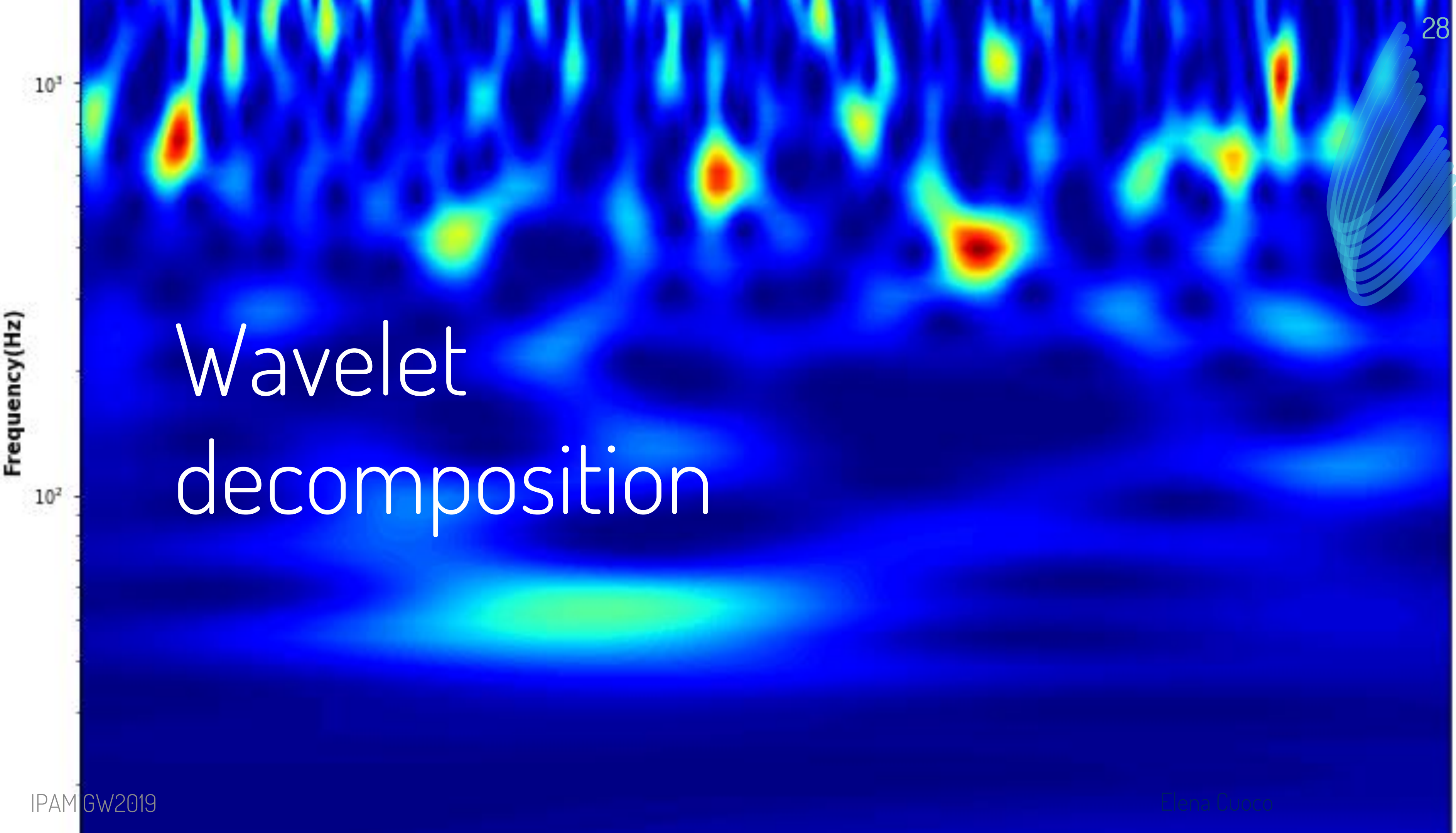


Signals in whitened data



Not Whitened

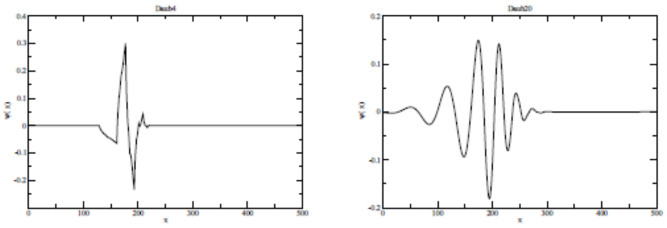
Whitened



Wavelet decomposition

Wavelet decomposition of time series

The wavelet transform replaces the Fourier transform sinusoidal waves by a family generated by translations and dilations of a window called a wavelet.



$$Wf(a,b) = < f, \psi_{a,b} > = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{b}} \psi^*\left(\frac{t-a}{b}\right) \, dt$$

Wavelet denoising

$$x_i = h_i + n_i \quad i = 0, 1, \ldots N-1$$

Wavelet transform

$$W(x) = W(h) + W(n)$$

Threshold function

$$t = \sqrt{2 \log N} \hat{\sigma}$$

Local noise

$$\hat{h} = W^{-1}(T(Wx))$$

Dohone and Johnston proposed two different thresholding strategy: the soft thresholding and the hard thresholding. Given a threshold t and w the wavelet coefficient, the hard threshold for the signal is w if $|w| > t$, and is 0 if $|w| < t$. The soft threshold for the signal is $sign(w)(|w| - t)$ if $|w| > t$ and is 0 if $|w| < t$.

Wavelet Detection filter as Event Trigger Generator

$$E_s = \sqrt{\sum_{k,j} w_{k,j}^2}$$

- Select highest values

$\propto$ Energy of the signal

$$SNR = \frac{E_s}{\sigma}$$

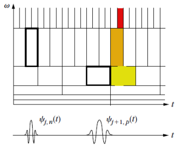
- Reconstruct a proto-SNR

$\propto$ SNR of the signal

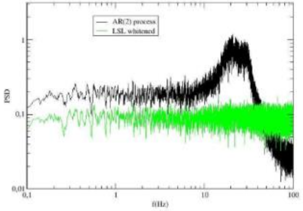
Wavelet Detection Filter (WDF) workflow

$$x_i = h_i + n_i, \quad i = 0, 1, \ldots N-1,$$

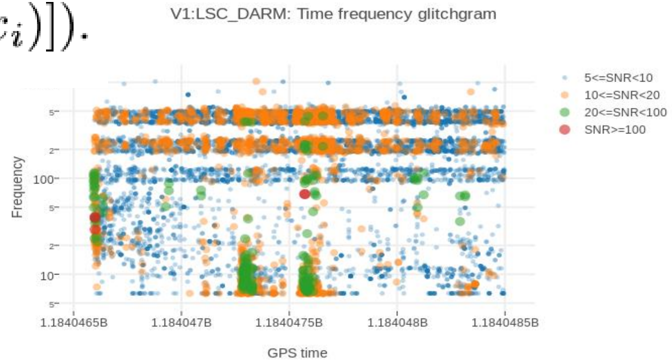
$$Wf(a,b) = \langle f, \psi_{a,b} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{b}} \psi^* \left(\frac{t-a}{b} \right) \, dt,$$



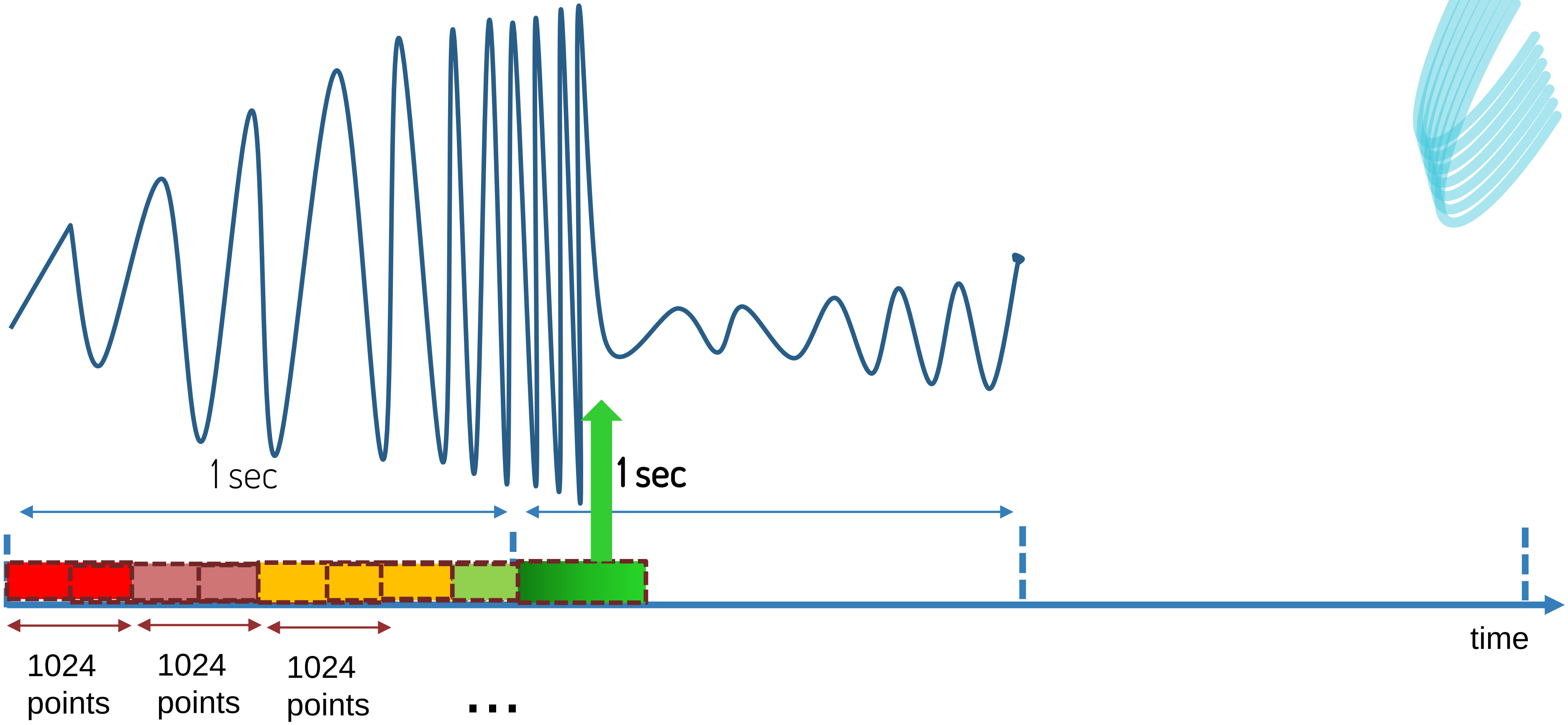
Data → Whitening in time domain → Wavelet transform → De-noising → Parameter estimation → Trigger list



$$\hat{h}_i = W^{-1}(t[W(x_i)]).$$

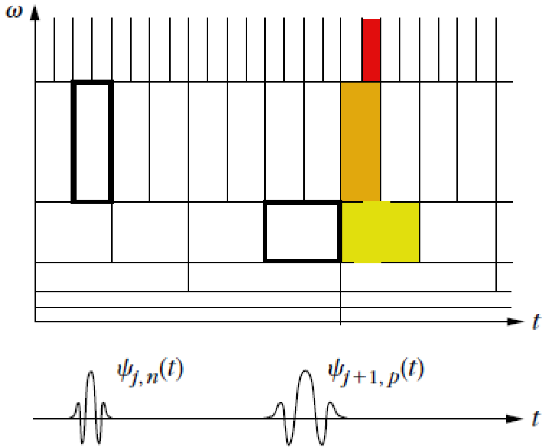


WDF: how it works



WDF waveform extraction

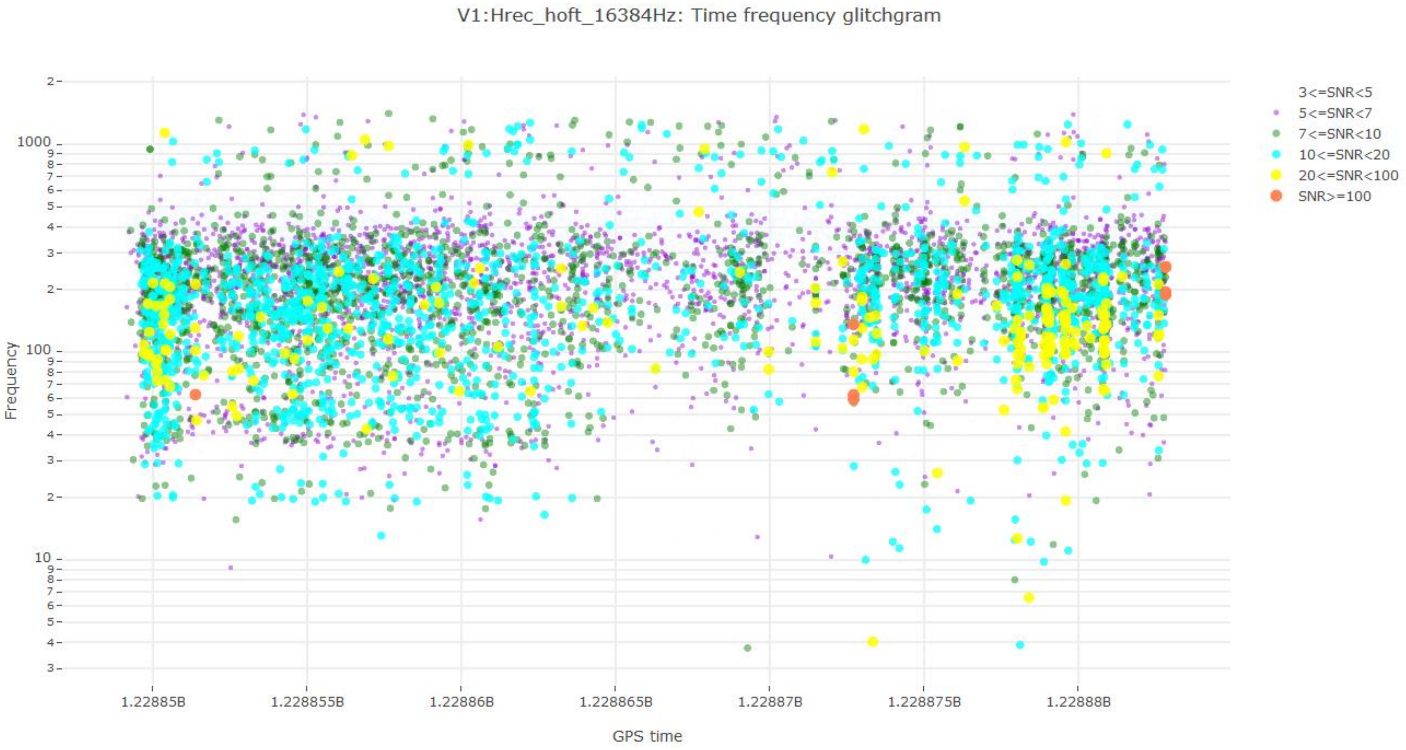
- ✓ Wavelet transform in the selected window size
- ✓ Retain only coefficients above a fixed threshod (Donoho-Johnston denoise method)
- ✓ Create a metrics for the energy using the selected coefficients and give back the trigger with all the wavelet coefficients.
- ✓ In the wavelet plane, select the highest values coefficients to build the event
- ✓ Inverse wavelet transform
- ✓ Estimate mean and max frequency and snr max of the cleaned event



Gps, duration, snr, snr@max, freq_mean, freq@max, wavelet type triggered + corresponding wavelets coefficients.

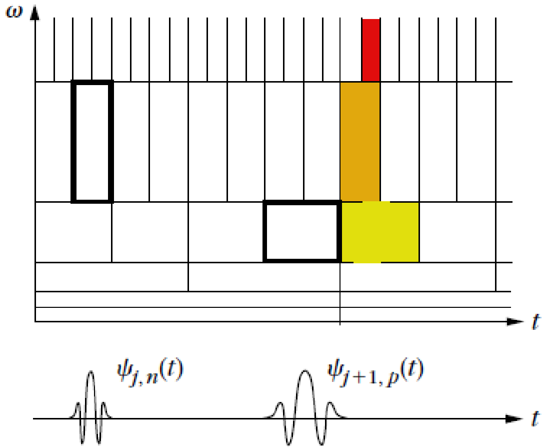
Glitchgram

Time-Frequency distribution by SNR slice



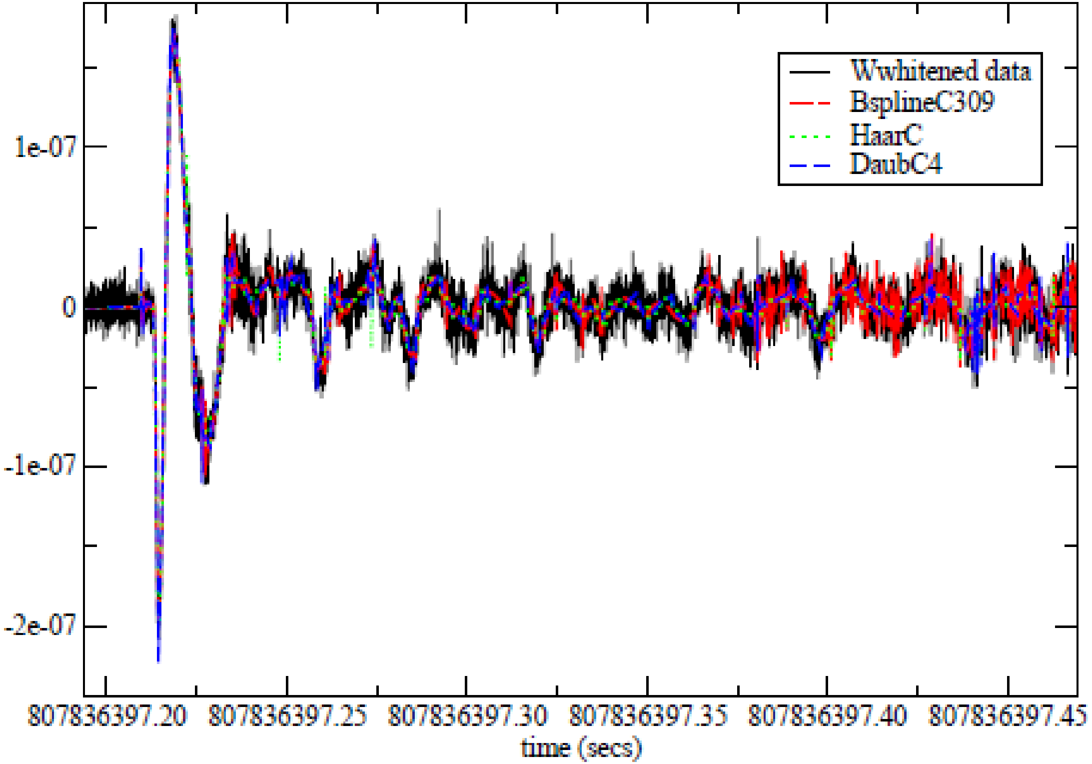
WDF waveform extraction

- ✓ Wavelet transform in the selected window size
- ✓ Retain only coefficients above a fixed threshod (Donoho-Johnston denoise method)
- ✓ Create a metrics for the energy using the selected coefficients and give back the trigger with all the wavelet coefficients.
- ✓ In the wavelet plane, select the highest values to build the event
- ✓ Inverse wavelet transform
- ✓ Estimate mean and max frequency and snr max of the cleaned event



Gps, duration, snr, snr@max, freq_mean, freq@max, wavelet type triggered + corresponding wavelets coefficients.

Waveform reconstruction



Elena Cuoco

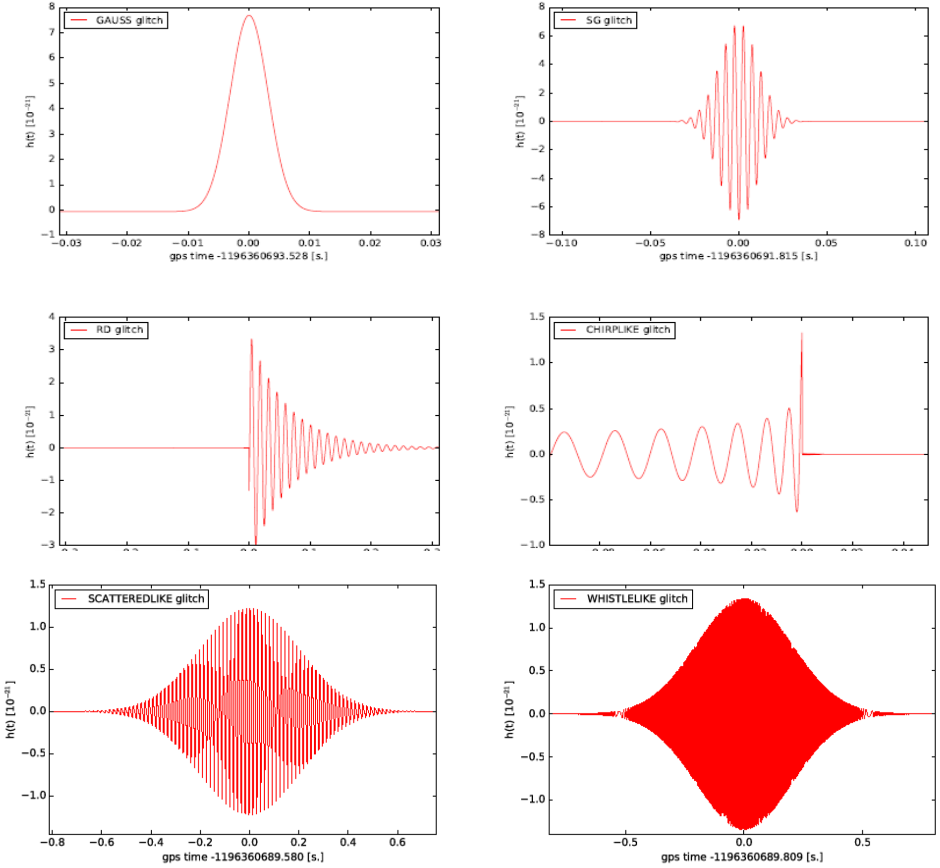
Test on simulated data sets

To test the pipeline, we prepared ad-hoc simulations

Simulate colored noise using public H1 sensitivity curve

Add 6 different classes of glitch shapes

Simulated signal families



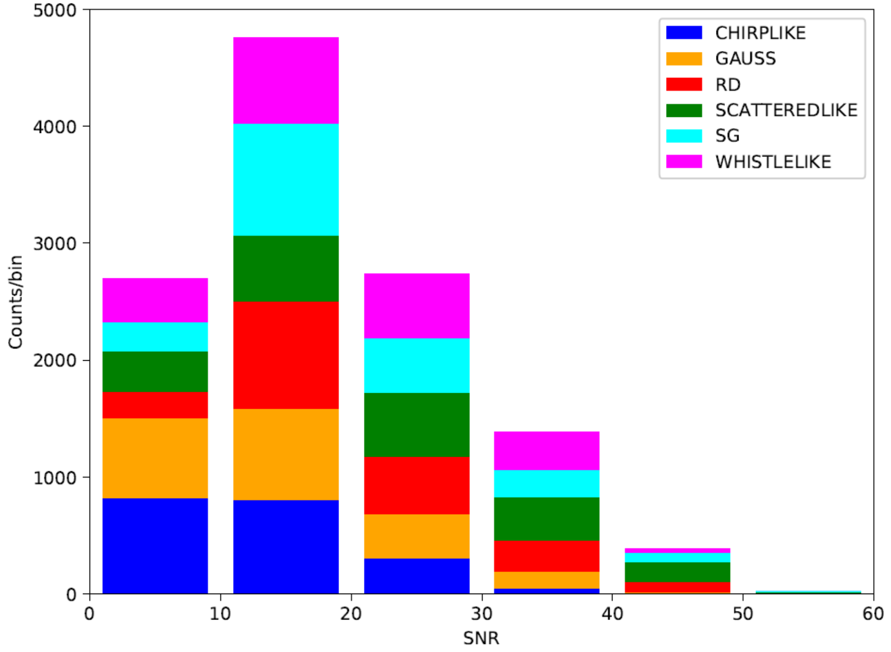
Waveform
Gaussian
Sine-Gaussian
Ring-Down
Chirp-like
Scattered-like
Whistle-like
NOISE (random)

To show the glitch time-series here we don't show the noise contribution

Razzano M., Cuoco E. CQG-104381.R3

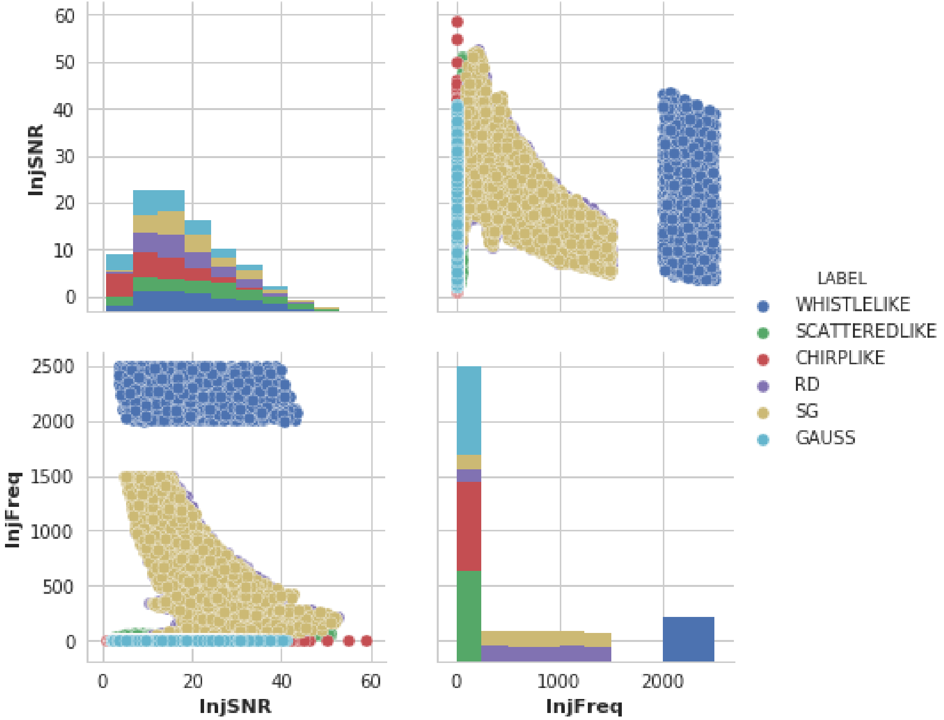
Signal distribution

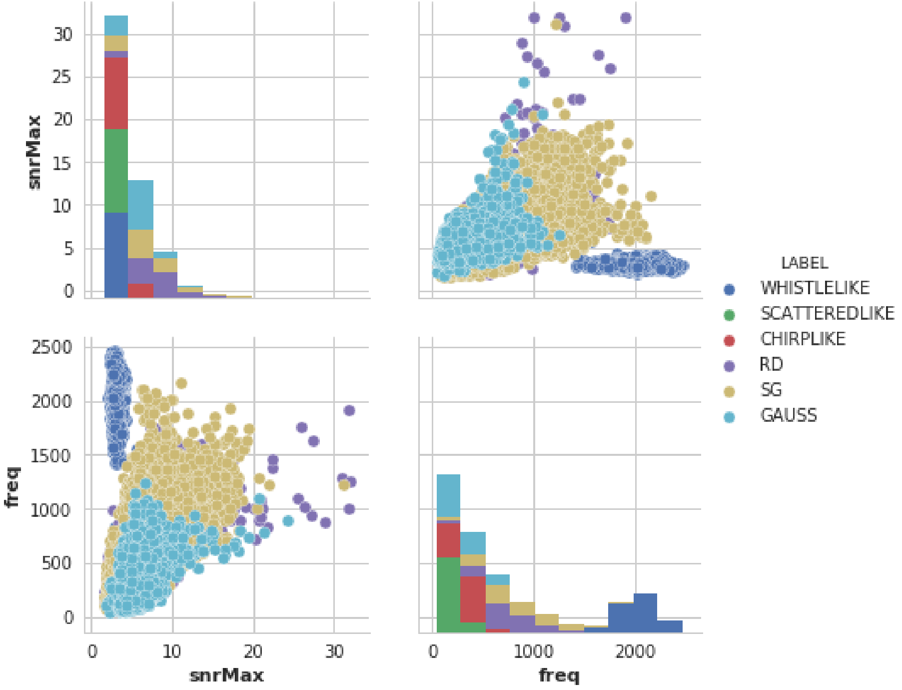
- Simulated time series with 8kHz sampling rate
- Glitches distributed with Poisson statistics m=0.5 Hz
- 2000 glitches per each family
- Glitch parameters are varied randomly to achieve various shapes and Signal-To-Noise ratio



WDF results

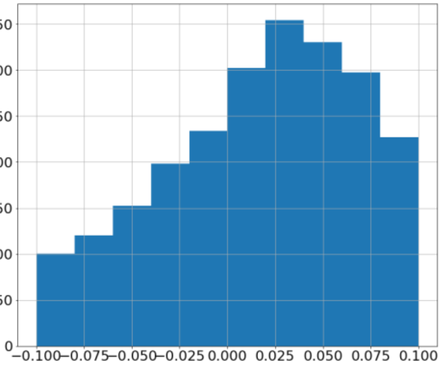
- Detected 97% of injected signals (some with SNR=1)
- False alarm rate: 10% for a time window shift of 1sec
- Good parameters estimation



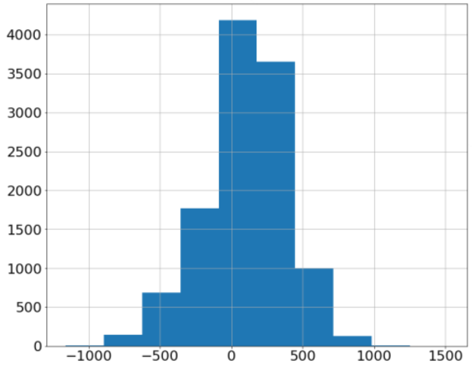


Parameter estimations in 0.1sec

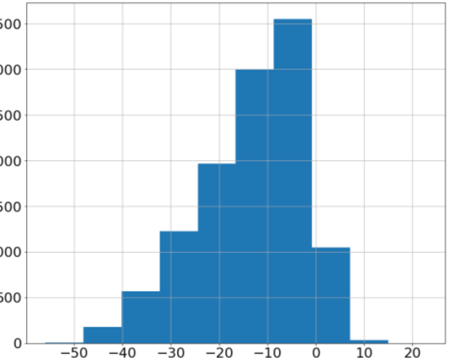
Time diff



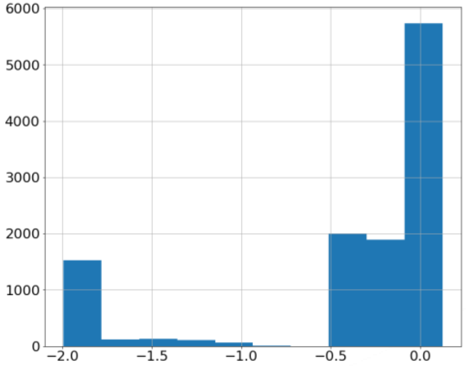
Frequency diff



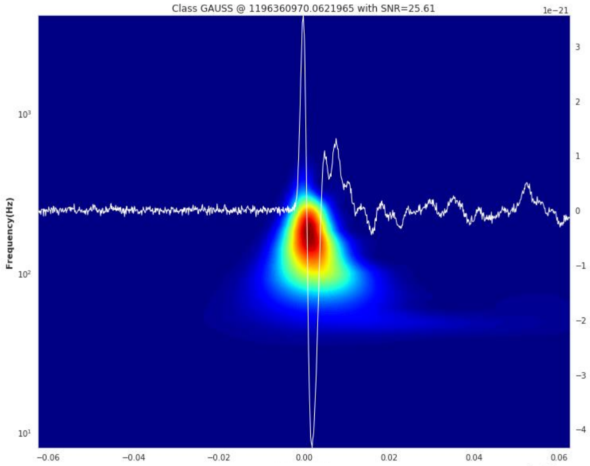
SNR diff



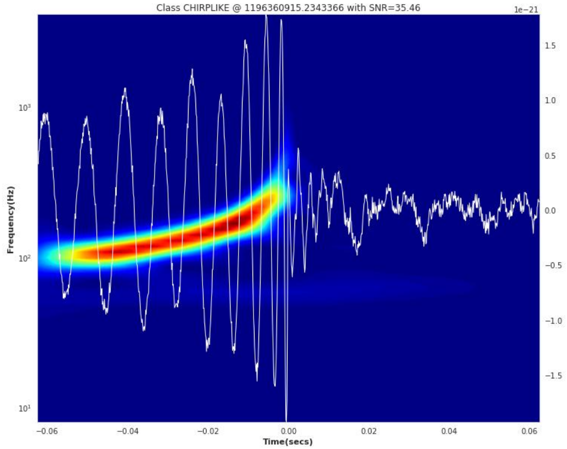
Duration diff

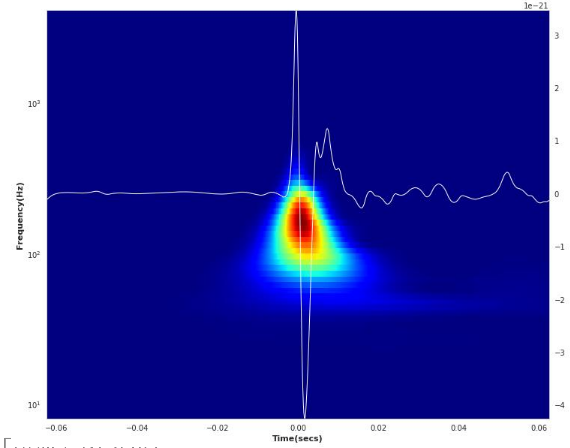


Waveform reconstruction: example

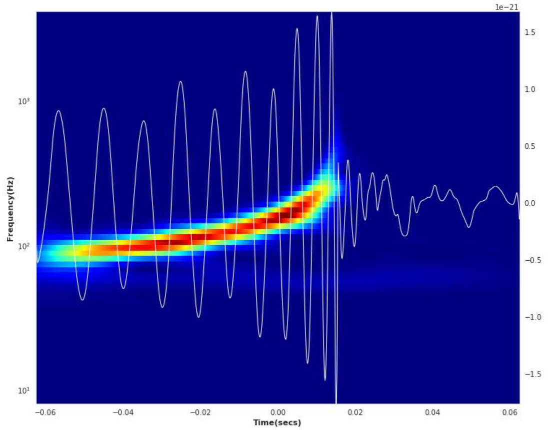


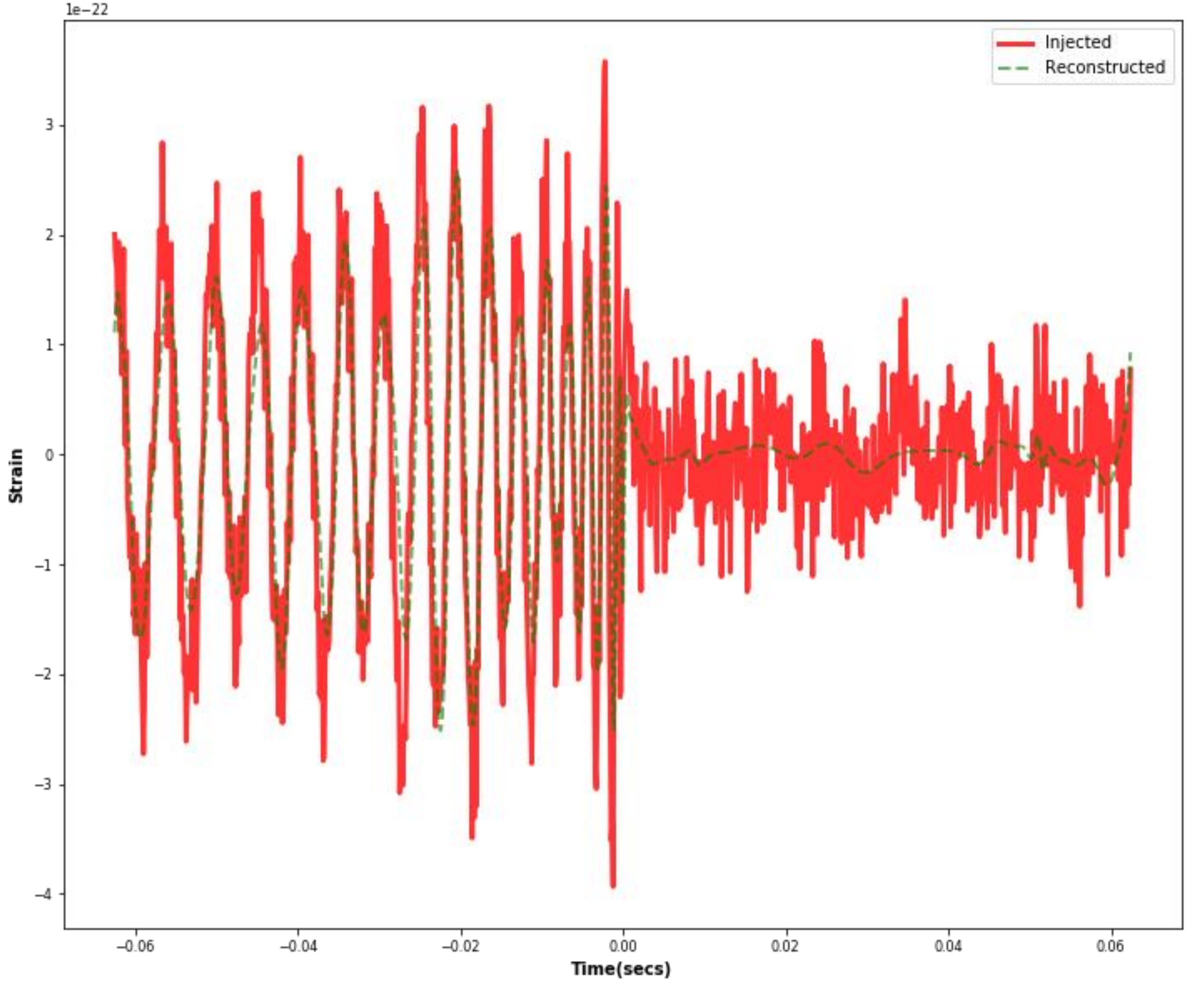
Injected



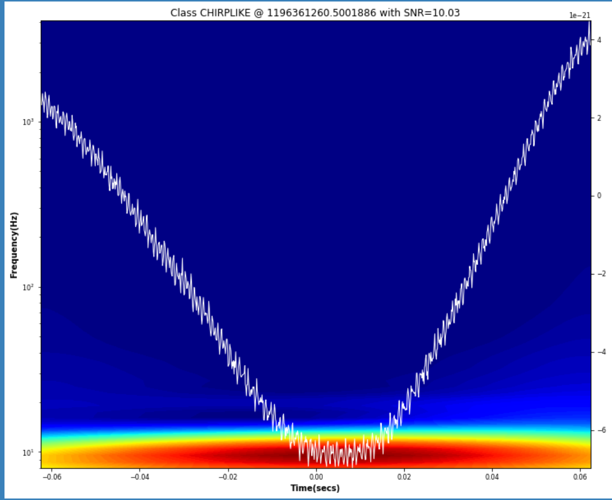


Detected



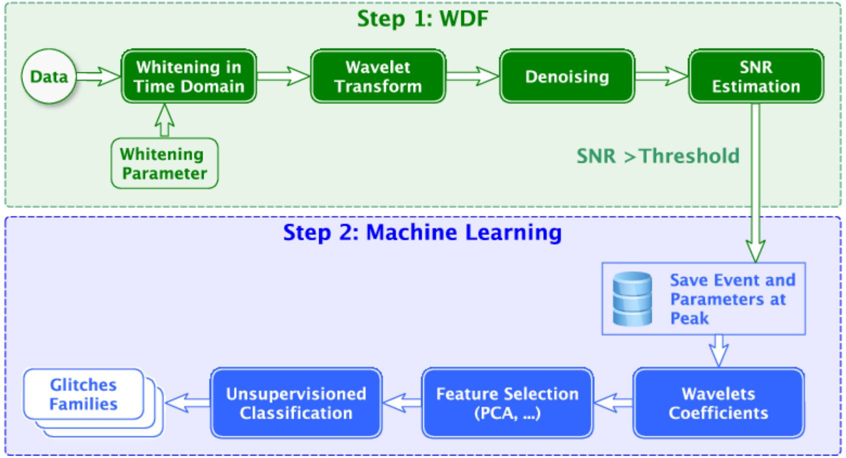


Injection and Reconstruction in perfect match

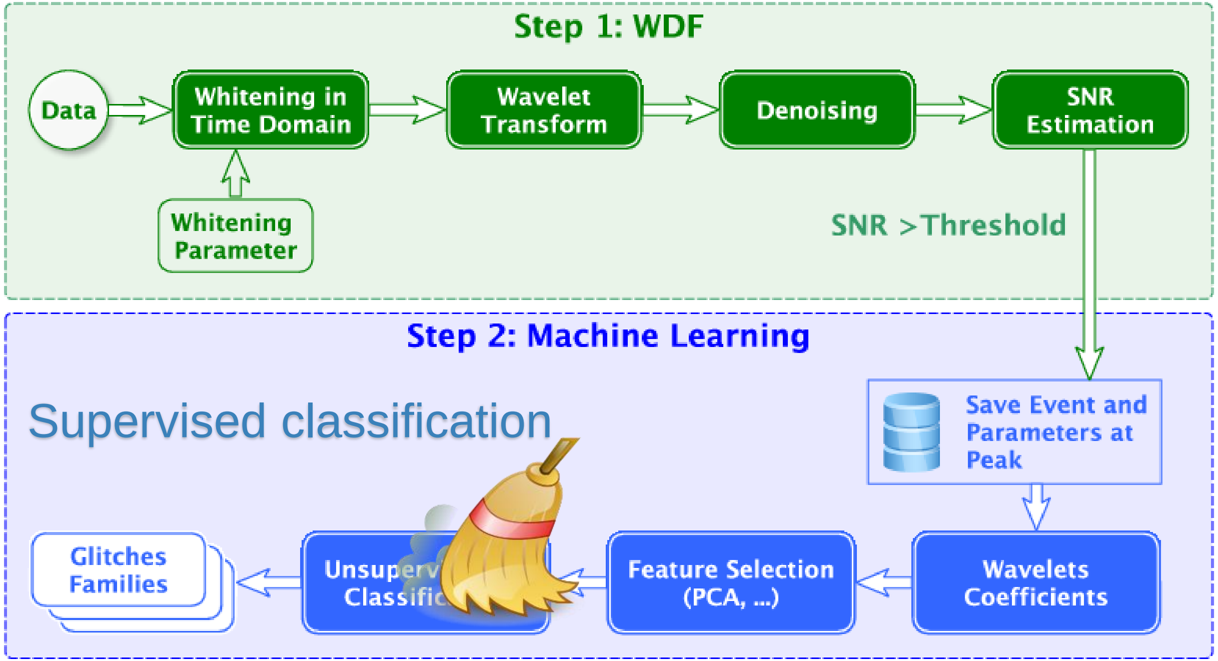


Glitch classification

- Unsupervised on Simulated data:
  - Classification methods for noise transients in advanced gravitational-wave detectors
    Jade Powell, Daniele Trifirò, **Elena Cuoco**, Ik Siong Heng, Marco Cavaglià, Class.Quant.Grav. 32 (2015) no.21, 215012
- Unsupervised on Real data (ER7):
  - Classification methods for noise transients in advanced gravitational-wave detectors II: performance tests on Advanced LIGO data, Jade Powell, Alejandro Torres-Forné, Ryan Lynch, Daniele Trifirò, **Elena Cuoco**, Marco Cavaglià, Ik Siong Heng, José A. Font, Class.Quant.Grav. 34 (2017) no.3, 034002

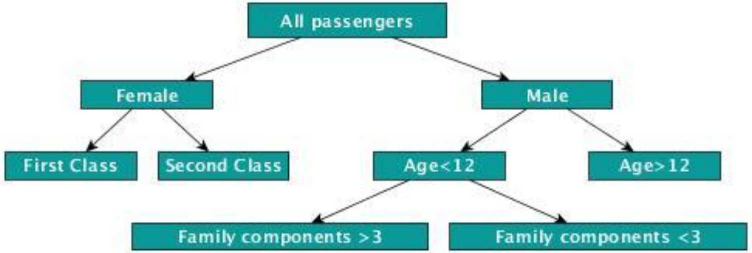


Wavelet Detection Filter and XGBoost (WDFX)



Supervised Classification: eXtreme Gradient Boosting

- https://github.com/dmlc/xgboost
- Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, 2016
- XGBoost originates from research project at University of Washington, see also the Project Page at UW.



Tree Ensemble

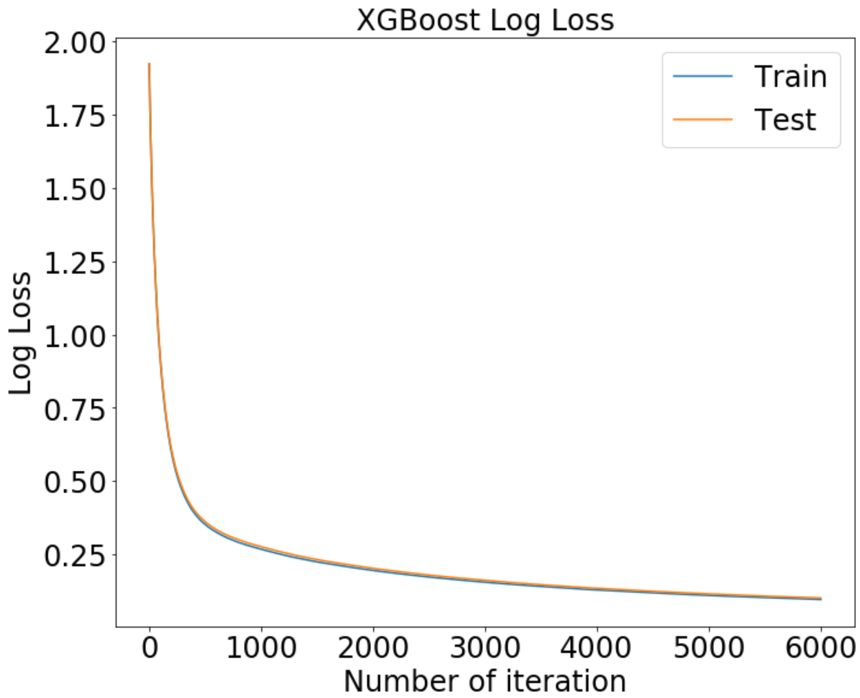
$$y_n = \sum_{k=1}^{K} f_k(x_n)$$

dmlc XGBoost

Xgboost

$$L = -\frac{1}{N}\sum_{1}^{N}((y_i \log(p_i) + (1 - y_i)(log(1 - p_i)) + \Omega$$

Train/validation/test set: 70/15/15



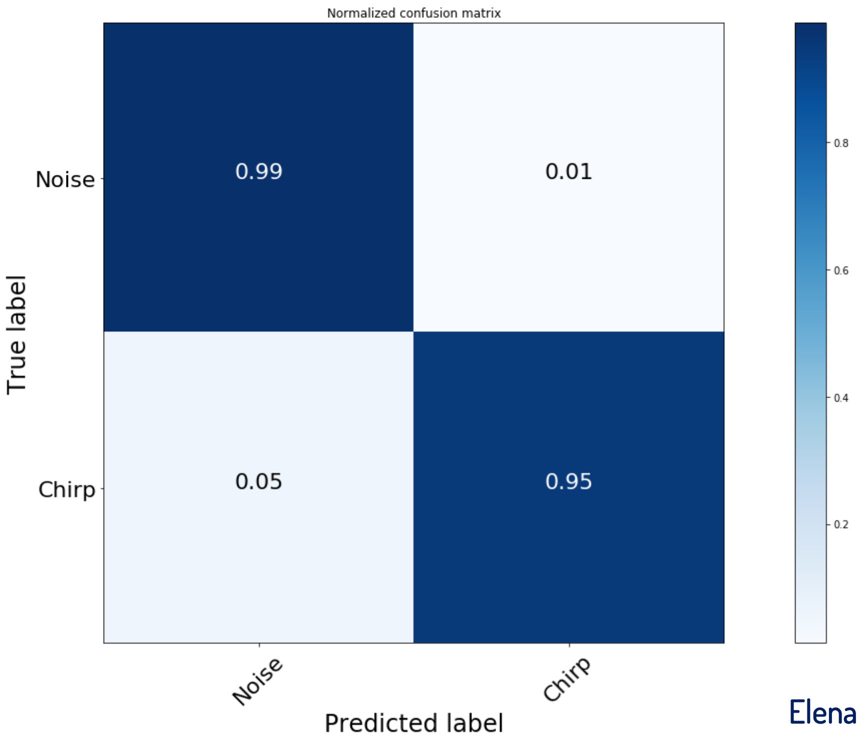
task	Classes	Learning-rate	Max_depth	estimators
Binary	2	0.01	7	5000
Multi-label	7	0.01	10	6000

WDFX: Binary Classification Results

Chirp-like signals
OR
Noise

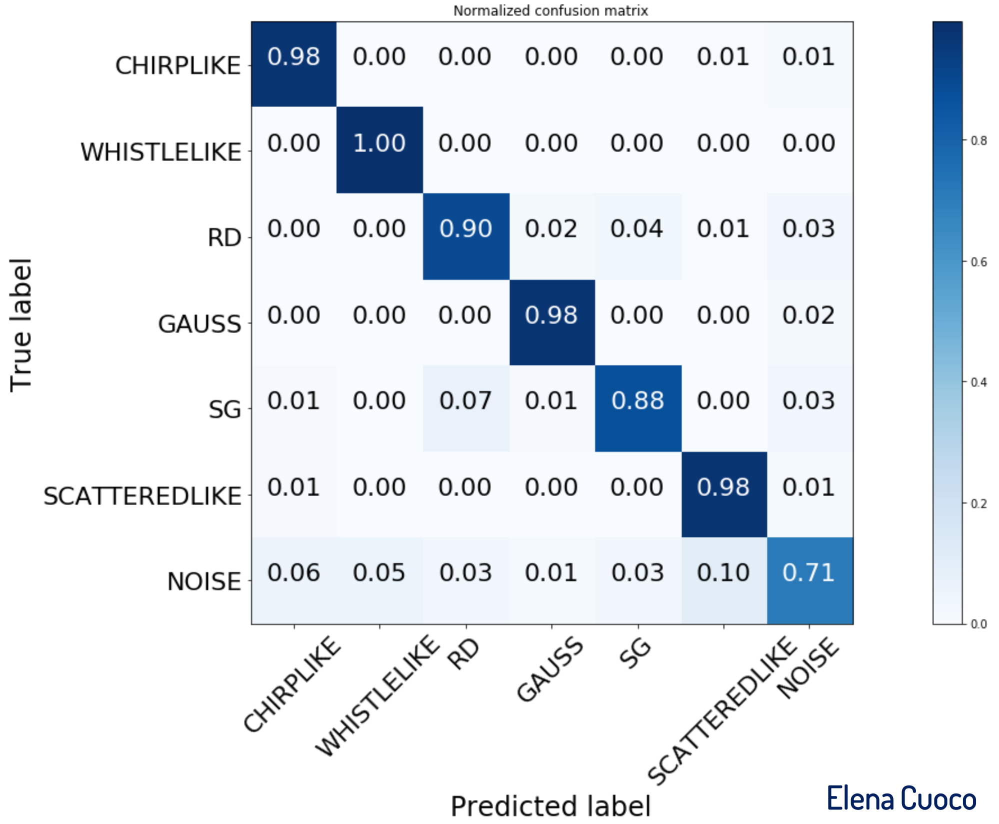
Overall accuracy >98%

Updated results



WDFX Results: Multi-Label Classification

Overall accuracy >93%



Updated results

Cuoco, Razzano in preparation

Wavefier

Real time Gravitational Wave transient signal classifier

A project in collaboration with LAPP and Trust-IT services

H2020-ASTERICS project brings together for the first time scientists and communities from astronomy, astrophysics, particle astrophysics & big data.
http://www.asterics2020.eu

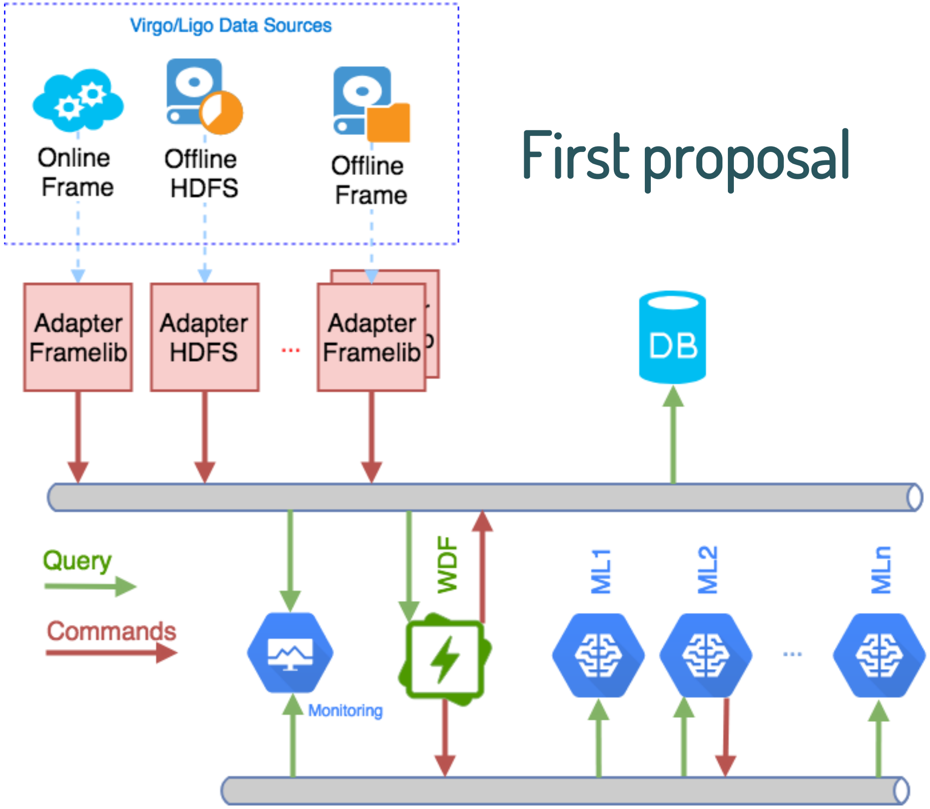
H2020-Astronomy ESFRI and Research Infrastructure Cluster
(Grant Agreement number: 653477).

Wavefier

Real time Gravitational Wave transient signal classifier

First proposal

- release an end to end framework for the glitches identification, classification and archiving ML classification schemes for GW glitches.
- To evaluate possible HPC solutions for DL pipelines for online glitch classification.



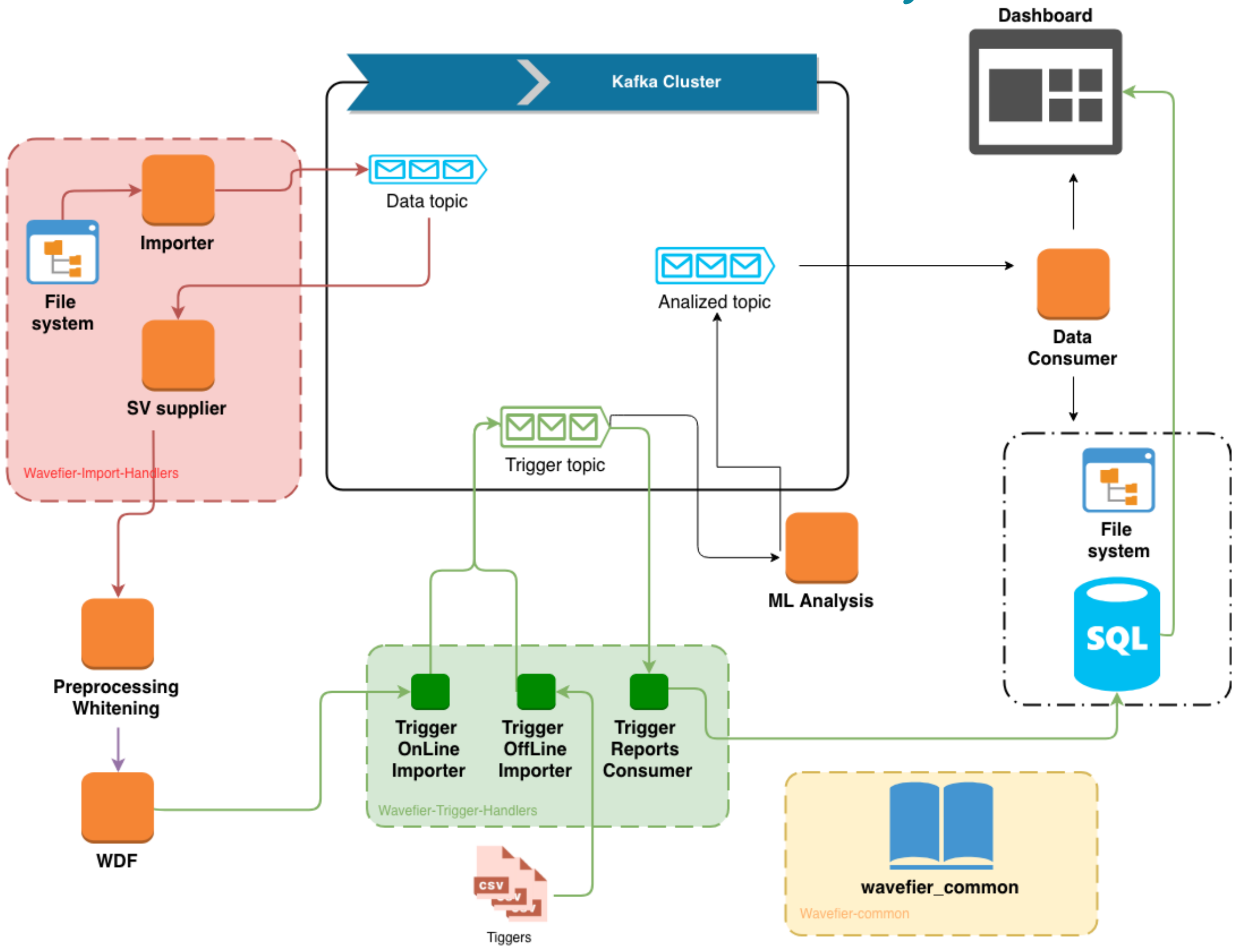
LAPP, Trust-IT Services company, EGO

Wavefier

Real time Gravitational Wave transient signal classifier

Wavefier: real time analysis

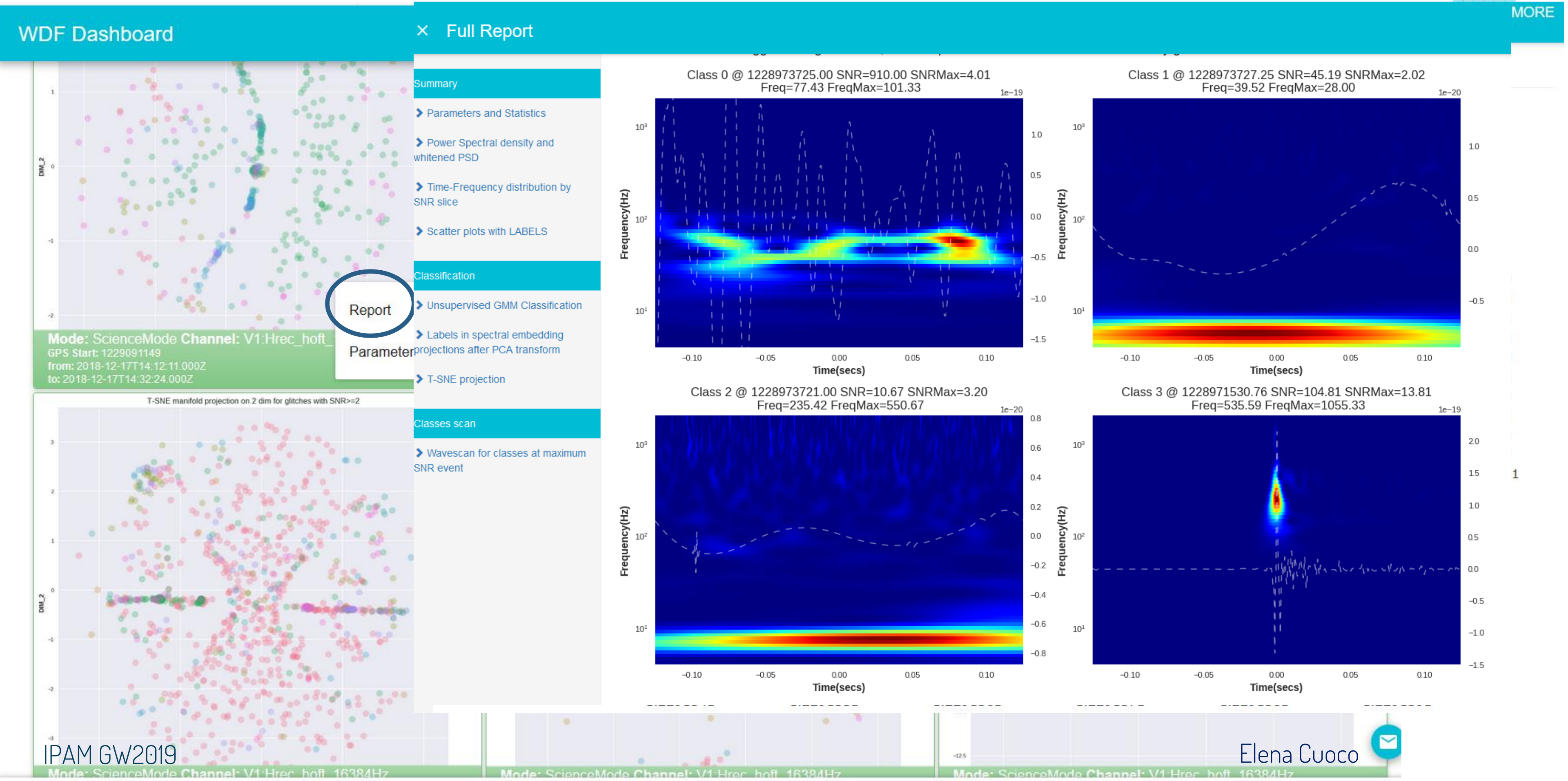
Stream to Stream



Different Machine Learning approaches

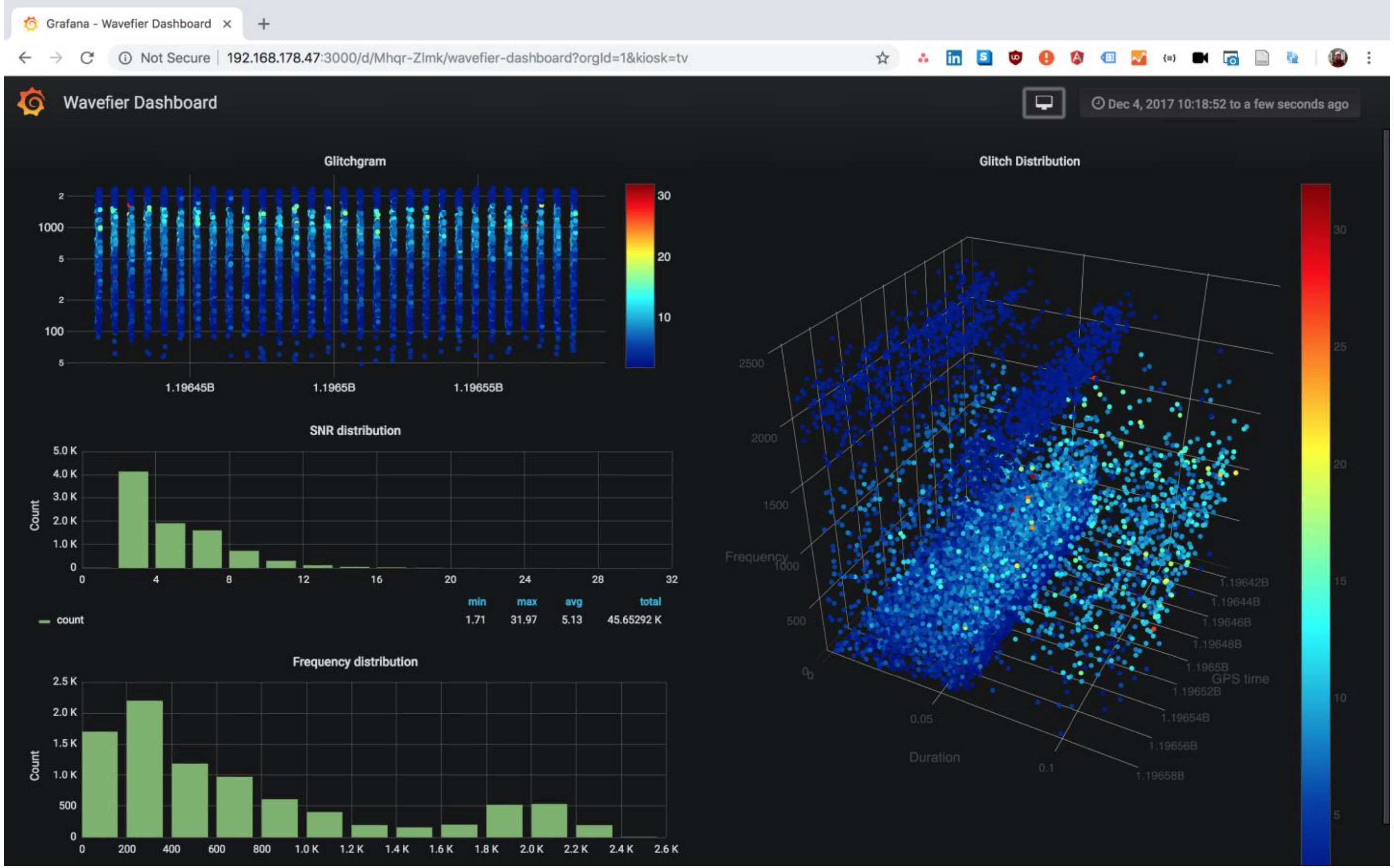
- Wavelet coefficients and some meta-parameters
- Reconstrutcted waveform in 1-D
- Images and CNN
- Transfer learning
- Semi supervised
- GANs to have a larger data set

Showing the results



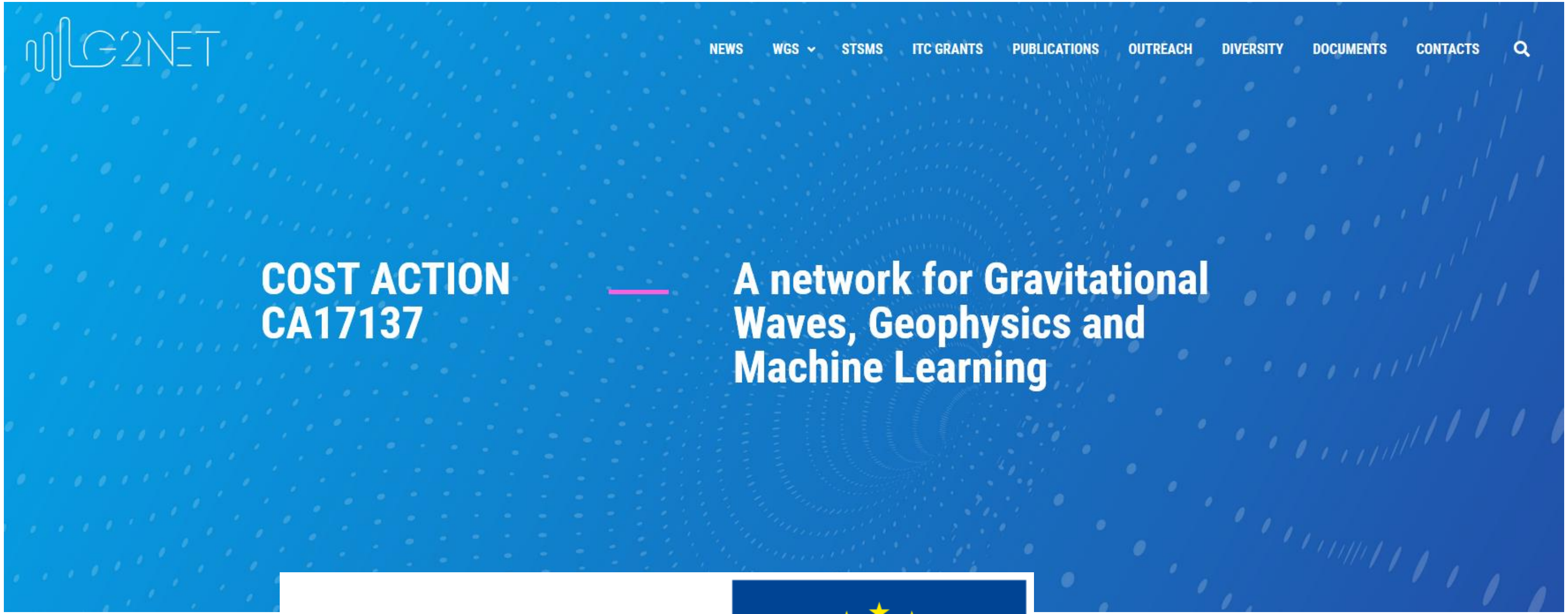
Wavefier
Real time Gravitational Wave transient signal classifier

Wavefier dashboard



Grafana. Web based dashboard

http://www.g2net.eu/



EUROPEAN COOPERATION IN SCIENCE & TECHNOLOGY

Elena Cuoco
Twitter: @elenacuoco
Website: www.elenacuoco.com

Thanks to Alberto Iess, Emanuel Marzini, Filip Morawski, Alessandro Petrocelli, Massimiliano Razzano, Alessandro Staniscia, Alejandro Torres, Andrei Utina

Thanks!