Summary of Last Lecture

$$R(\tau) = \int_{-\infty}^{\infty} df P(f) e^{i2\pi f\tau} \qquad R(\tau) \propto \delta(\tau)$$ **White noise**

$$R(0) = \langle x^2 \rangle = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x^2(t)\,dt = \int_0^{\infty} S(f)\,df$$

$$\left\langle \tilde{x}^*(f')\tilde{x}(f) \right\rangle = P(f)\delta(f - f')$$

Variance $\sigma_i^2 = \sigma^2(f_i) = \left\langle \left| n_i^R \right|^2 \right\rangle = \left\langle \left| n_i^I \right|^2 \right\rangle = \frac{1}{4\Delta f} S_n(f_i)$

$$\gamma\left(f, t, \hat{n}\right) = \sum_A F_{1A}\left(\hat{n}; T\right) F_{2A}\left(\hat{n}; T\right) e^{i2\pi f\left(\hat{n}\bullet\Delta\vec{x}/c\right)}$$

Isotropic Search

Goals of search

- Estimate the energy density of SGWB

$$\Omega_{gw}(f) = \frac{1}{\rho_c}\frac{d\rho_{gw}}{d\ln f} \propto f^{\alpha}$$

- Test the sources of SGWB
- Test GR different polarization modes

Overlap Reduction Function

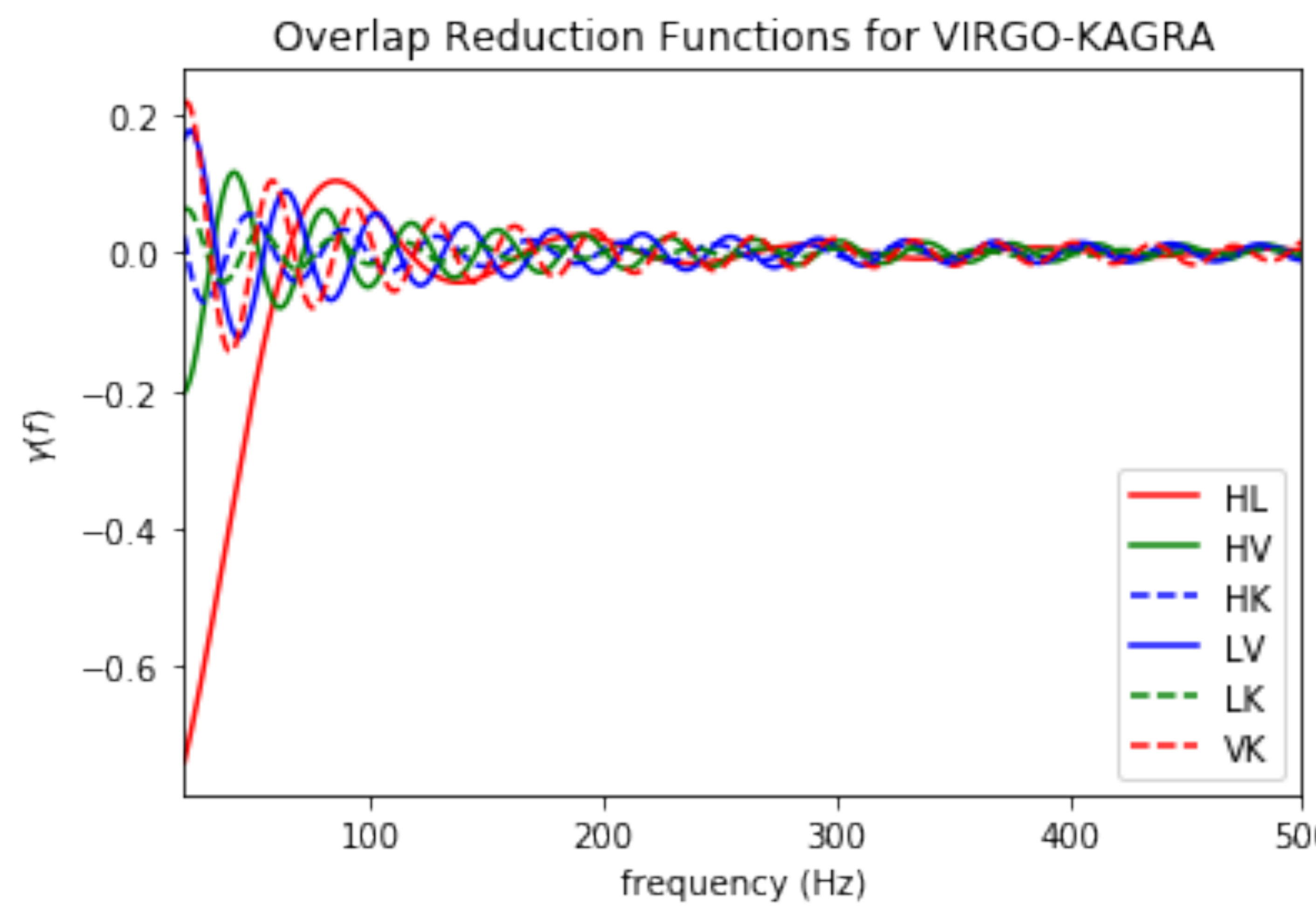
$$\left\langle h_1^*(f;T)h_2(f;T)\right\rangle = \int d^2\hat{n}\sum_A F_{1A}\left(\hat{n};T\right)F_{2A}\left(\hat{n};T\right)\mathbf{P}_A\left(\hat{n},f\right)e^{i2\pi f\left(\hat{n}\cdot\Delta\vec{x}/c\right)}$$

Since the GW is isotropic $\mathbf{P}_A\left(\hat{n},f\right) = H(f)$

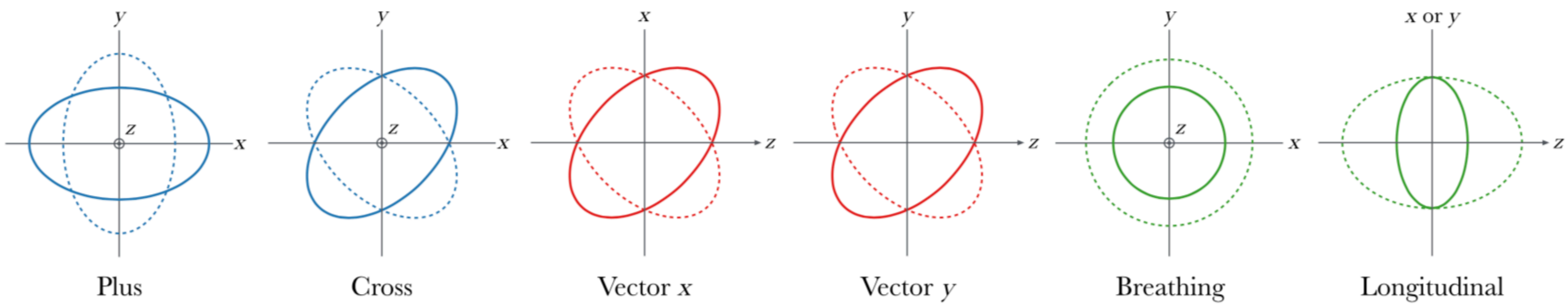
$$\gamma(f) = \frac{5}{8\pi}\int d^2\hat{n}\sum_A F_{1A}\left(\hat{n};T\right)F_{2A}\left(\hat{n};T\right)e^{i2\pi f\left(\hat{n}\cdot\Delta\vec{x}/c\right)}$$

Normalization factor due to

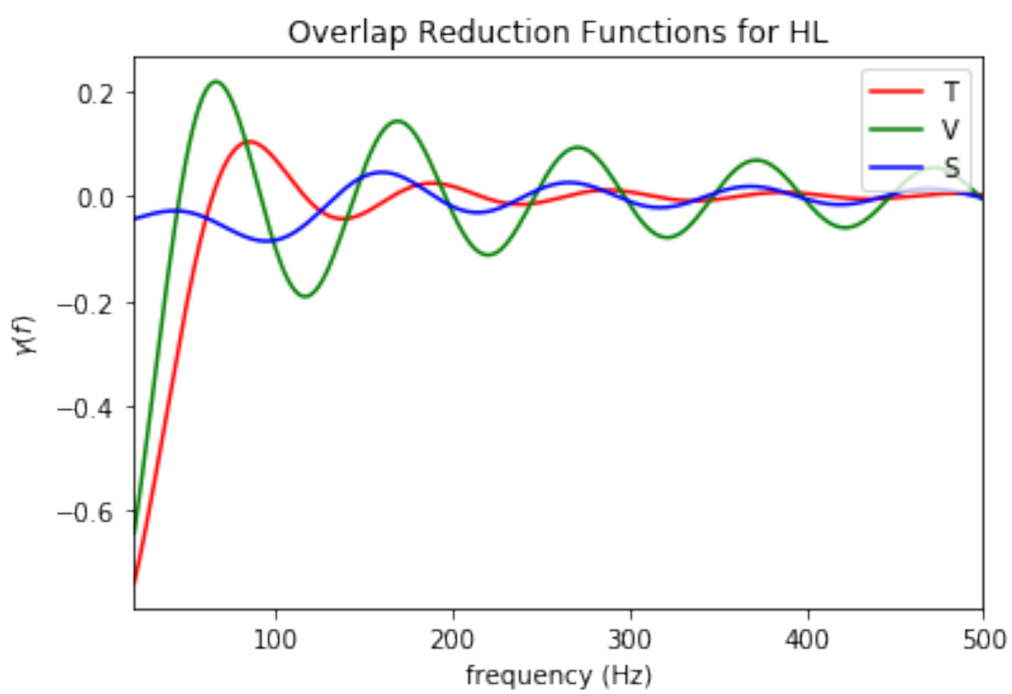
$$\int d^2\hat{n}\left|F_A\left(\hat{n};T\right)\right|^2 = \frac{1}{2}$$

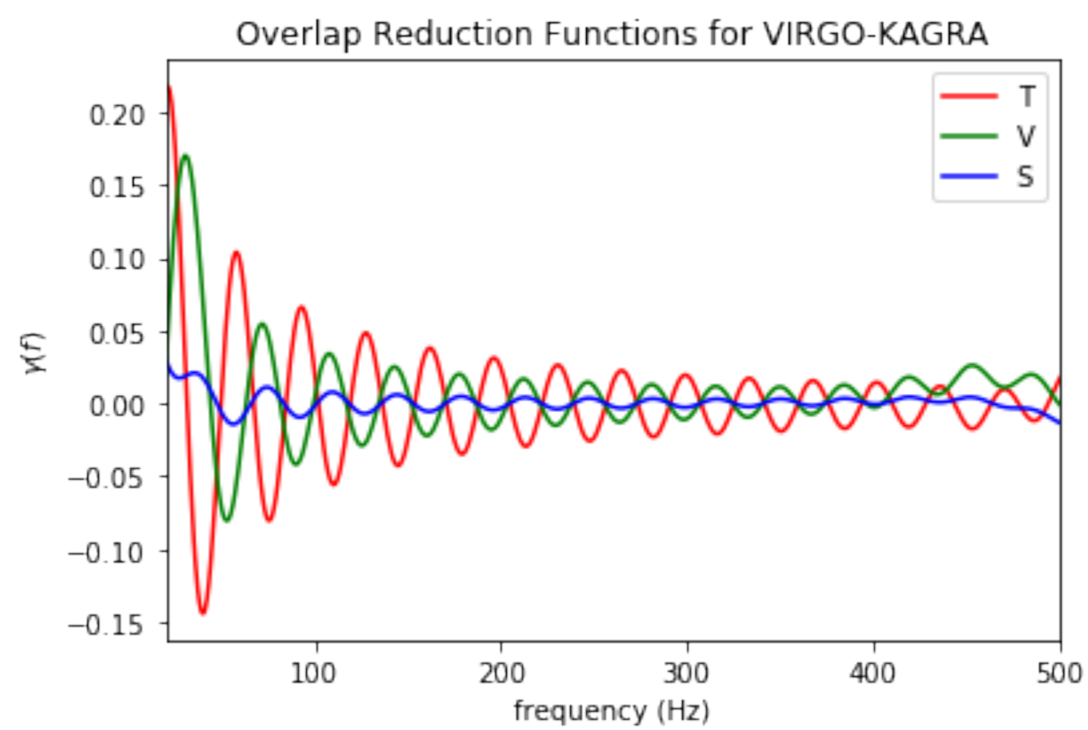


Beyond standard model



Callister et al. 2017





```
from pycbc.detector import Detector
from lal.antenna import AntennaResponse
resp1 = AntennaResponse(detector1, ra[i], dec[i], psi=polarization,
scalar=True, vector=True, times=gps_time)
resp2 = AntennaResponse(detector2, ra[i], dec[i], psi=polarization,
scalar=True, vector=True, times=gps_time)
prod_antenna_T[i] =
(resp1.plus*resp2.plus+resp1.cross*resp2.cross)*2.5
prod_antenna_V[i] = (resp1.x*resp2.x+resp1.y*resp2.y)*2.5
prod_antenna_S[i] = (resp1.x*resp2.b+resp1.y*resp2.l)*2.5
```

Beyond standard model

For the tensor mode

$$\langle C(f) \rangle \propto \gamma_T(f)\Omega_{gw}^T(f)$$

T: +/x

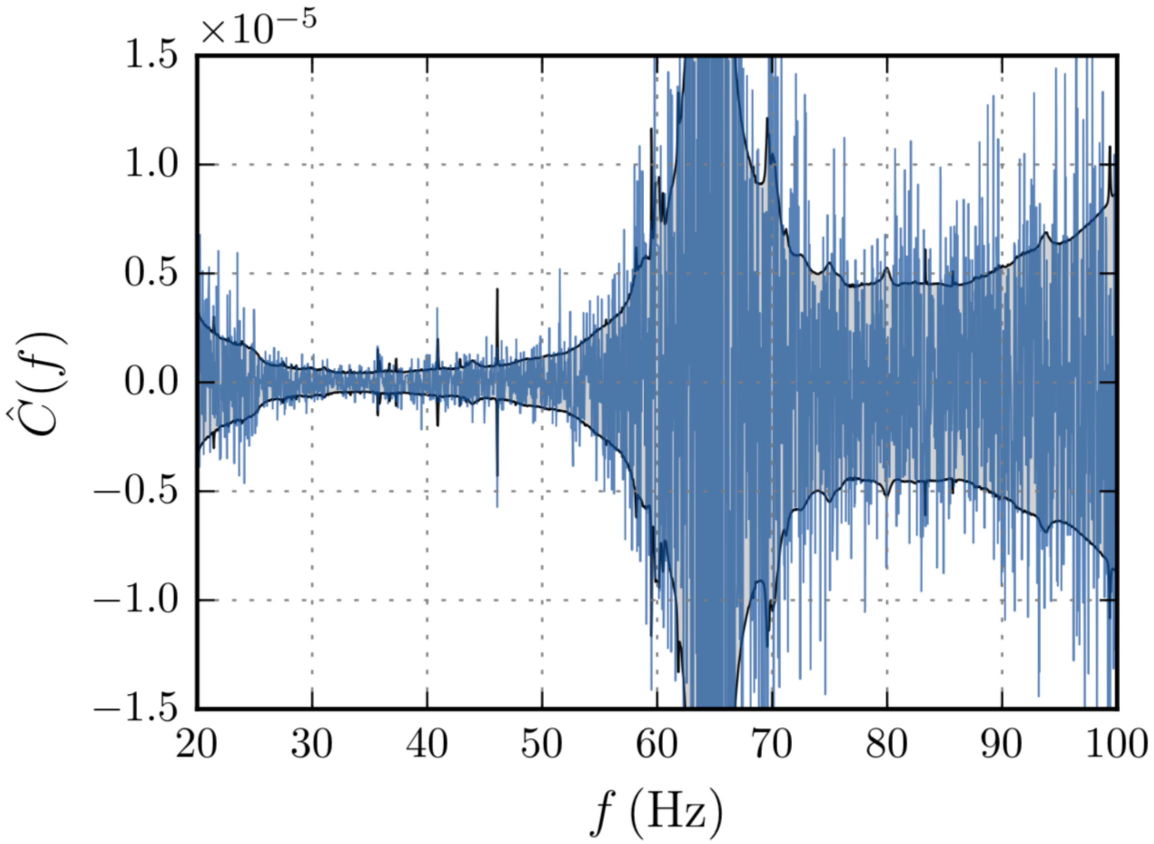
In general

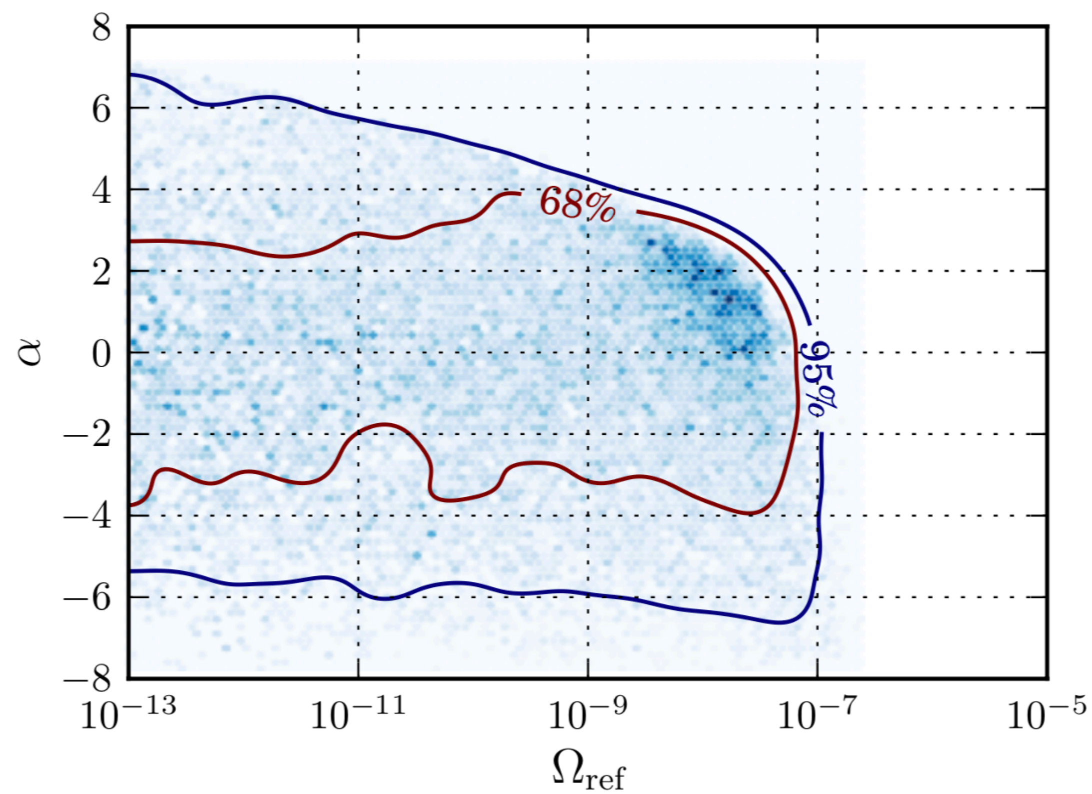
$$\langle C(f) \rangle \propto \sum_A \gamma_A(f)\Omega_{gw}^A(f)$$

LV Isotropic Search with O1+O2

$$\widehat{C}(f) = \frac{2}{T}\frac{\mathrm{Re}[\tilde{d}_1^* \tilde{d}_2]}{\gamma_T(f) S_0(f)},$$

$$S_0(f) = \frac{3H_0^2}{10\pi^2 f^3}$$





LV Scientific Collaboration 2019

Matched Filter

Gaussian Noise

Suppose in Fourier domain, we have a set of noise $\{\tilde{n}(f_i)\}$ follow the Gaussian distribution with $\{\sigma(f_i)\}$ and fi is uniformly sampled with interval Δf .

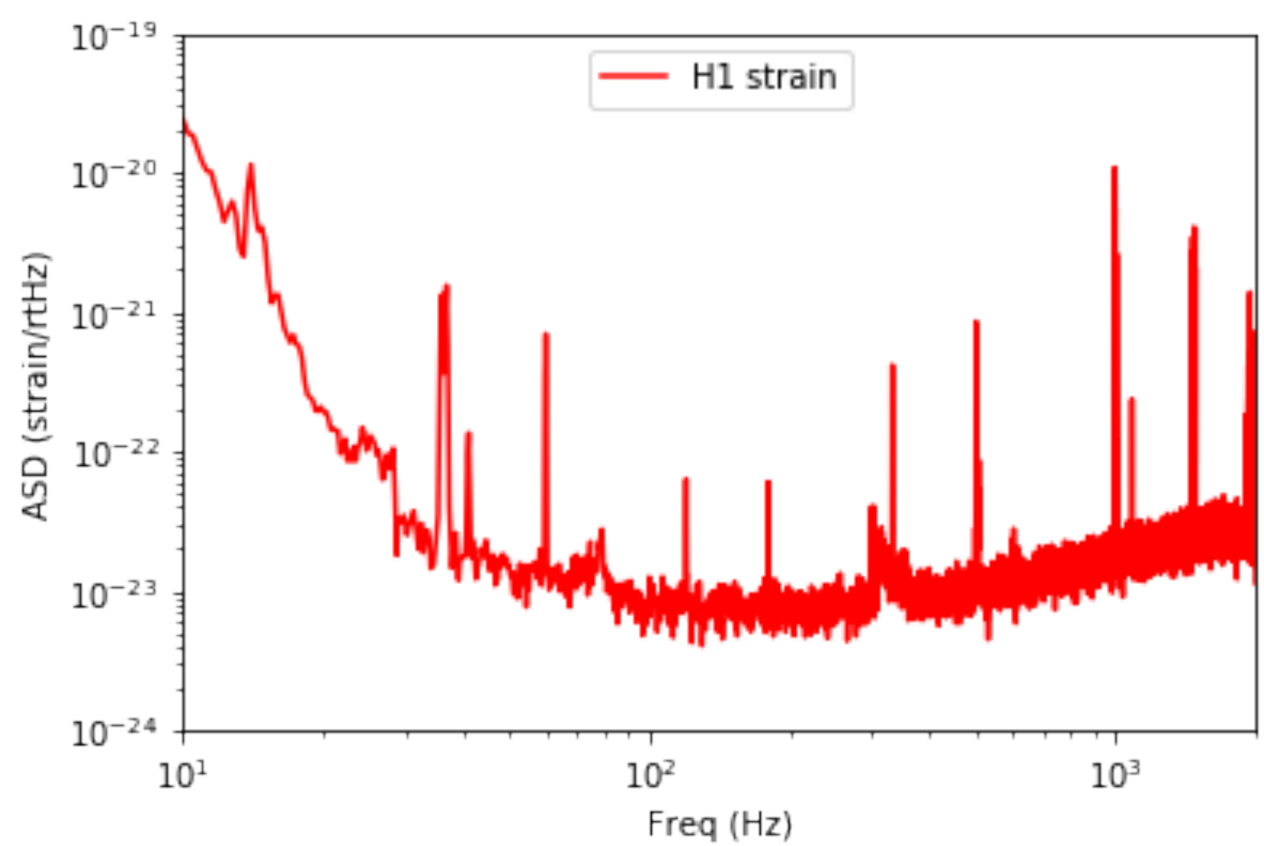
Joint probability

$$p\left(\{\tilde{n}\}\right)=\left(\frac{1}{\sqrt{2\pi}}\right)^N \frac{1}{\sigma_0\sigma_1^2\sigma_2^2....\sigma_{N/2}} \exp\left[-\frac{1}{2}\sum_k \frac{\left|\tilde{n}_k^2\right|}{\sigma_k^2}\right]$$

exponent parts

$$-\frac{1}{2}\sum_k \frac{\left|\tilde{n}_k^2\right|}{\sigma_k^2} = -\frac{1}{2}4\sum_k \frac{\left|\tilde{n}_k^2\right|}{S_n\left(\left|f_k\right|\right)}\Delta f$$

$$\sim \frac{1}{2}4\int_0^\infty \frac{\left|\tilde{n}(f)\right|^2}{S_n\left(\left|f_k\right|\right)}df = -\frac{1}{2}2\int_{-\infty}^\infty \frac{\left|\tilde{n}(f)\right|^2}{S_n\left(\left|f_k\right|\right)}df$$



Gaussian Noise

Define the inner product

$$(n,n) \equiv 2\int_{-\infty}^{\infty} \frac{\tilde{n}(f)\tilde{n}^*(f)}{S_n(|f_k|)}\,df$$

Joint probability density of noise

$$p \propto e^{-\frac{1}{2}(n,n)}$$

We can also define

$$(a,b) \equiv 2\int_{-\infty}^{\infty} \frac{\tilde{a}(f)\tilde{b}^*(f)}{S_n(|f_k|)}\,df$$

Likelihood ratio

Suppose we have an observed data d, how do we know whether there is gw or not?

Hypothesis 0 : d is pure noise $d = n$

Hypothesis 1: d contains gw $d = h + n$

Likelihood ratio

$$\Lambda = \frac{p(d|H_1)}{p(d|H_0)}$$

If the noise is gaussian, the probability densities under the Ho is

$$p(d|H_0) = Ce^{-\frac{1}{2}(d,d)}$$

Under the hypothesis H_1 and assume we know h

$$p(d|H_1) = p(d-h|H_0) = Ce^{-\frac{1}{2}(d-h,d-h)} = Ce^{-\frac{1}{2}(d,d)+(d,h)-\frac{1}{2}(h,h)}$$

Likelihood ratio will be

$$\Lambda = \frac{p(d|H_1)}{p(d|H_0)} = e^{(d,h)-\frac{1}{2}(h,h)}$$

Likelihood ratio

We want to find the maximum value of lambda

$$\lambda = \ln \Lambda = (d,h) - \frac{1}{2}(h,h)$$

rescale $h = a\hat{h}$ **so that** $(\hat{h},\hat{h}) = 1$ **normalized template**

$$\lambda = \ln \Lambda = a(d,\hat{h}) - \frac{1}{2}a^2(\hat{h},\hat{h})$$

This is maximized for an amplitude a where

$$a_{\max} = \frac{(d,\hat{h})}{(\hat{h},\hat{h})} = (d,\hat{h})$$

and

$$\lambda_{\max} = \frac{1}{2}(d,\hat{h})^2$$

Matched filter: noise-weighted correlation of the anticipated signal with data

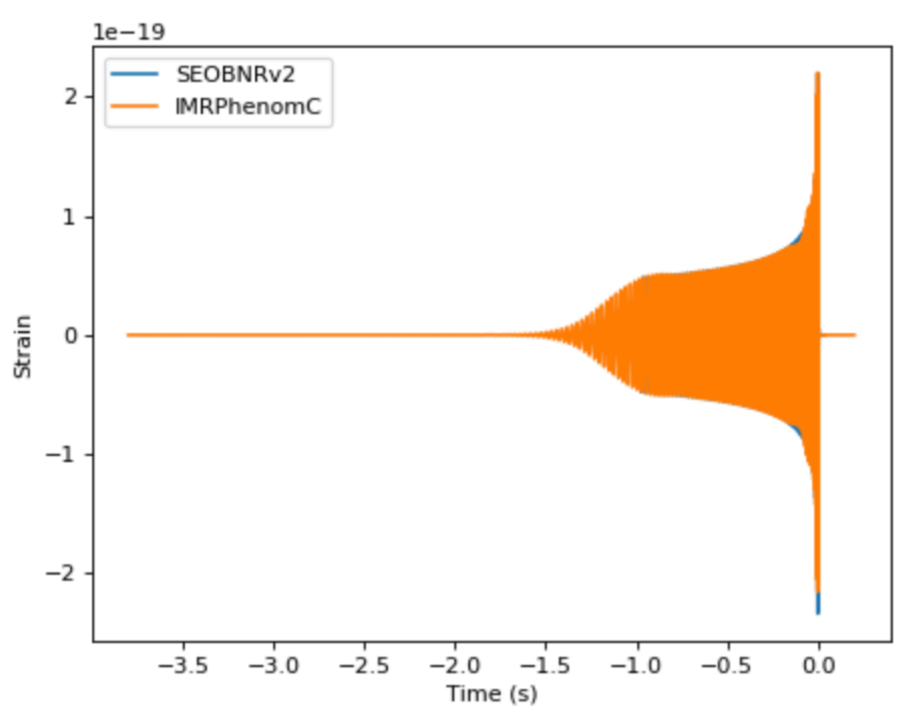
$$\rho = (d,\hat{h})$$

Unknown arrival time

Suppose we have a signal with known shape but an unknown arrival time.

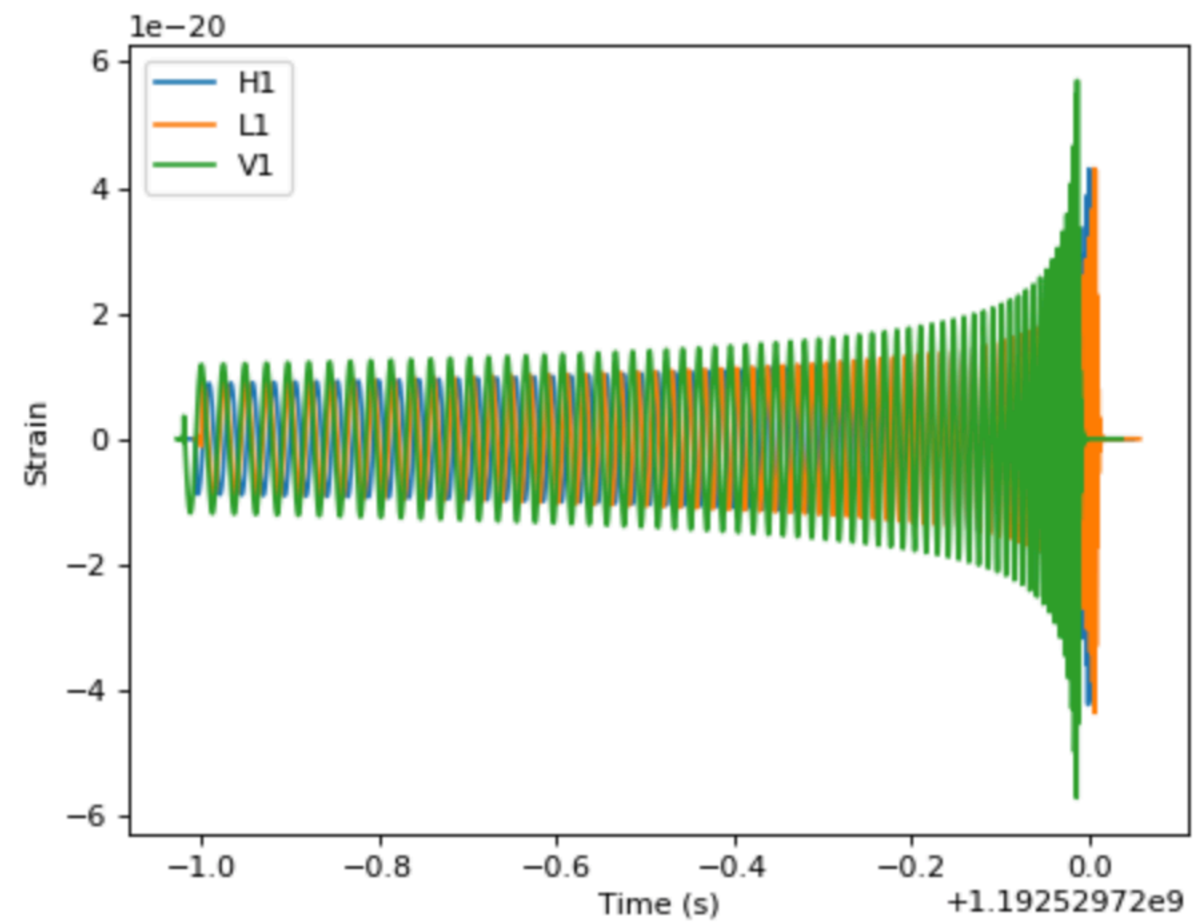
```
import pylab
from pycbc.waveform import get_td_waveform

for apx in ['SEOBNRv2', 'IMRPhenomC']:
    hp, hc = get_td_waveform(approximant=apx,
                             mass1=10,
                             mass2=10,
                             spin1z=0.9,
                             delta_t=1.0/4096,
                             f_lower=40)

    pylab.plot(hp.sample_times, hp, label=apx)

pylab.ylabel('Strain')
pylab.xlabel('Time (s)')
pylab.legend()
pylab.show()
```

(Source code, png, hires.png, pdf)



https://pycbc.org/pycbc/latest/html/waveform.html

```python
import pylab
from pycbc.waveform import get_td_waveform
from pycbc.detector import Detector

apx = 'SEOBNRv4'
# NOTE: Inclination runs from 0 to pi, with poles at 0 and pi
#       coa_phase runs from 0 to 2 pi.
hp, hc = get_td_waveform(approximant=apx,
                         mass1=10,
                         mass2=10,
                         spin1z=0.9,
                         spin2z=0.4,
                         inclination=1.23,
                         coa_phase=2.45,
                         delta_t=1.0/4096,
                         f_lower=40)

det_h1 = Detector('H1')
det_l1 = Detector('L1')
det_v1 = Detector('V1')

# Choose a GPS end time, sky location, and polarization phase for the merger
# NOTE: Right ascension and polarization phase runs from 0 to 2pi
#       Declination runs from pi/2. to -pi/2 with the poles at pi/2. and -pi/2.
end_time = 1192529720
declination = 0.65
right_ascension = 4.67
polarization = 2.34
hp.start_time += end_time
hc.start_time += end_time

signal_h1 = det_h1.project_wave(hp, hc,  right_ascension, declination, polarization)
signal_l1 = det_l1.project_wave(hp, hc,  right_ascension, declination, polarization)
signal_v1 = det_v1.project_wave(hp, hc,  right_ascension, declination, polarization)

pylab.plot(signal_h1.sample_times, signal_h1, label='H1')
pylab.plot(signal_l1.sample_times, signal_l1, label='L1')
pylab.plot(signal_v1.sample_times, signal_v1, label='V1')

pylab.ylabel('Strain')
pylab.xlabel('Time (s)')
pylab.legend()
pylab.show()
```



https://pycbc.org/pycbc/latest/html/waveform.html

Unknown arrival time

Signal's arrival time t_0

$$\hat{h}(t) = \hat{h}'(t - t_0)$$

In Foruier domain

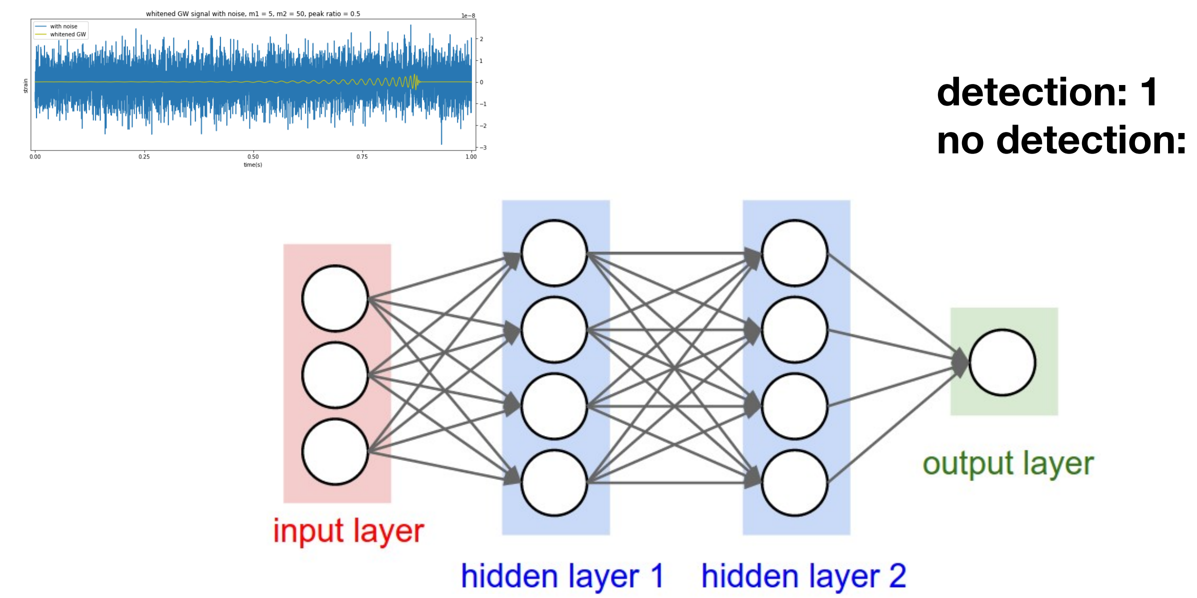
$$\tilde{\hat{h}}(f) = \int \hat{h}(t) e^{-i2\pi ft}\, dt = \int \tilde{\hat{h}}'(t - t_0) e^{-i2\pi ft}\, dt = e^{i2\pi f t_0} \int \tilde{\hat{h}}'(t') e^{-i2\pi ft'}\, dt' = \tilde{\hat{h}}'(f) e^{-i2\pi i f t_0}$$

Matched filter

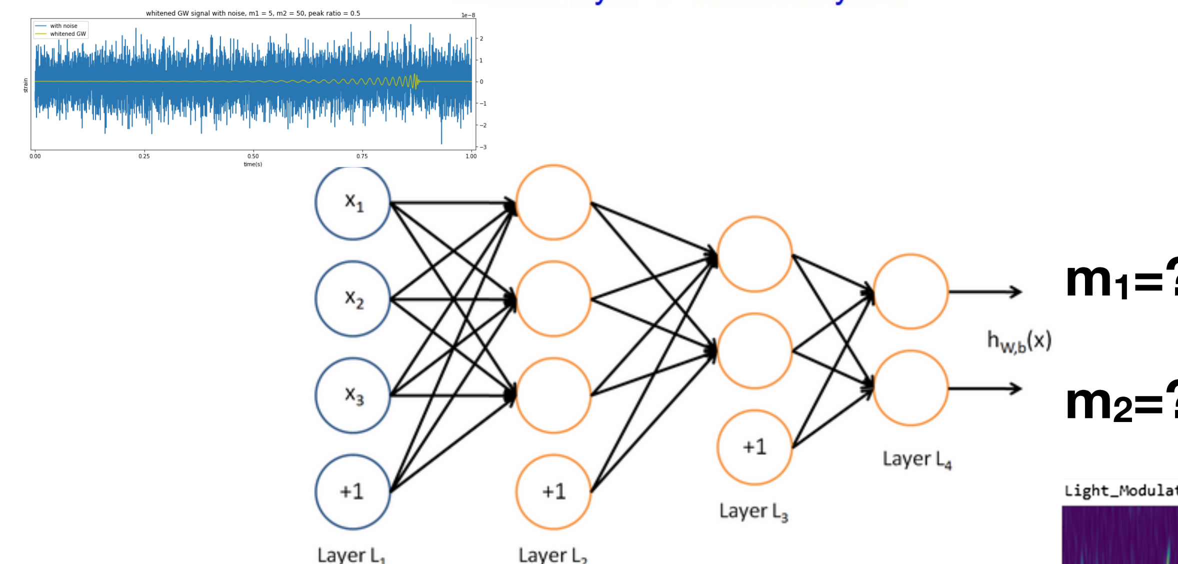
$$\rho = \left(d, \hat{h}\right)(t_0) = 2\int_{-\infty}^{\infty} \frac{\tilde{d}(f)\tilde{\hat{h}}'^{*}(f)}{S_n\left(\left|f_k\right|\right)} e^{i2\pi f t_0}\, df$$

Deep Learning on GW

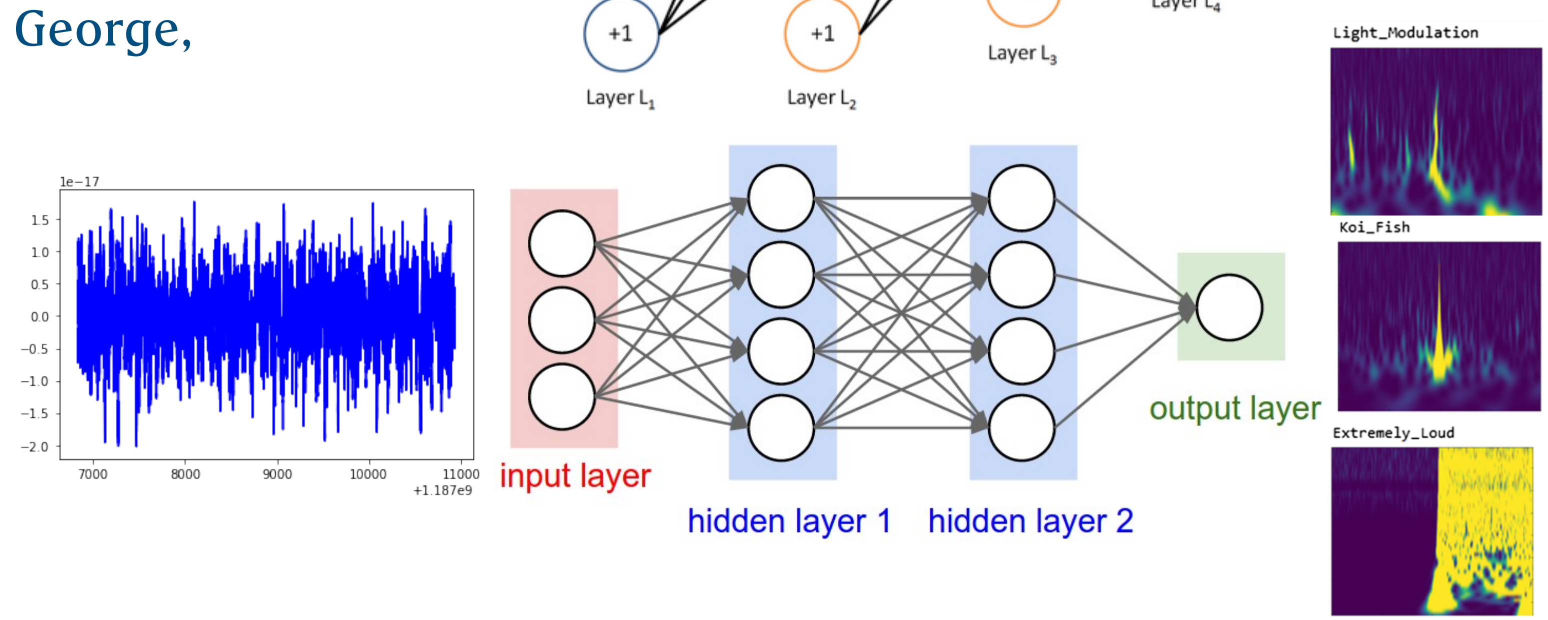
- Search of gravitational signals
  George, Shen & Huerta 2018; Gabbard et al. 2018



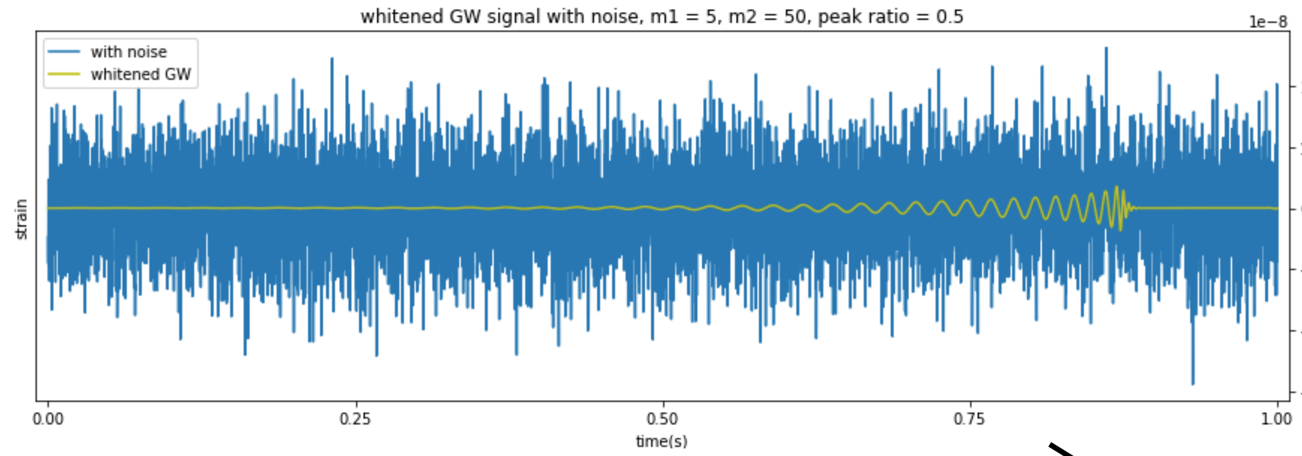
- Parameter estimation
  George, Shen & Huerta 2017, Gabbard et al. 2019, Chatterjee et al. 2019

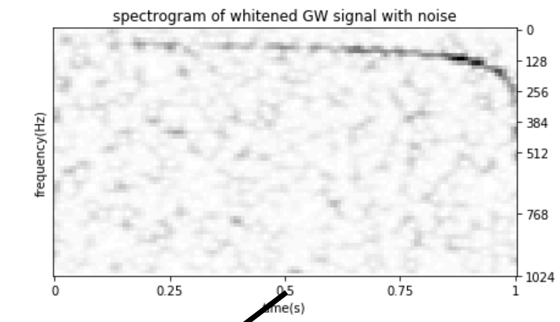


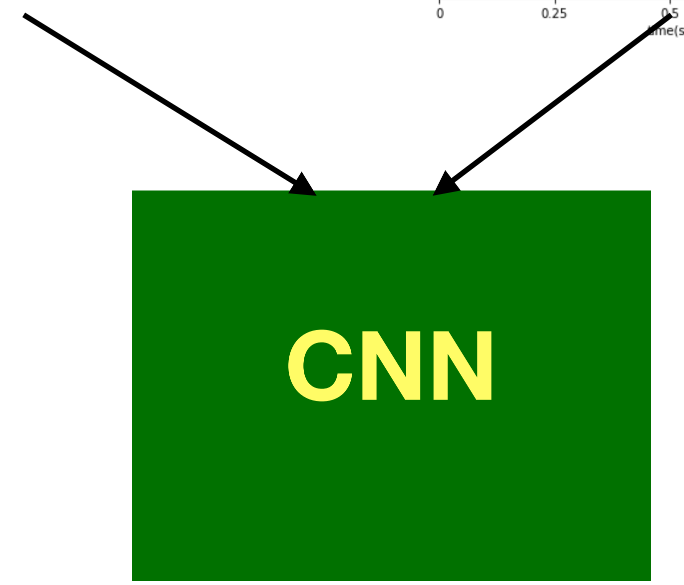
- Transient study
  Mukund et al. 2017; Zevin et al. 2017; George, Shen & Huerta 2018



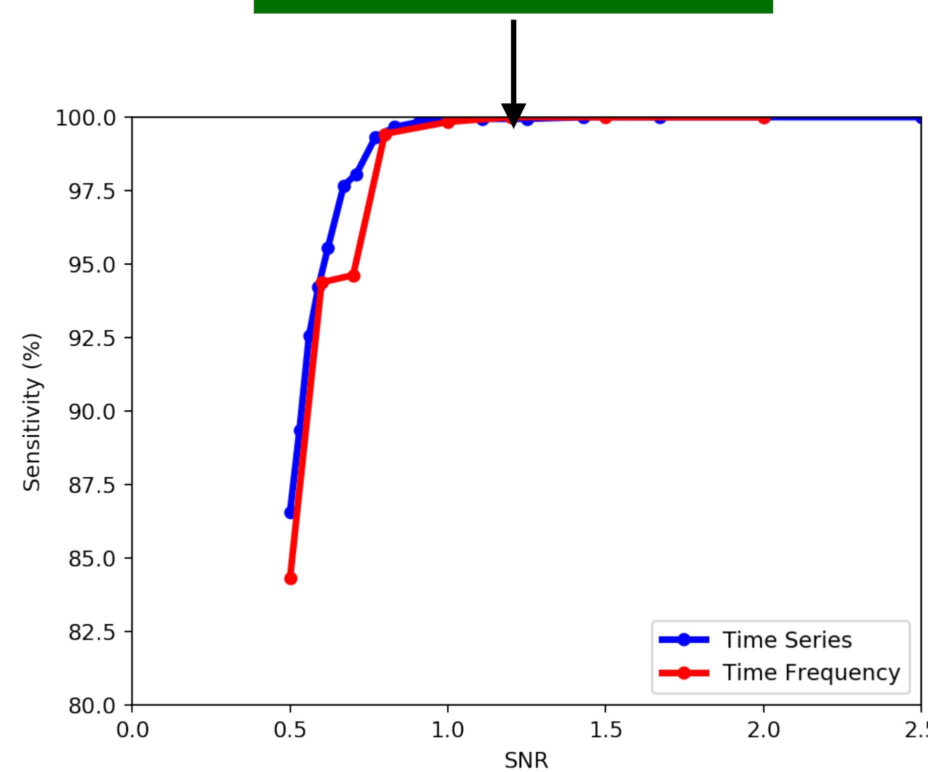
CNN for Detection Test



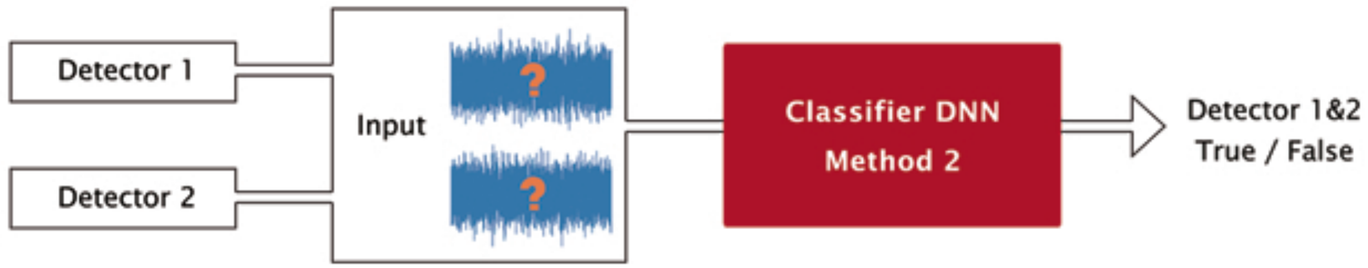


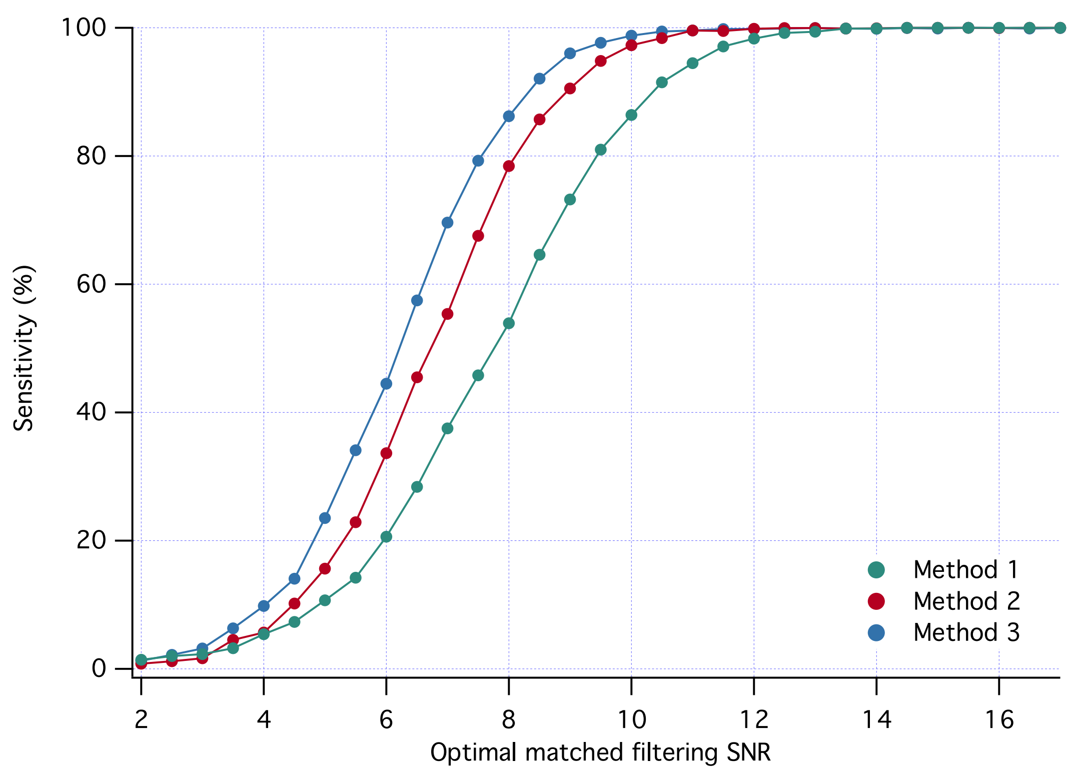


- Training part:
  - 1 second data with 8192 sampling rate
  - whiten Signal(by EOBNRv2) + white noise 9462
  - Pure white noise 8000
  - Mass range: 5-75 Ms in steps of 0.5Ms, mass ratio<10
- Testing part:
  - 3492 test data (S+N/N)
  - Running time about a few seconds (accelerated by K80)



Multi-detector

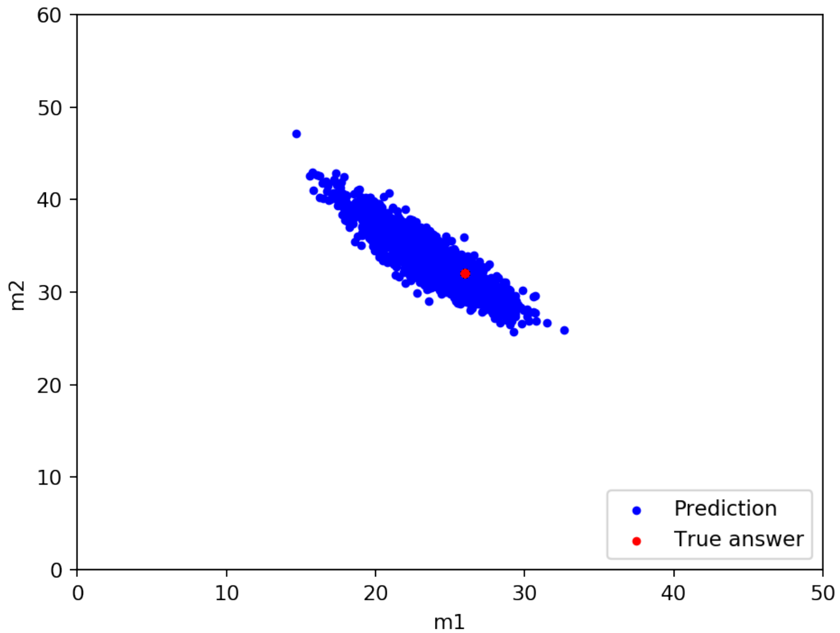


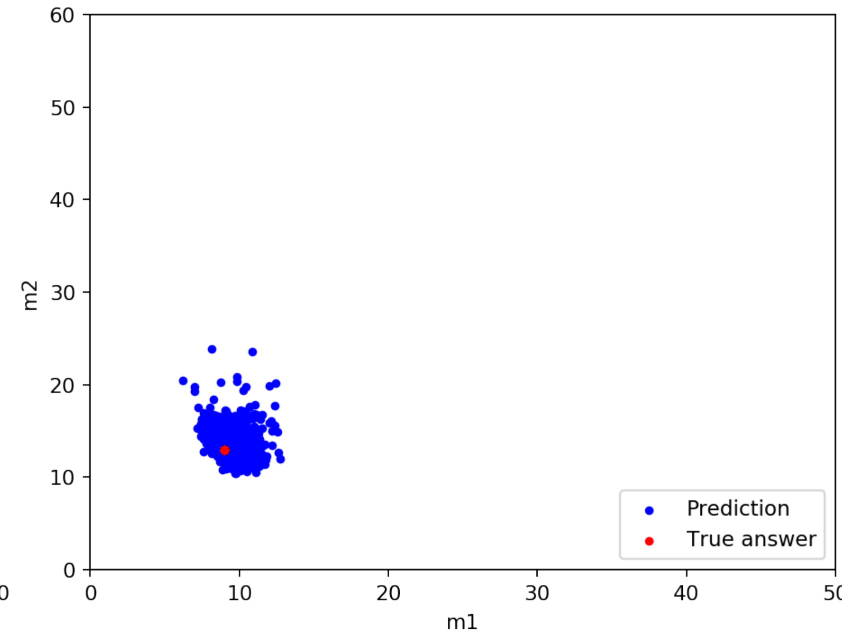


Results for mass estimation

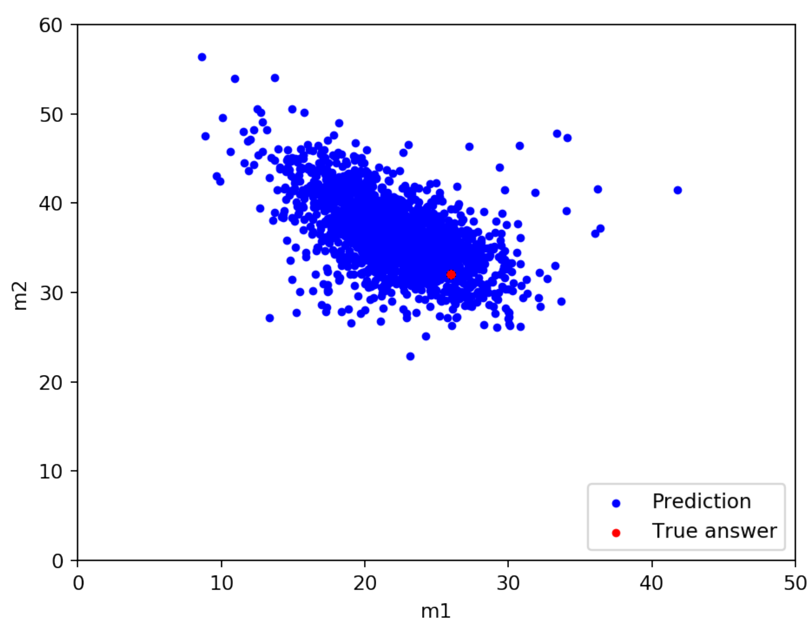
input (32,26)   input (9,13)

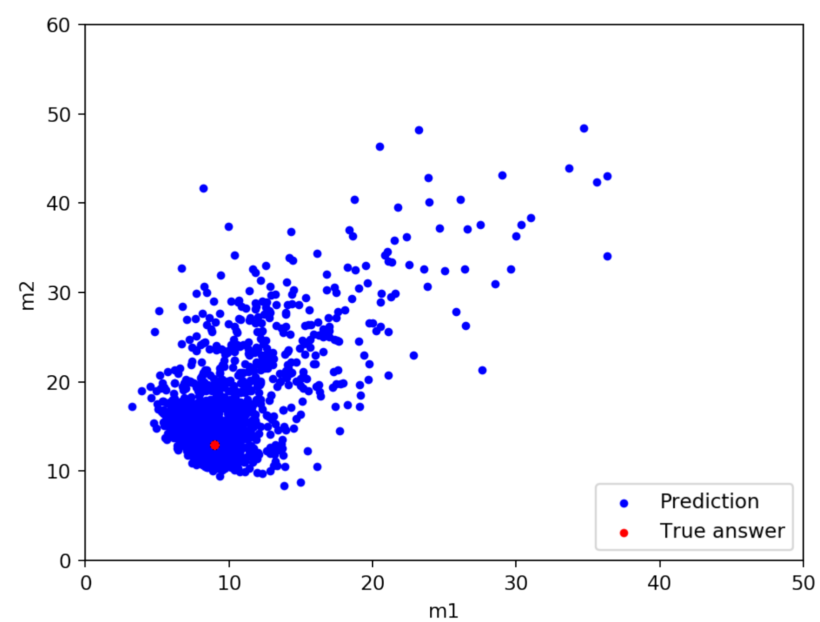
SNR=2

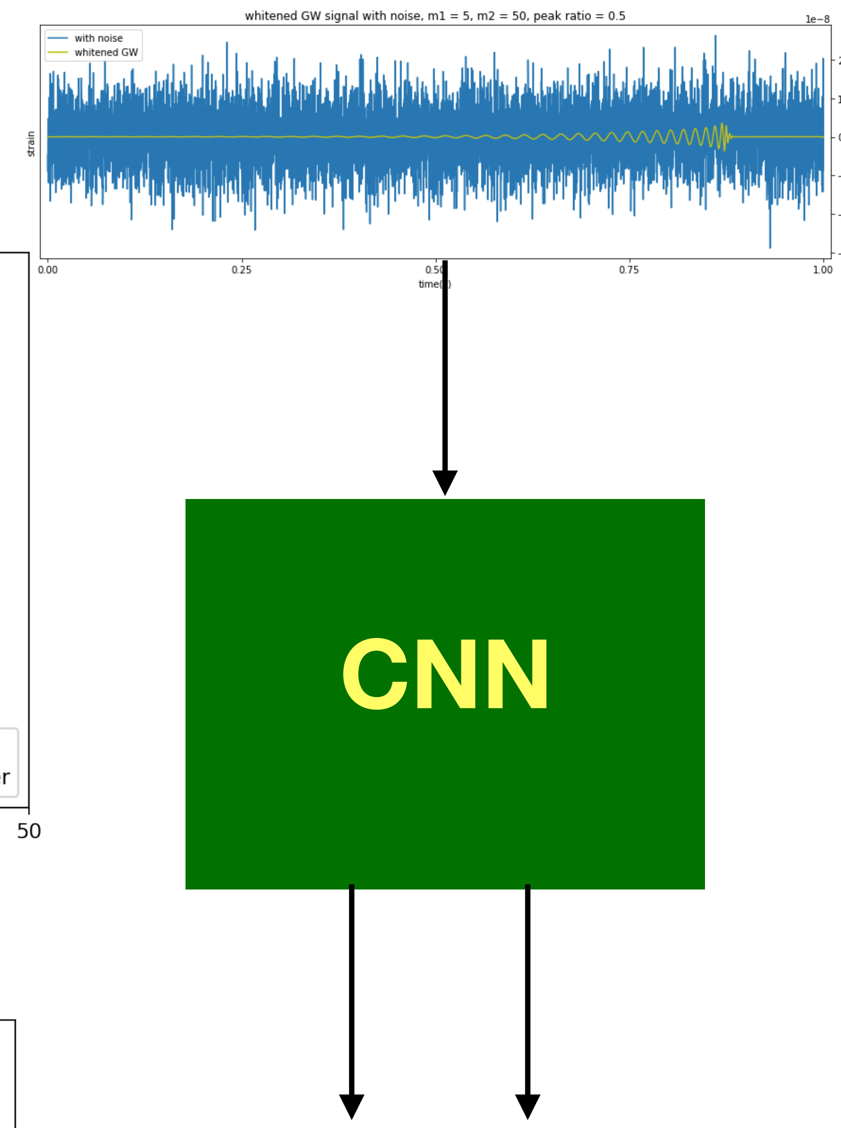




SNR=0.83

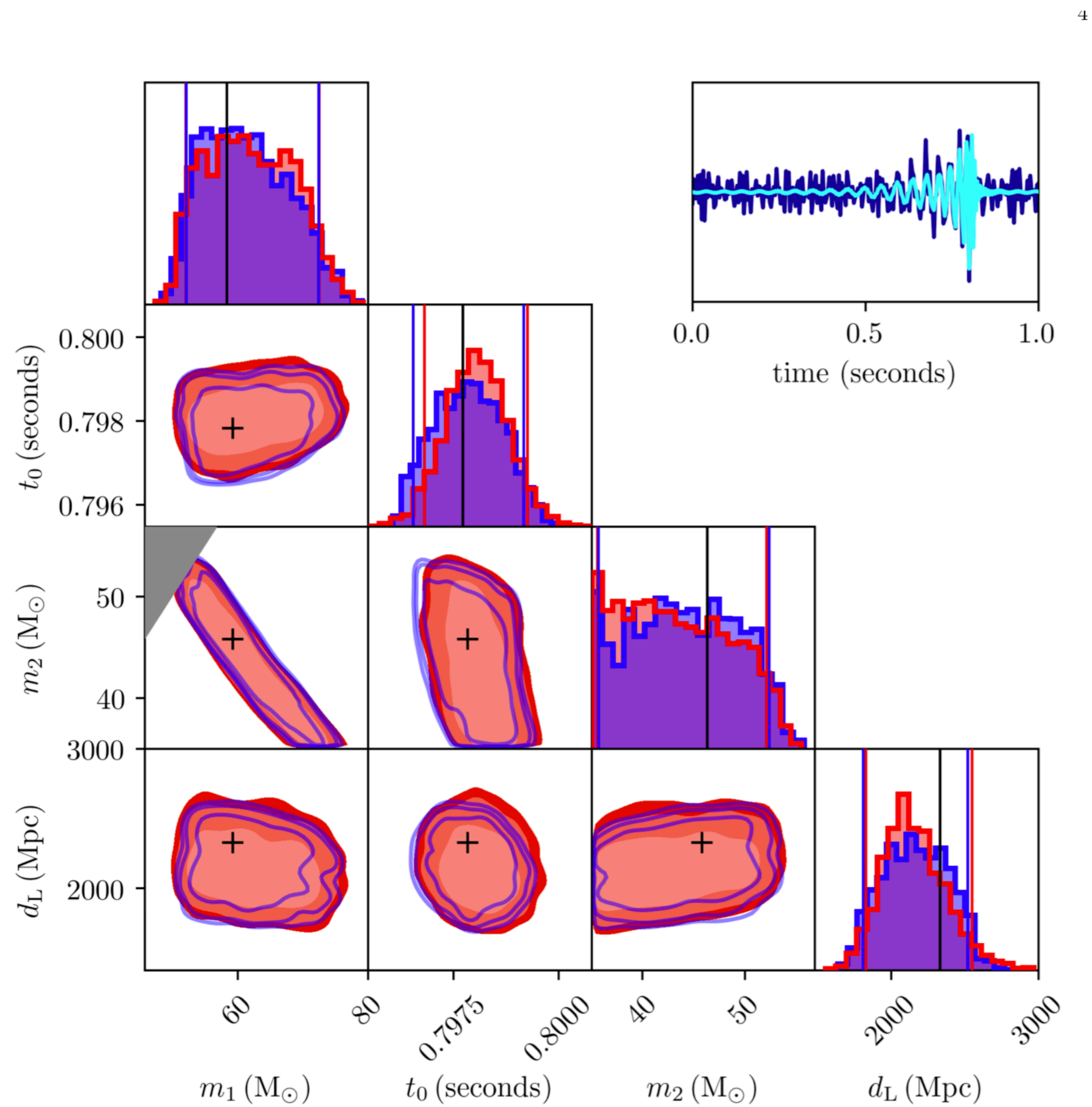






m_1= ? m_2=?

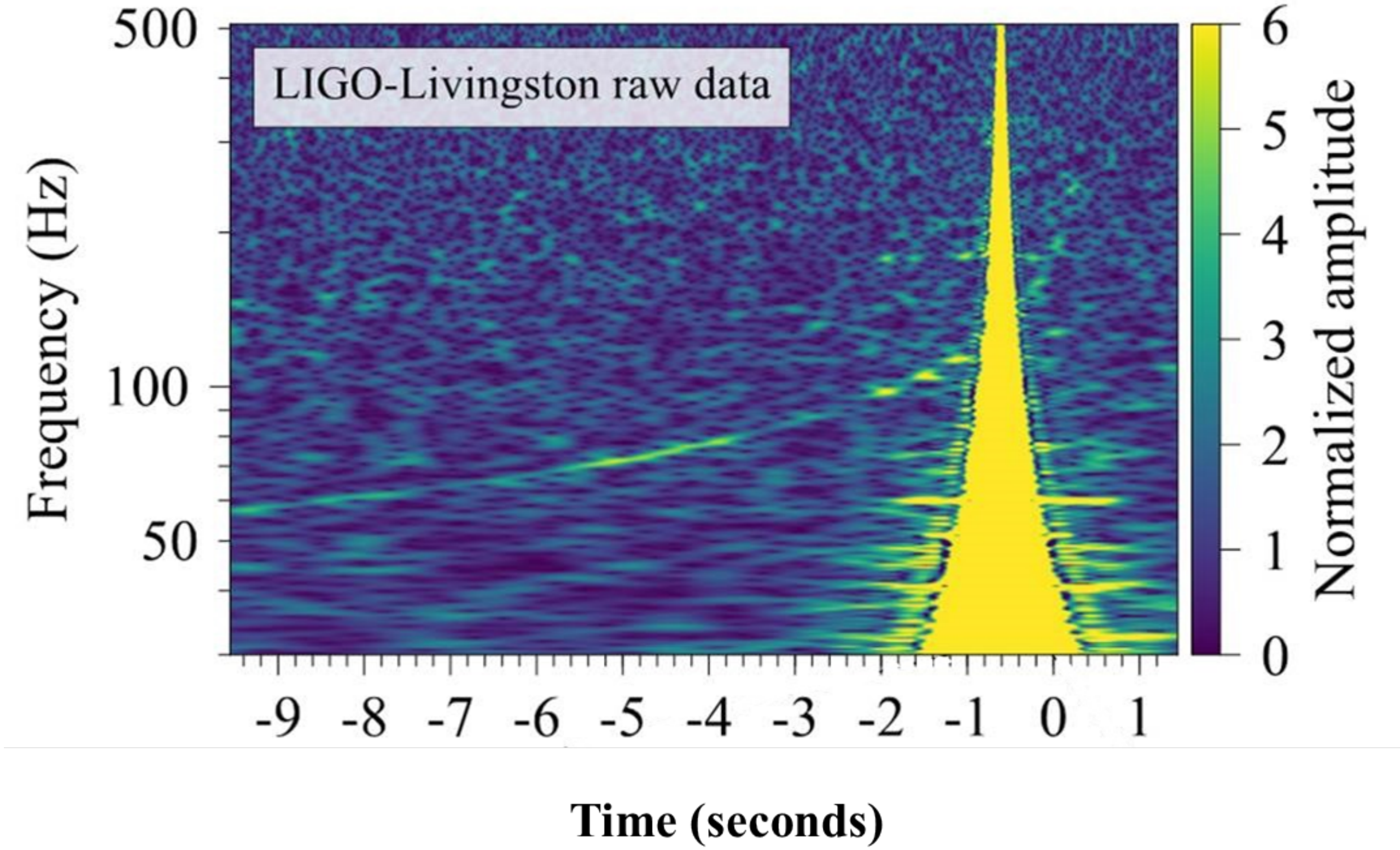
Conditional Variational Autoencoder



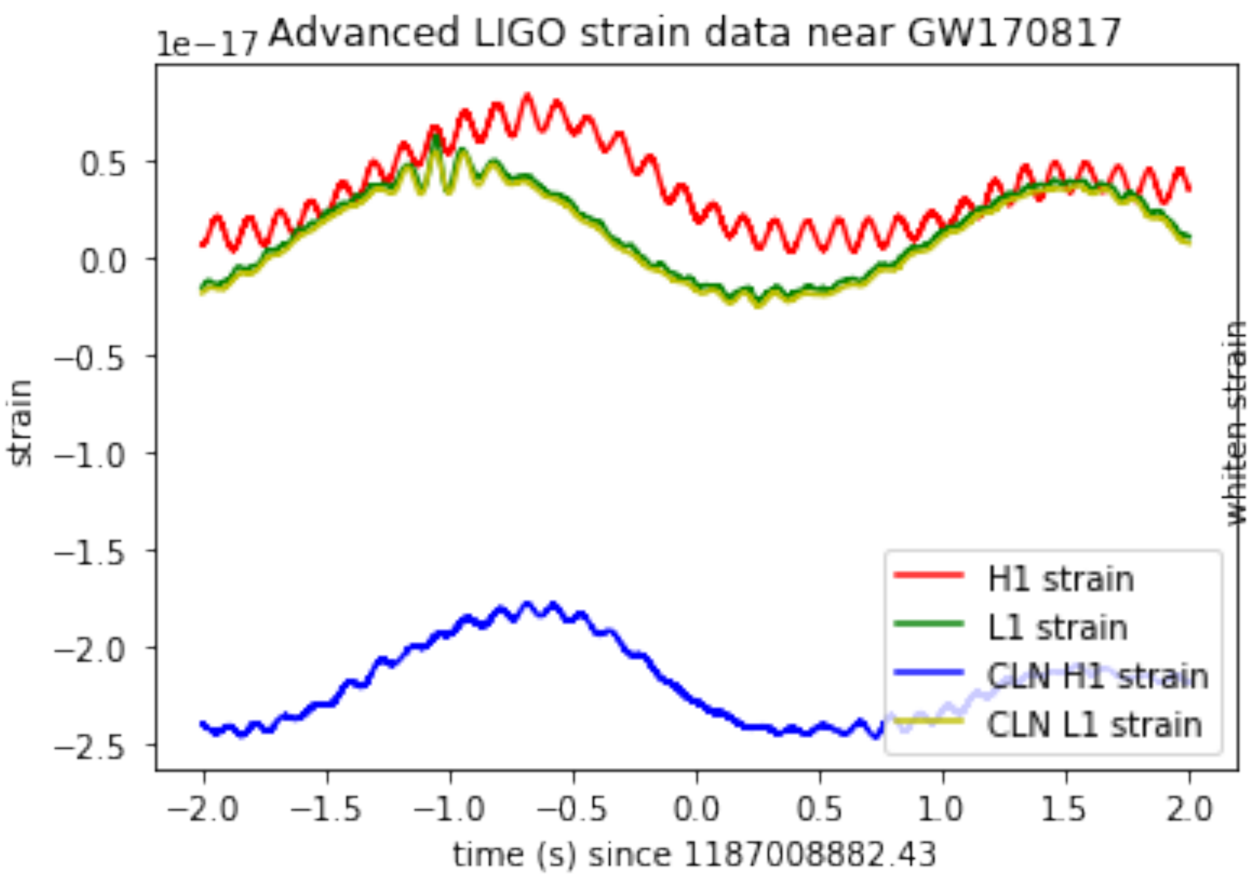
Gabbard et al. arxiv 1909.06296

Deglitch

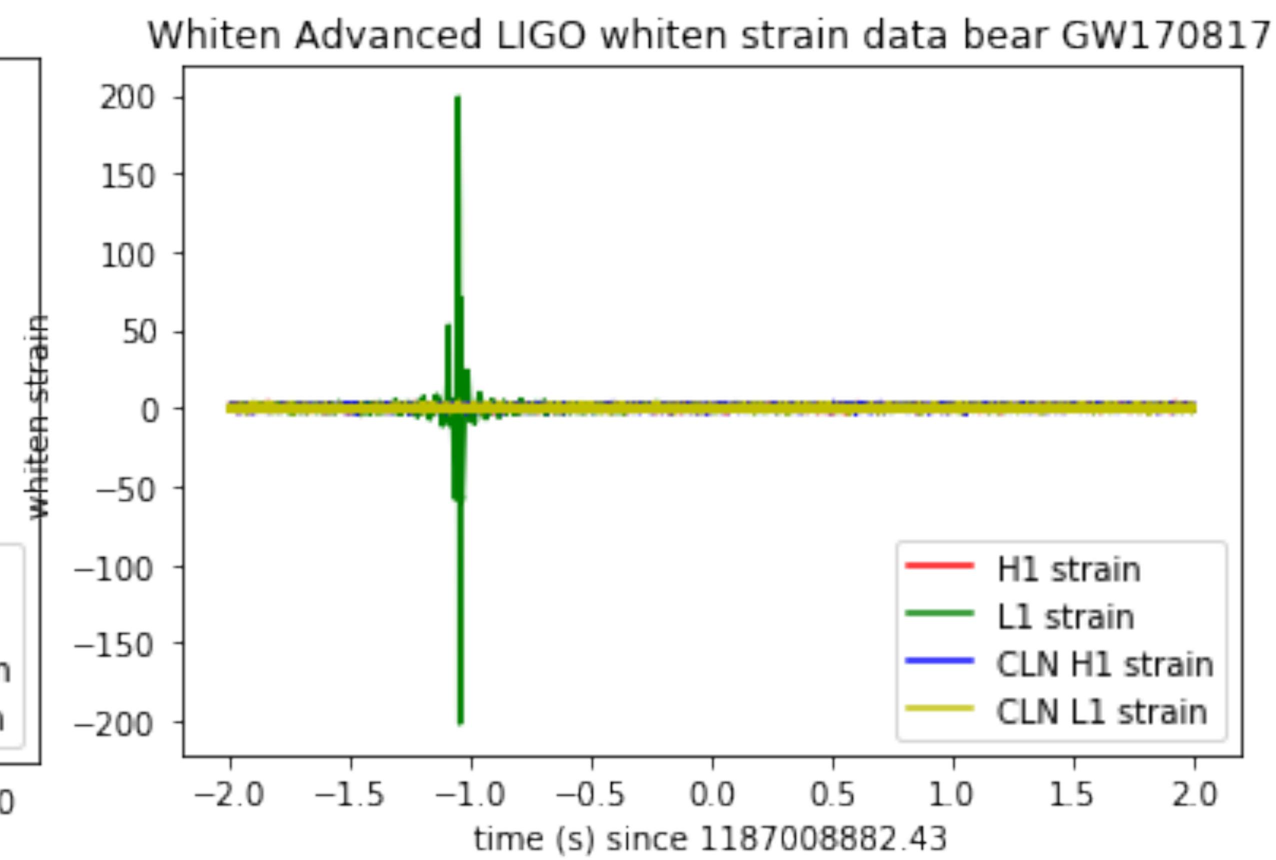
- false detection of GW signal
- change the PSD



Deglitch



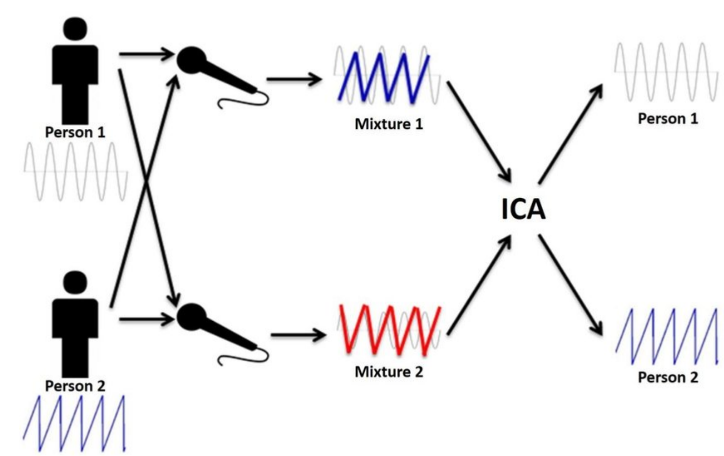
1e−17
Advanced LIGO strain data near GW170817
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
strain
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
time (s) since 1187008882.43
H1 strain
L1 strain
CLN H1 strain
CLN L1 strain



Whiten Advanced LIGO whiten strain data bear GW170817
200
150
100
50
0
−50
−100
−150
−200
whiten strain
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
time (s) since 1187008882.43
H1 strain
L1 strain
CLN H1 strain
CLN L1 strain

Independent Component Analysis

- Originally be used to solve the cocktail party problem
- Wildly used in many fields, like brain imaging-electroencephalogram
- Can it be applied to GW?



L.-R. David 2018

Independent Component Analysis

Consider a simple case: only two signal sources and two detectors

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \qquad \mathbf{x} = \mathbf{A}\mathbf{s}$$

observations mixing matrix signals

If we have the inverse matrix of A

$$\mathbf{s} = \mathbf{A}^{-1}\mathbf{x}$$

Consider the linear combination

$$\hat{\mathbf{s}} = \mathbf{w}\mathbf{x} = \mathbf{w}\mathbf{A}\mathbf{s} = \mathbf{z}\mathbf{s}$$

The Central Limit Theorem: the distribution of a sum of independent random variables tends toward a gaussian distribution

Maximize the nongaussianity of each $\hat{s}_i = \sum_j w_{ij} x_j$

FastICA

- Hyvarinen & Oja 2000
- Maximize the Negentropy $J(Y)=H(Y_{gauss})-H(Y)$

$$H(Y) = -\sum_j P(y_j) \log P(y_j)$$

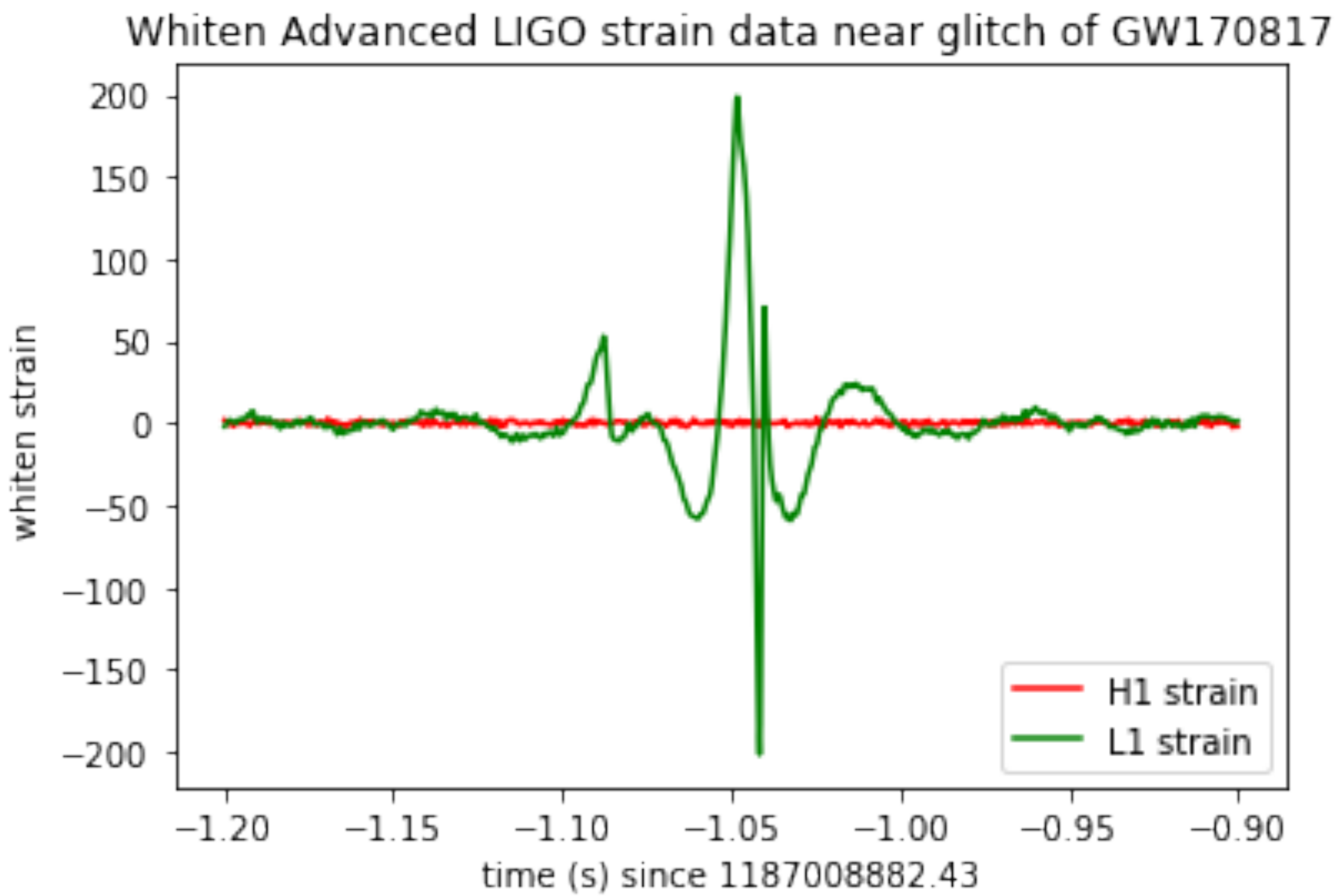
- Approximation of negentropy:

$$J(Y) \propto \left[E\{\phi(Y)\} - E\{\phi(v)\} \right]^2$$

$\phi(.)$: non-quadrtic func , v:gaussian random variable

- Algorithm: for i-th component, w^T_i the i-th row of w
  - step 1. initialize w^T_i (random)
  - step 2:
  - step 3:
  - step 4: if not converged, go back to step2

Whitened Strain

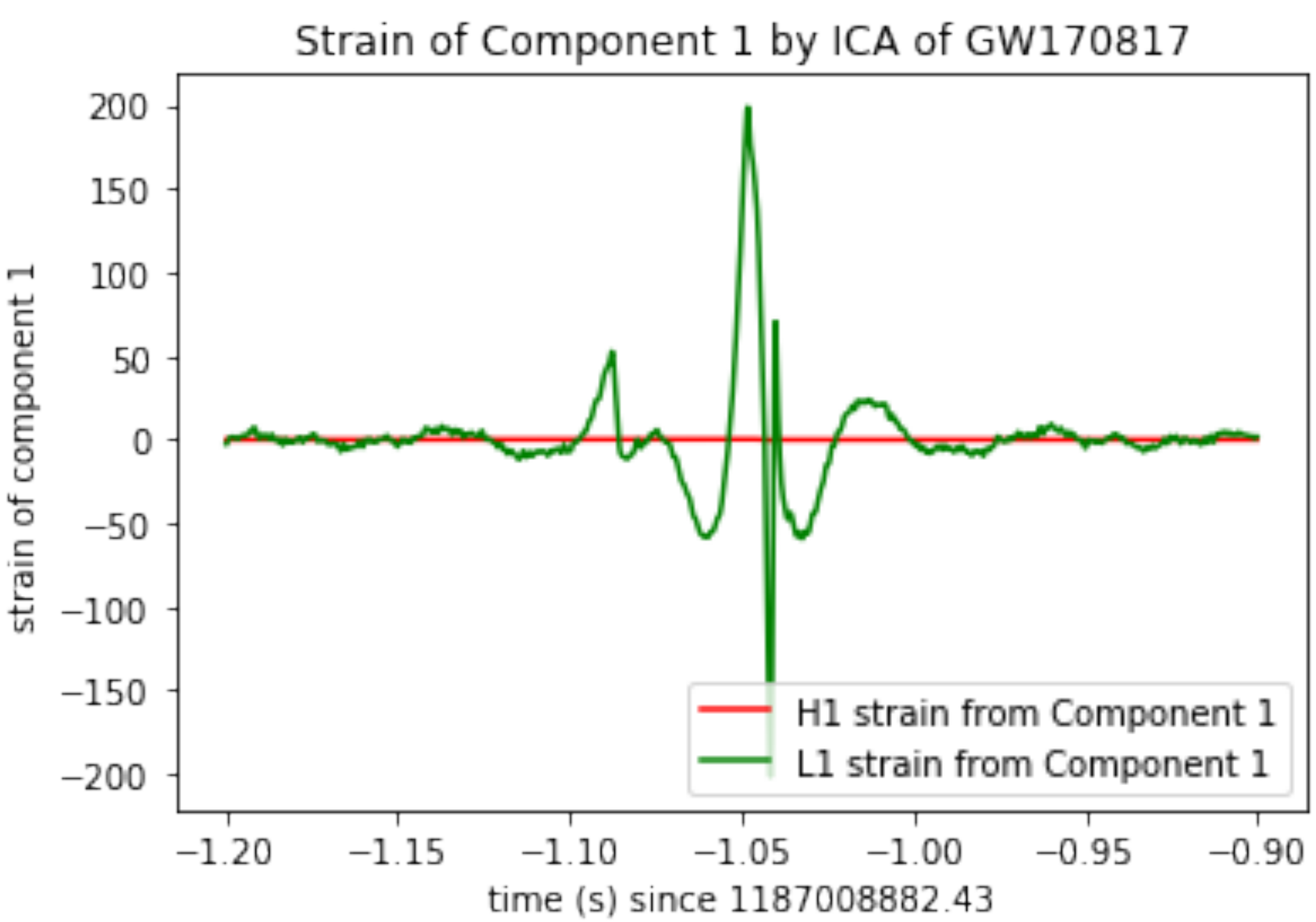


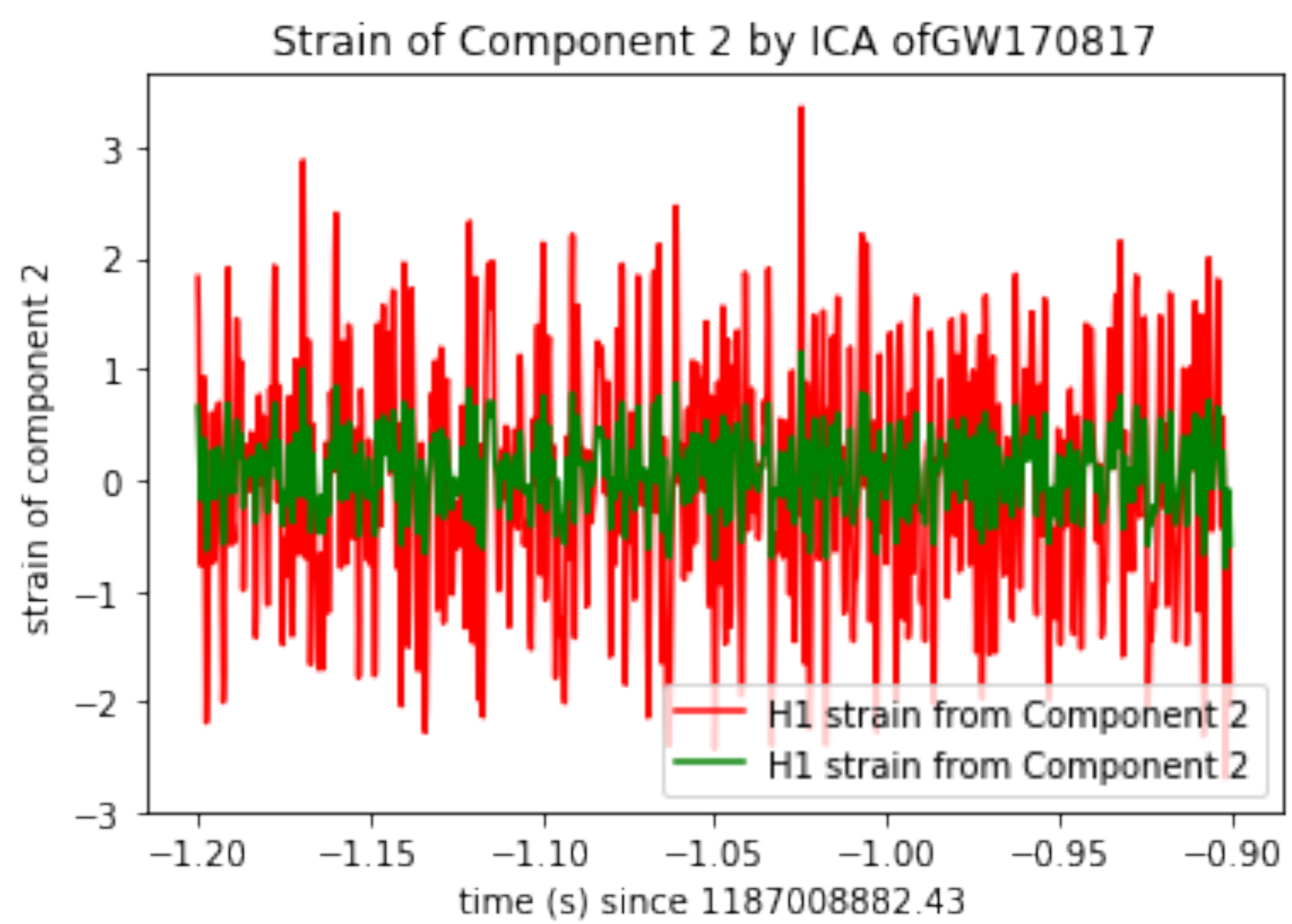
Applying FastICA

$$\begin{pmatrix} H1 \\ L1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} g \\ s \end{pmatrix}$$

$$\begin{pmatrix} H1' \\ L1' \end{pmatrix} = \mathbf{w}^{-1} \begin{pmatrix} g \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} H1'' \\ L1'' \end{pmatrix} = \mathbf{w}^{-1} \begin{pmatrix} 0 \\ s \end{pmatrix}$$



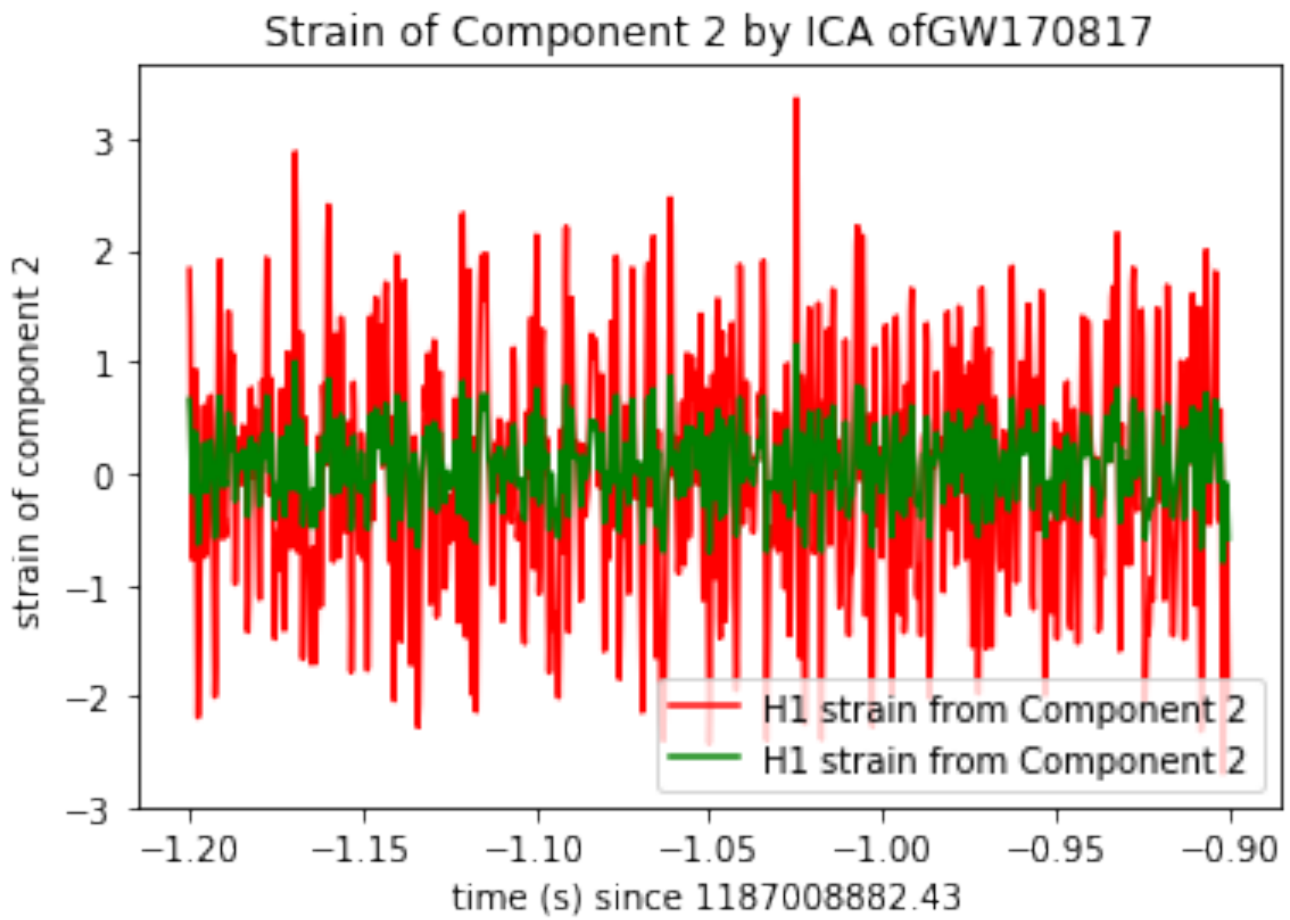


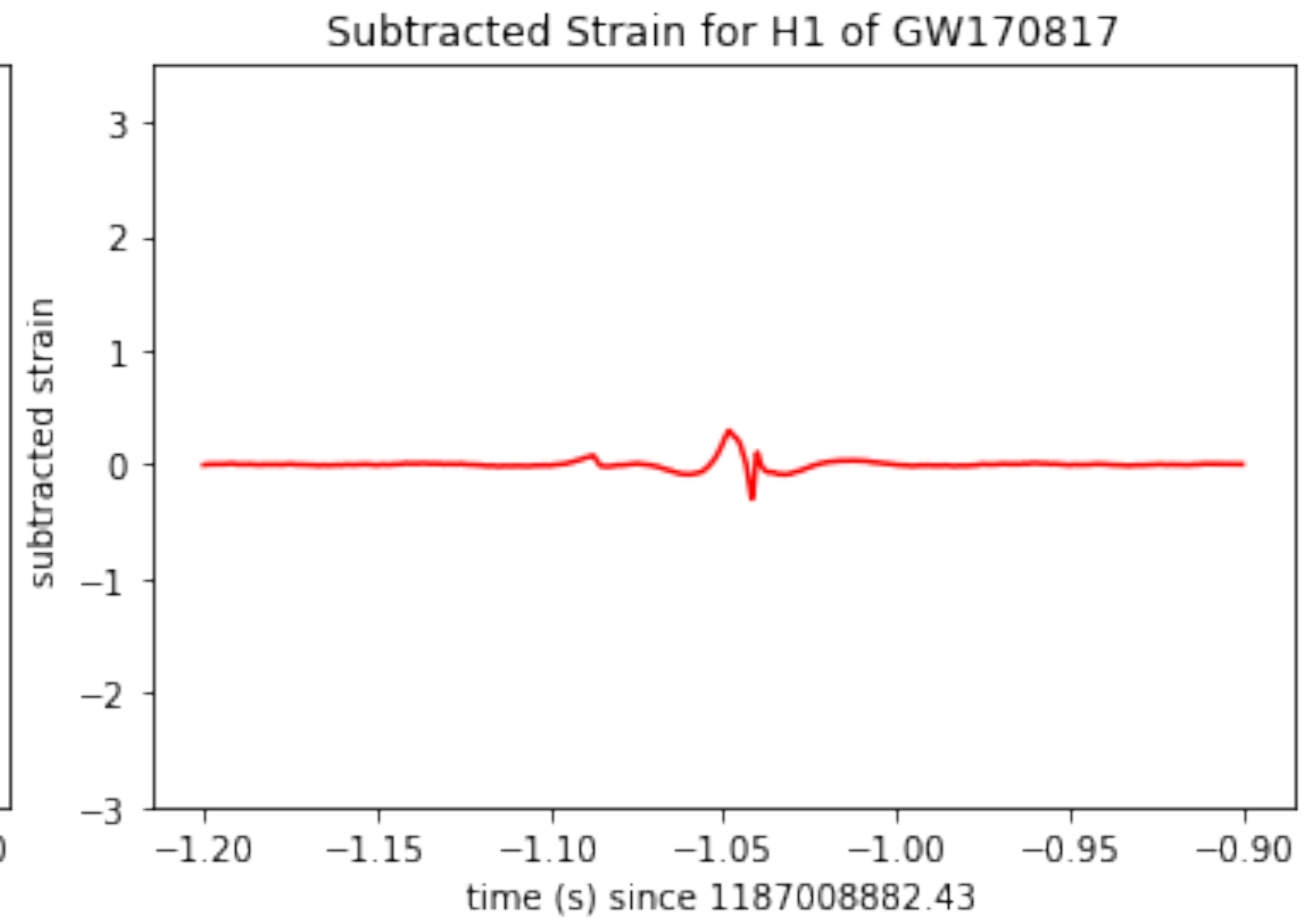
Applying FastICA

$$\begin{pmatrix} H1 \\ L1 \end{pmatrix} = \mathbf{A} \begin{pmatrix} g \\ s \end{pmatrix}$$

$$\begin{pmatrix} H1'' \\ L1'' \end{pmatrix} = \mathbf{w}^{-1} \begin{pmatrix} 0 \\ s \end{pmatrix}$$

$$H1 - H1''$$





- Glitch may be used to study the correlation of strain data with environment channels by ML
- about 30-40 channels are identified with correlation > 0.5 (Yokoyama; Oh)
- Using ICA with Environment Channels may be more efficient to deglitch

Define

$$\rho_{\mathrm{gw}} = \frac{c^2}{32\pi G} \langle \dot{h}_{ab}(t,\vec{x}) \dot{h}^{ab}(t,\vec{x}) \rangle$$

$$\left\langle \tilde{h}_I(t,f) \tilde{h}_J^*(t,f) \right\rangle = H(f) \int_{S^2} d\hat{\Omega} \left[F_I^+(\hat{\Omega},t) F_J^+(\hat{\Omega},t) P_+(\hat{\Omega}) + F_I^\times(\hat{\Omega},t) F_J^\times(\hat{\Omega},t) P_\times(\hat{\Omega}) \right] e^{i2\pi f\left(\hat{\Omega}\cdot\Delta\vec{x}/c\right)}$$

We can rewrite it as

$$\left\langle \tilde{h}_I(t,f) \tilde{h}_J^*(t,f) \right\rangle = H(f) \int_{S^2} d\hat{\Omega} \left[F_I^+(\hat{\Omega},t) F_J^+(\hat{\Omega},t) P_+(\hat{\Omega}) + F_I^\times(\hat{\Omega},t) F_J^\times(\hat{\Omega},t) P_\times(\hat{\Omega}) \right] e^{i2\pi f\left(\hat{\Omega}\cdot\Delta\vec{x}/c\right)}$$

What We Can Learn From Stochastic?

- Phase transition in the early universe
- Cosmic strings
- Superposition of
  - core-collapse supernovae
  - neutron-star instabilities
  - Binary mergers

Mechanisms for Anisotropic Background

- Confusion background arises from binary mergers (Regimbau & Chauvineaux 2007, Farmer & Phinney 2003)
- Core-collapse supernovae (Howell et al. 2004)
- Neutron-star excitations (Ferrari, Matarrese and Schneider 1999)
- Persistent emission from neutron stars (Regimbau & de Freitas Pacheco 2006)
- Compact objects orbit supermassive black hole (Sigl, Schnittman & Buonanno 2004) (massive stars form in a self-gravitating accretion disk around an active galactic nucleus)