Validation of Speech Processing Techniques for Real-Time Detection of Gravitational Waves

Francesco P. Barone[3], Marco Russo[3,4], Daniele Dell'Aquila[1,2]

[1]Dipartimento di Chimica e Farmacia, Università degli Studi di Sassari, Sassari, Italy
[2]INFN-Laboratori Nazionali del Sud, Catania, Italy
[3]Dipartimento di Fisica e Astronomia, Università degli Studi di Catania, Catania, Italy
[4]INFN-Sezione di Catania, Catania, Italy

ddellaquila@uniss.it

Second generation ground-based interferometers for gravitational wave detection (**aLIGO**, **aVirgo**) → capable to probe an extremely large volume of space and an unprecedently large volume of the astrophysical parameter space → a great challenge for high-performance computing and high-throughput computing!

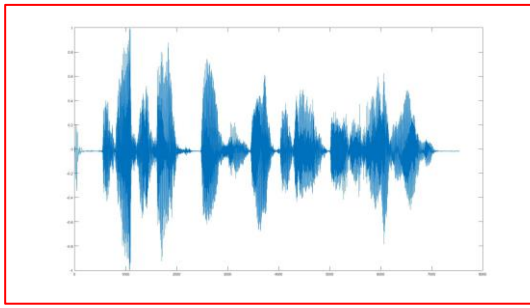
a key aspect → enabling *real-time* detection of gravitational wave signals and possibly their parameter estimation.

In this context →**Artificial intelligence** → particularly promising to accelerate the search for gravitational wave signals even in online applications →

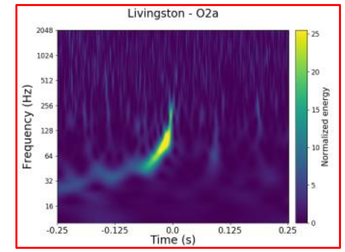
H. Gabbard et al., Phys. Rev. Lett. 120 (2018) 141103.
D. George and E.A. Huerta, Phys. Rev. D 97 (2018) 044039.
D. George and E.A. Huerta, Phys. Lett. B 778 (2018) 64.

→ the application of existing CNN algorithms in GW interferometers is made particularly challenging by the continuous nature of the stream of data.

Our novel framework → inspired by *speech processing* techniques (short term analysis)



voiced



unvoiced (noise)

→ key aspects and novelties:

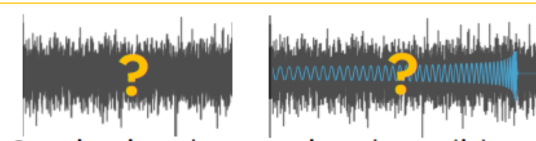
1) particularly suitable for continuous analysis;
2) layered approach;
3) high-modularity;
4) low computational complexity → ideal for real-time applications.

Università degli Studi di Sassari & INFN-LNS

level 0 trigger

level 1 trigger

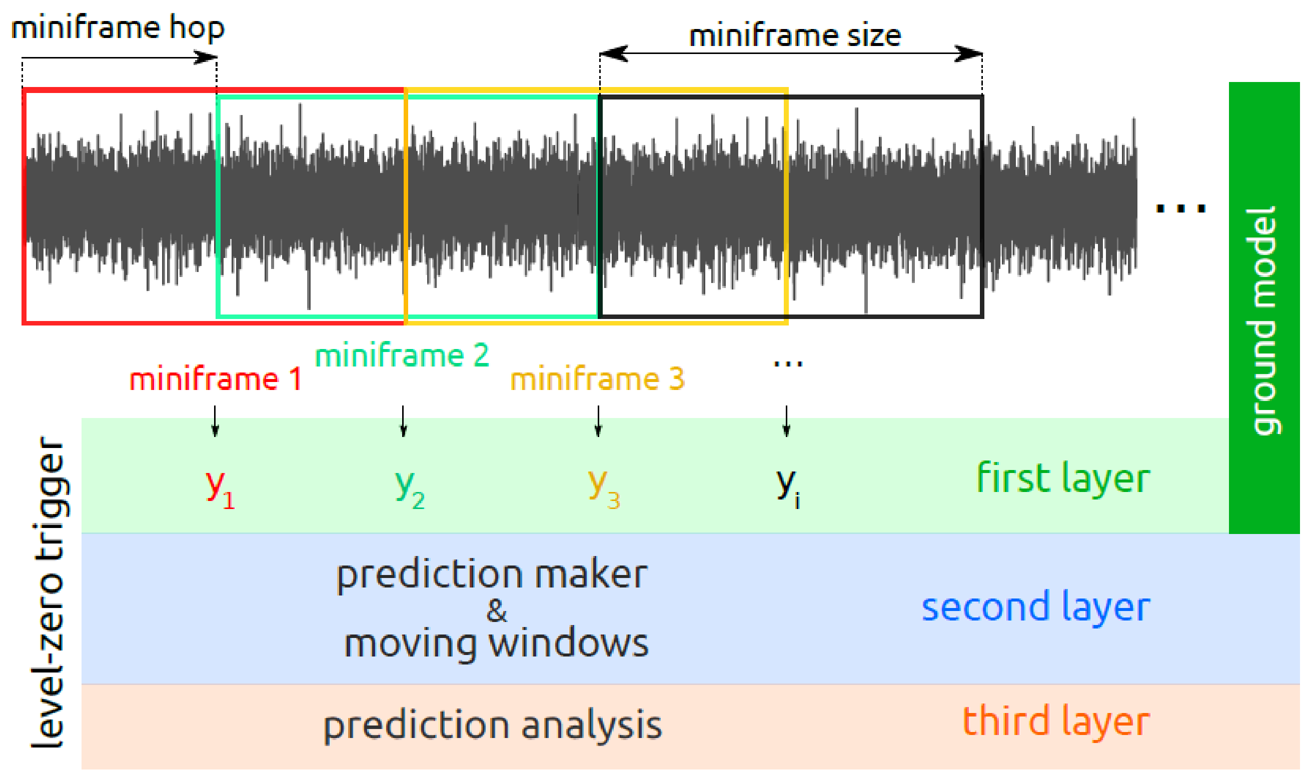
(…)



Gravitational wave signal candidates

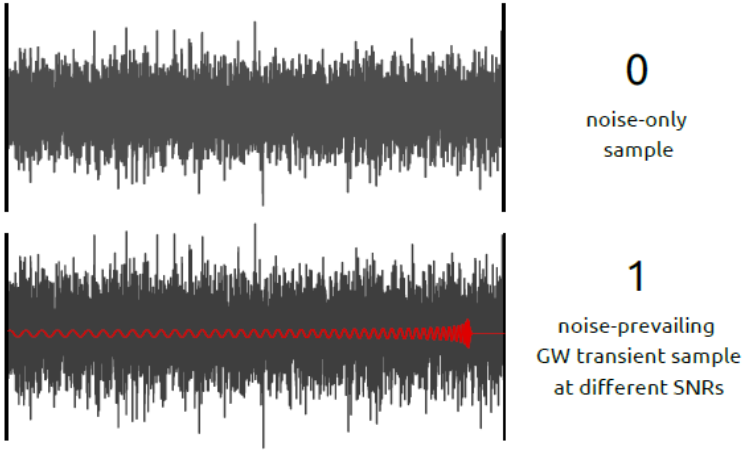
Università degli Studi di Sassari & INFN-LNS

Level-zero trigger: triggers the next levels; extremely low computational complexity.
→ based on the predictions of the ***ground model***.



Ground model derivation

→ using machine learning (*supervised learning*) methods to derive a model to link *patterns* (each pattern is a miniframe in which the stream is subdivided, suitably represented through some *features*) to an output (which is close to "0" if the miniframe does not contain a GW or close to "1" if a GW is present).



Dataset containing examples with 2 possible labels:

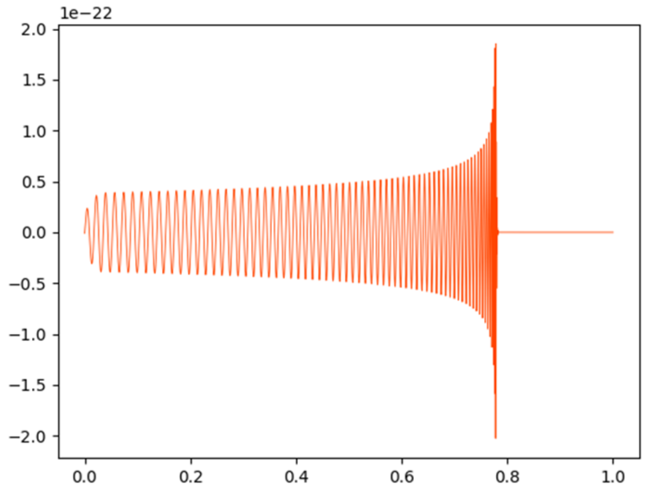
0 → noise only sample

1 → signal + noise

Simulation of a *Binary Black Hole merger* (inspiral+merger+ring down) via state-of-the-art simulation tools (IMRPhenomD-type waveform is adopted).

→ **key parameters**:

- m_1 (from 5 to 100 solar masses);
- $m_2 < m_1$, $m_1 + m_2 < 100$ solar masses;
- m_1 and m_2 follow the canonical logarithmic mass distribution from *B.P. Abbott et al., Phys. Rev. X* ***6*** *(2016) 041015*;
- inclination angle is generated randomly according to a sin distribution;
- phase generated randomly according to a uniform distribution;
- polarisation angle is generated randomly according to a uniform distribution;
- a random sky location is considered (uniformly on the sphere).



example of generated signal as seen by a ground interferometer. 1 second data segment.

LIGO Document T1800044-v5

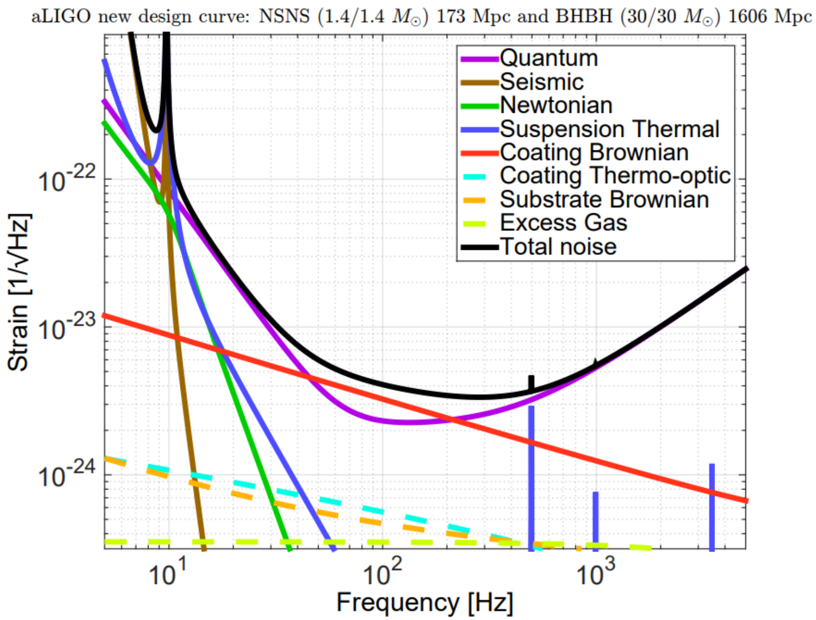
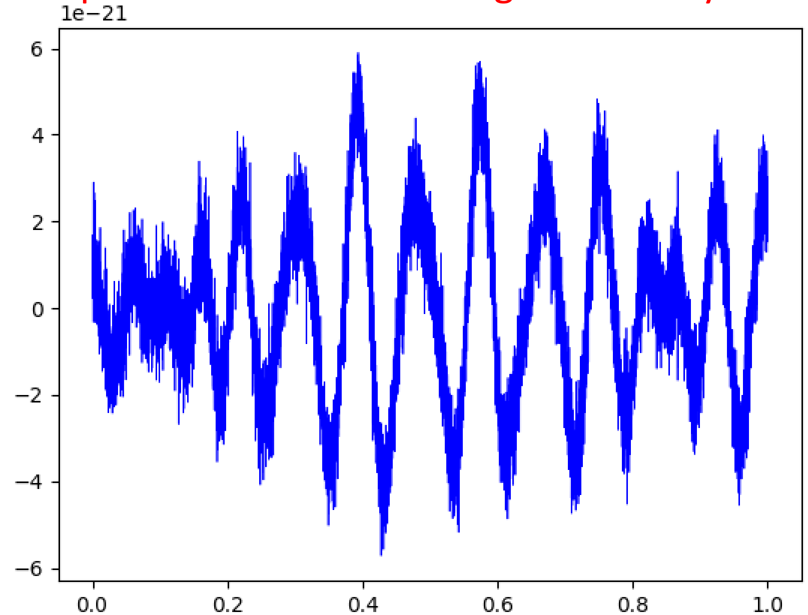
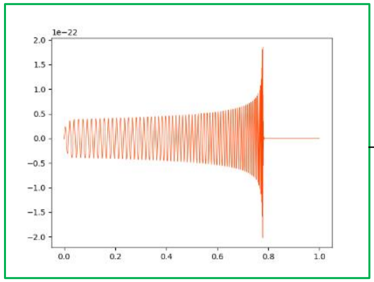


Figure 1: Updated estimate of the Advanced LIGO design curve.

Generated 1 second noise segment from the up-to-date aLIGO H1 design sensitivity.

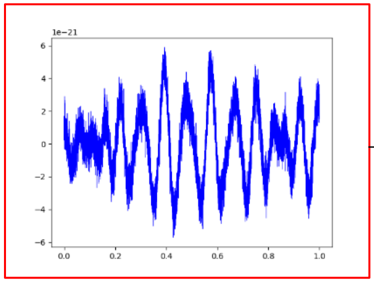


signal



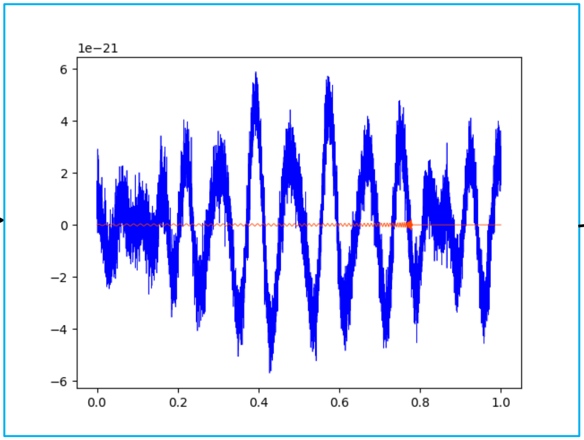
\+

noise

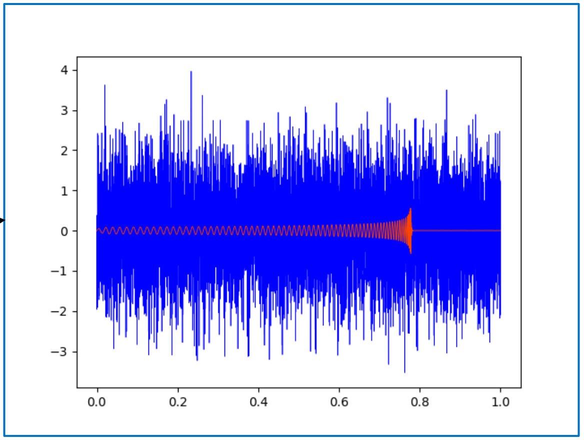


scaled at the desired SNR

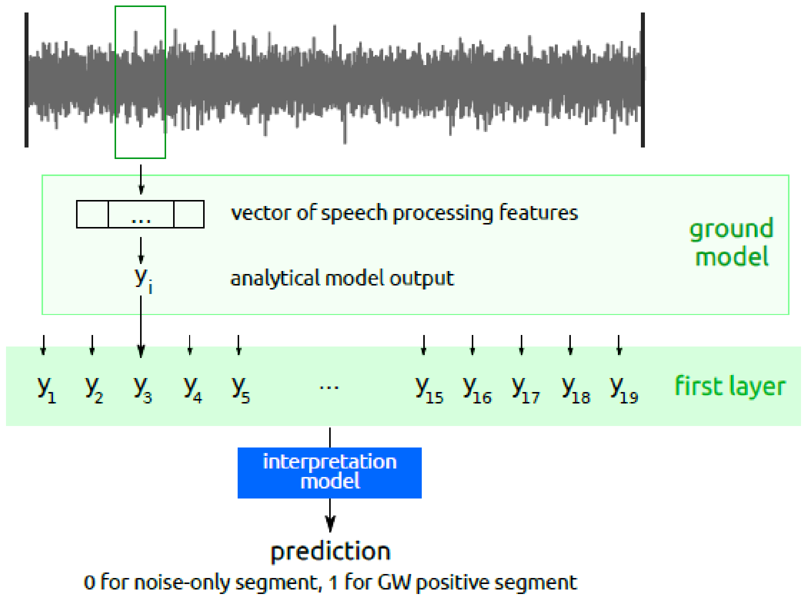
simulation of the strain observed by H1 for a given astrophysical source at a given distance (SNR)



data segment after whitening for the aLIGO design sensitivity curve



Our framework: ground model derivation



features from speech processing (1840 features)

BOB library
- *MFCC*
- *LFCC*
- *RFCC*
- *SCFC*
- *SCMC*

python speech features
- *MFCC*
- *logfbank library*

pyAudioAnalysis
https://github.com/tyiannak/pyAudioAnalysis/wiki/3.-Feature-Extraction

essentia library
 # 0 -> lowLevel.barkbands (vector)
 # 5 -> mfcc (vector)
 # 12 -> spectral_complexity
 # 15 -> spectral_energy
 # 16 -> spectral_energyband_low
 # 17 -> spectral_energyband_middle_low
 # 18 -> spectral_energyband_middle_high
 # 19 -> spectral_energyband_high
 # 20 -> spectral_flatness_db
 # 21 -> spectral_flux
 # 22 -> spectral_rms
 # 23 -> spectral_rolloff
 # 24 -> spectral_strongpeak

Brain Project

A parallel and distributed tool for the formal modeling of data based on neuro-genetic programming

formal model of data to be used as the classification function for each mini-frame

$$y_i = -0.07133 + 0.86274 \cdot pyAA_{31} \cdot \exp\left[\min\left(0.036685, db17_{ess3}\right) - 1.0623 \cdot db17_{ess3} + c\right] + 70.47328 \cdot pyAA_{33}$$

$$\text{where} \quad c = \max\left(0.22473 + \ln\left(db17_{ess3}\right) \cdot \left(db32_{bb4}\right) - 2.6911 \cdot pyAA_{31},\ db17_{ess3}\right)$$

the following 4 features have been selected:

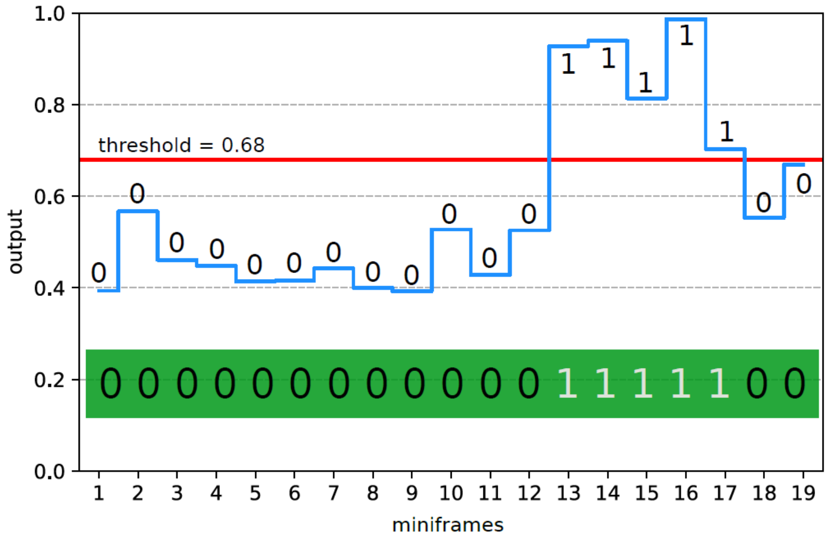
$pyAA_{31}$	The second last element of the *chroma vector*[a] given by the pyAA library.
$pyAA_{33}$	The standard deviation of the 12 elements of the chroma vector given by the pyAA library.
$db17_{ess3}$	The energy of the waveform in band]150,800] Hz given by Essentia library on the db17 pre-processed waveform.
$db32_{bb4}$	The energy of the waveform in band]200,300] Hz given by Essentia library on the db32 pre-processed waveform.

[a] The chroma vector is a 12-element representation of the spectral energy, where the bins represent the 12 equal-tempered pitch classes of music (in the western-type semitone spacing).

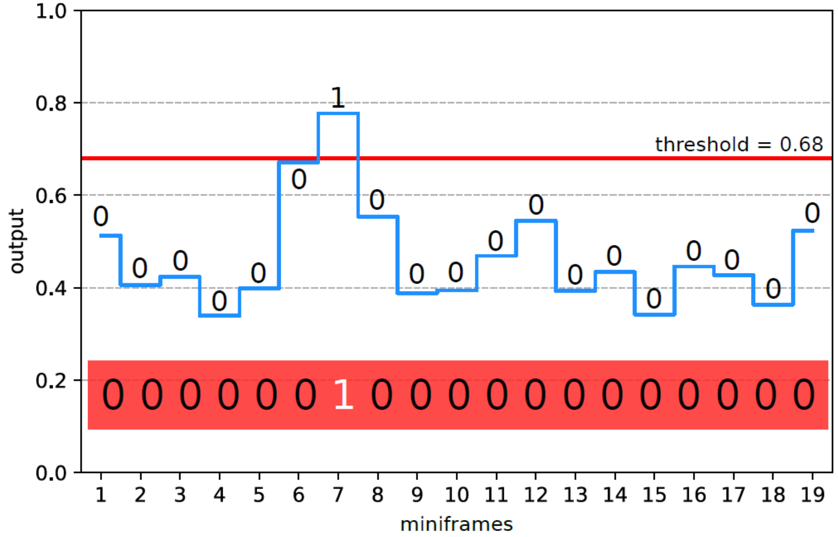
Ground model interpretation

→ thresholded and unthresholded models.

noise+signal example

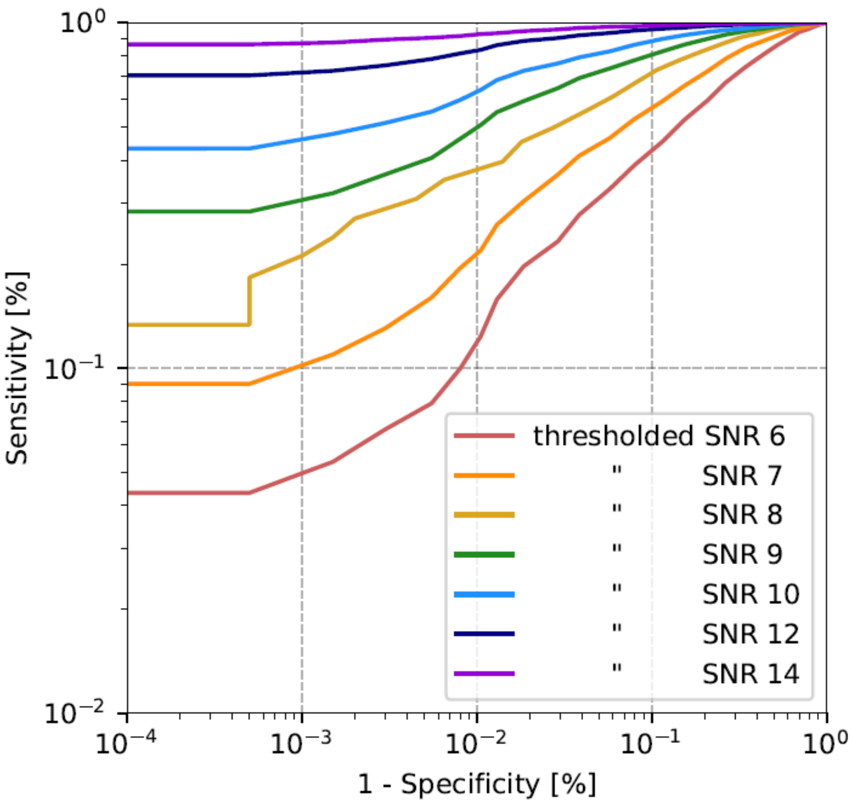


noise only example



Ground model interpretation

→ thresholded and unthresholded models.



sensitivity = *fraction of GW samples correctly identified as GW.*

specificity = *fraction of noise samples correctly identified as noise.*

Ground model interpretation

→ uthresholded model → multi-modeling.

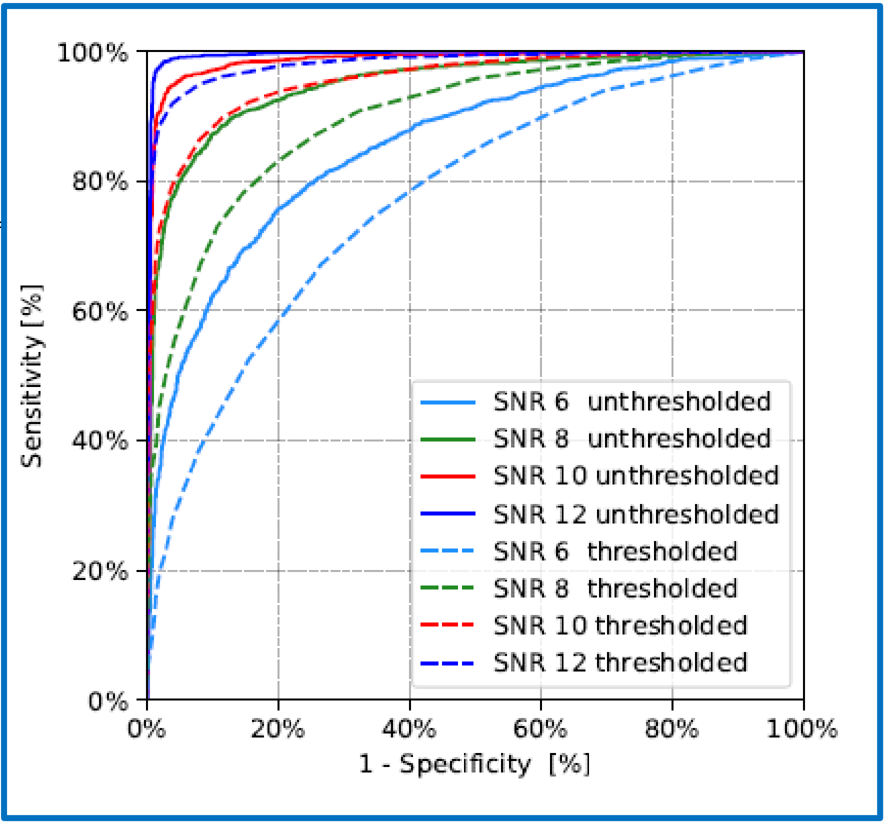
$$Y = \tanh\left\{1.83 \cdot \max\left[\, c_1,\ \max(c_2, b_4)\,\right]\right\}^{19.5073 \cdot \arctan(b_6)}$$

$$\text{where } c_1 = 1.829 \cdot b_4 \cdot \max\left\{1.829 \cdot b_4 \cdot \tan(b_4),\ \ 1.732176 \cdot b_4 \cdot \sin[\max(\tan(b_3), b_4)] + 0.947062 \cdot b_2\right\} + b_2$$

$$\text{and } c_2 = \{1.829 \cdot \sinh[\operatorname{erf}(b_3 \cdot b_1)] \cdot \operatorname{erf}[\tan(b_4)] \cdot b_5\} + b_2$$

the following features have been selected:

b_1	$russo8_3$
b_2	$russo9_5$
b_3	$sm2ampl$
b_4	$sm3weav$
b_5	$sm5barone7_2$
b_6	$sm5barone9_0$



Comparison with CNNs and matched filtering.