Matching Matched-Filtering with Deep Networks for Gravitational wave Astronomy

Chris Messenger

On behalf of
Hunter Gabbard, Michael Williams & Fergus Hayes
Glasgow University

LIGO-G1702164

Forward

- This work has already been presented to the Machine Learning group at the CERN LVC.
- Hands up - I assumed that this constituted an internal LVC presentation.
- Hence we have a paper draft on the DCC
  - https://dcc.ligo.org/LIGO-P1700378
- This is the work of 2 undergraduate students (Fergus Hayes, Michael Williams) and PhD student Hunter Gabbard.

The main aim of the paper

- In early 2017 our work on deep networks was interrupted by evidence that machine learning can produce incredible results.

  - George & Huerta, arXiv:1701.00008 (2017)

- We have since striven to empirically prove that deep networks can reproduce the results of matched-filtering.

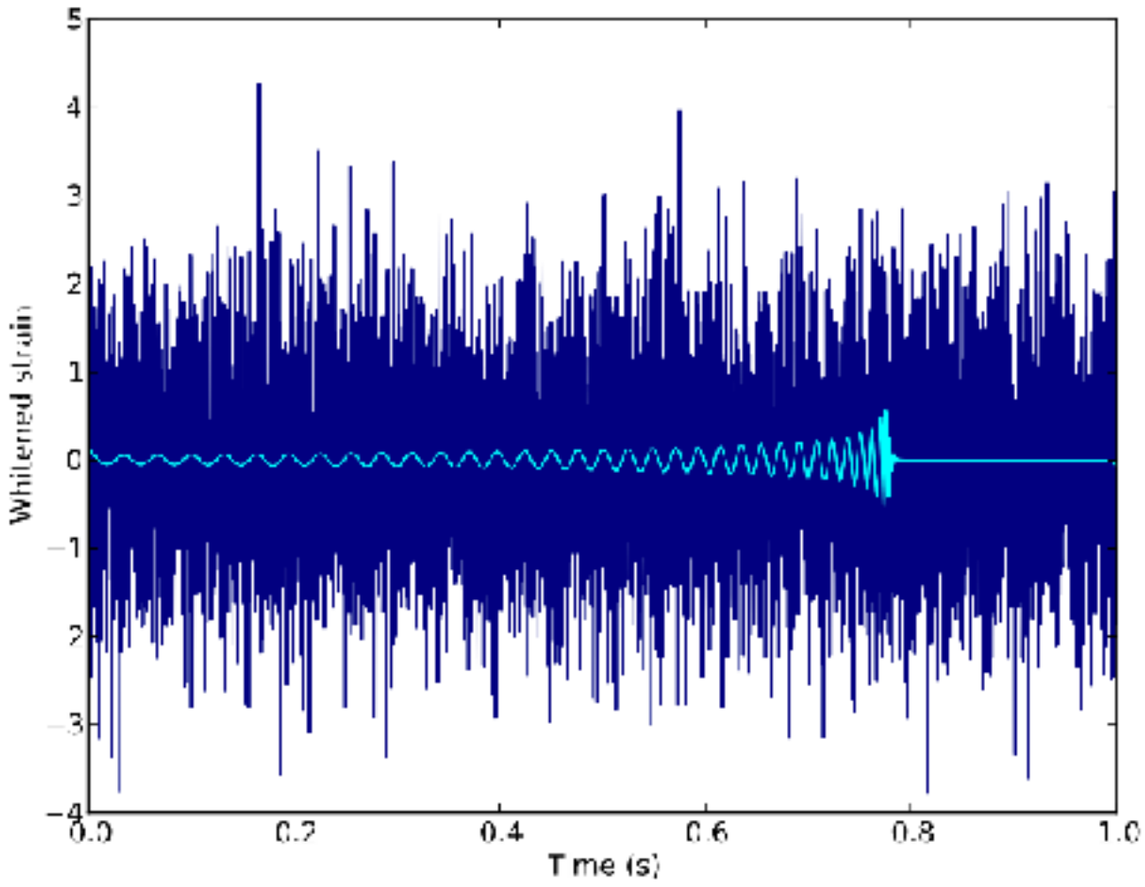
- We therefore show that this is the case for a simple but realistic binary black hole search.

- Finally, why? Speed

Deep Learning - a way to think about it

The analysis overview

- Using BBH time-series in Gaussian noise (whitened).
- 2 classes (signal+noise, noise)
- 60K training data + (12K validation and testing) for each optimal SNR value (2-12)
- Develop a deep network (trial and error)
- Compare with matched filtering (crucial).



The CNN details

Parameter (Option)	Layer 1	2	3	4	5	6	7	8	9	10
Type	C	C	C	C	C	C	C	C	H	H
No. Neurons	8	16	16	32	64	64	128	128	64	2
Filter Size	32	16	16	16	8	8	4	4	n/a	n/a
MaxPool Size	8	n/a	n/a	n/a	6	n/a	n/a	4	n/a	n/a
Drop out	0	0	0	0	0	0	0	0	0.5	0
Act. Func.	Elu	Elu	Elu	Elu	Elu	Elu	Elu	Elu	Elu	SMax

- We tried - data pre-processing, transfer learning, network depth, weight and bias initialisation, activation functions, batch size, batch normalisation, dropout, regularisation, dilation, max pooling, Padding, Stride, gradient descent, …
- Also, input data parameter distributions.

The CNN procedure

- The CNN is trained on waveforms with all parameters randomised (inclination, polarisation, sky, etc…)

    - masses are distributed randomly according to the metric (min mass $5M_{\odot}$, max total mass $100M_{\odot}$)

- Each training signal is used 25 times in different realisations of noise.

- The output statistic is the class probability given by the final SoftMax layer.

- The results are those output from the test data.

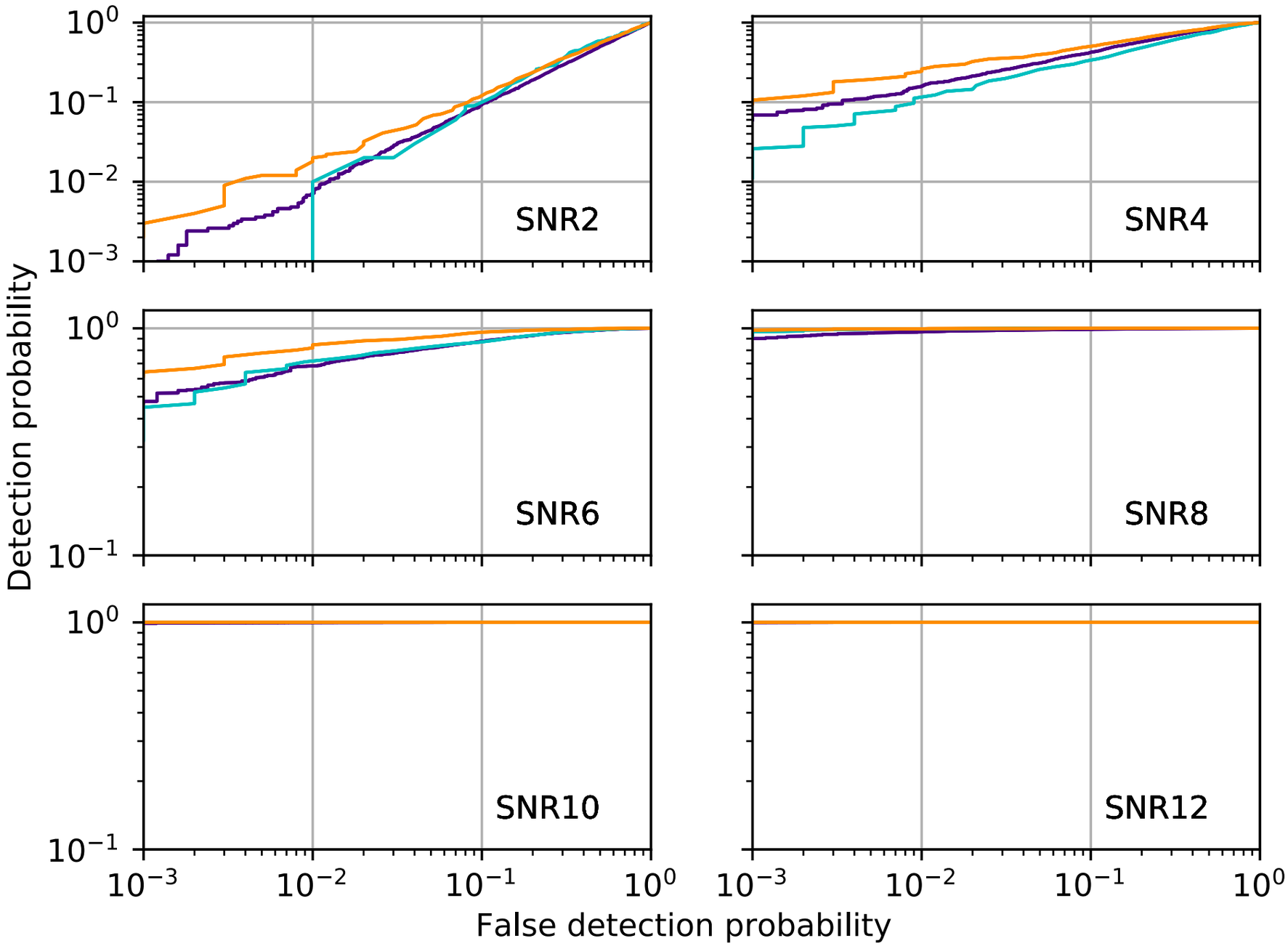
The matched-filtering comparison

- A template bank is constructed using PyCBC with a max mismatch of 3%

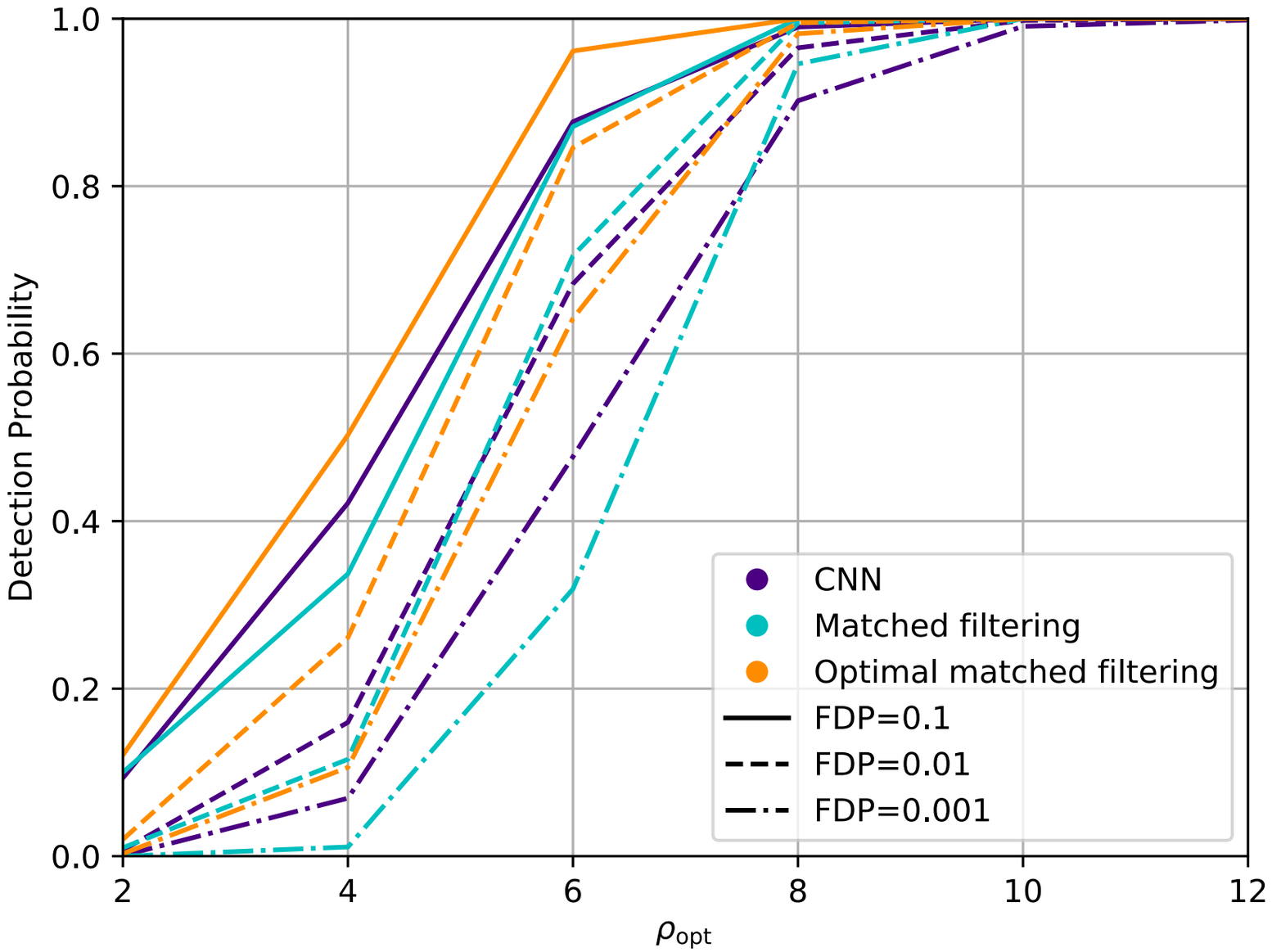
- The phase, amplitude, time and mass maximised measured SNR is recovered using the bank.

- We apply this bank to the same test data as used in the CNN analysis.

The main results - ROC curves



The main results - efficiency



The conclusions

- Deep learning approaches are incredibly powerful.
- We have shown that in a realistic scenario they can achieve matched filtering sensitivities.
- There are many outstanding questions
  - What about non-Gaussianities?
  - Multiple detectors? (this is straight forward)
  - Can it deal with longer waveforms?
  - What about other signals, CW, Burst?

Thank you for your attention