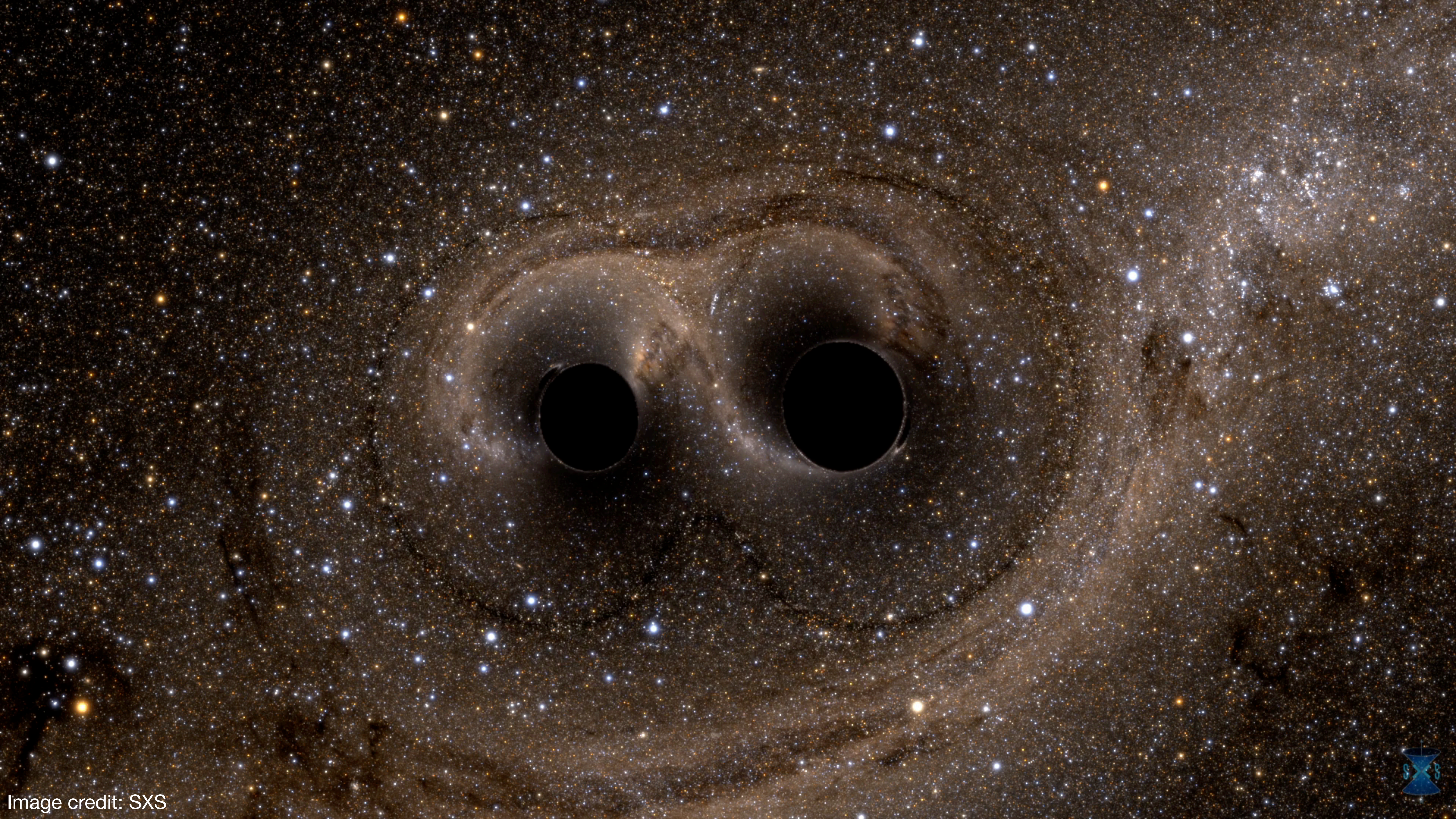


Image credit: SXS

Introduction to Parameter Estimation of Compact Binary Coalescences

Vivien Raymond
Cardiff University

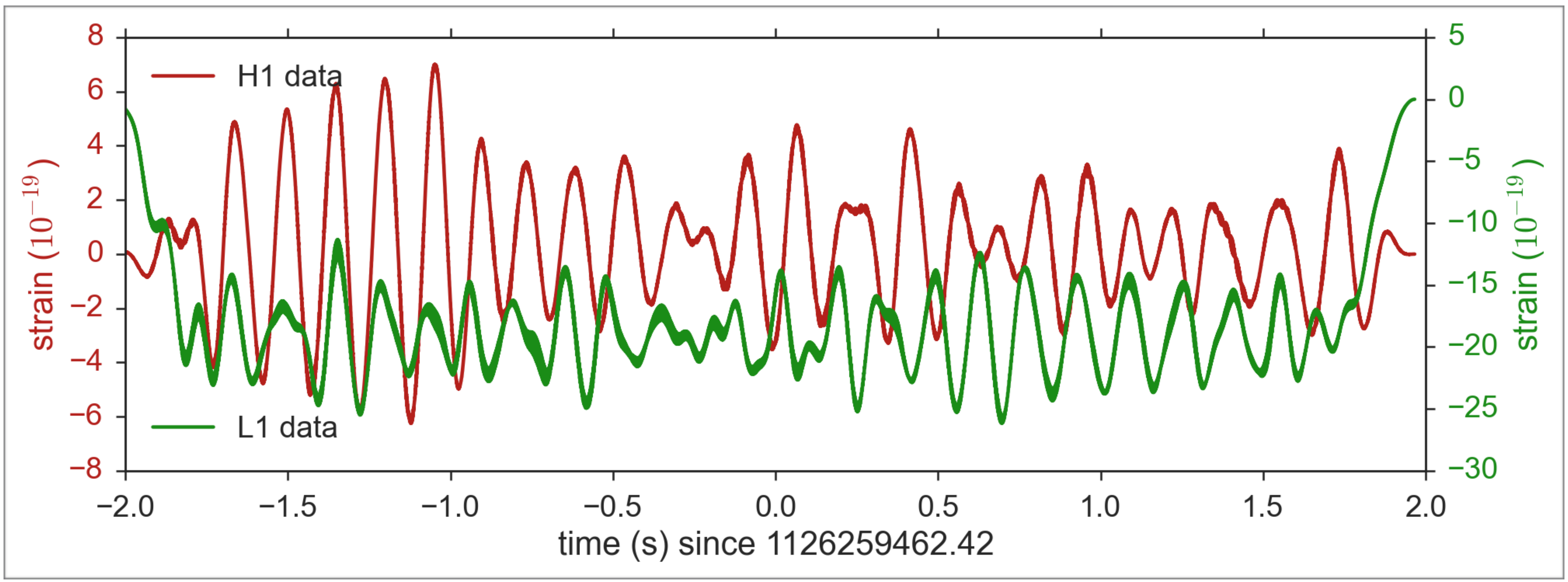
with contributions from ***Katerina Chatziioannou****, Flatiron Institute*

GW-ODW2 held from 08 Apr 2019 to 10 Apr 2019 in Paris, France

LIGO-G1900659

Starting from **strain data**

- GW150914: September 14, 2015 at 09:50:45 UTC:

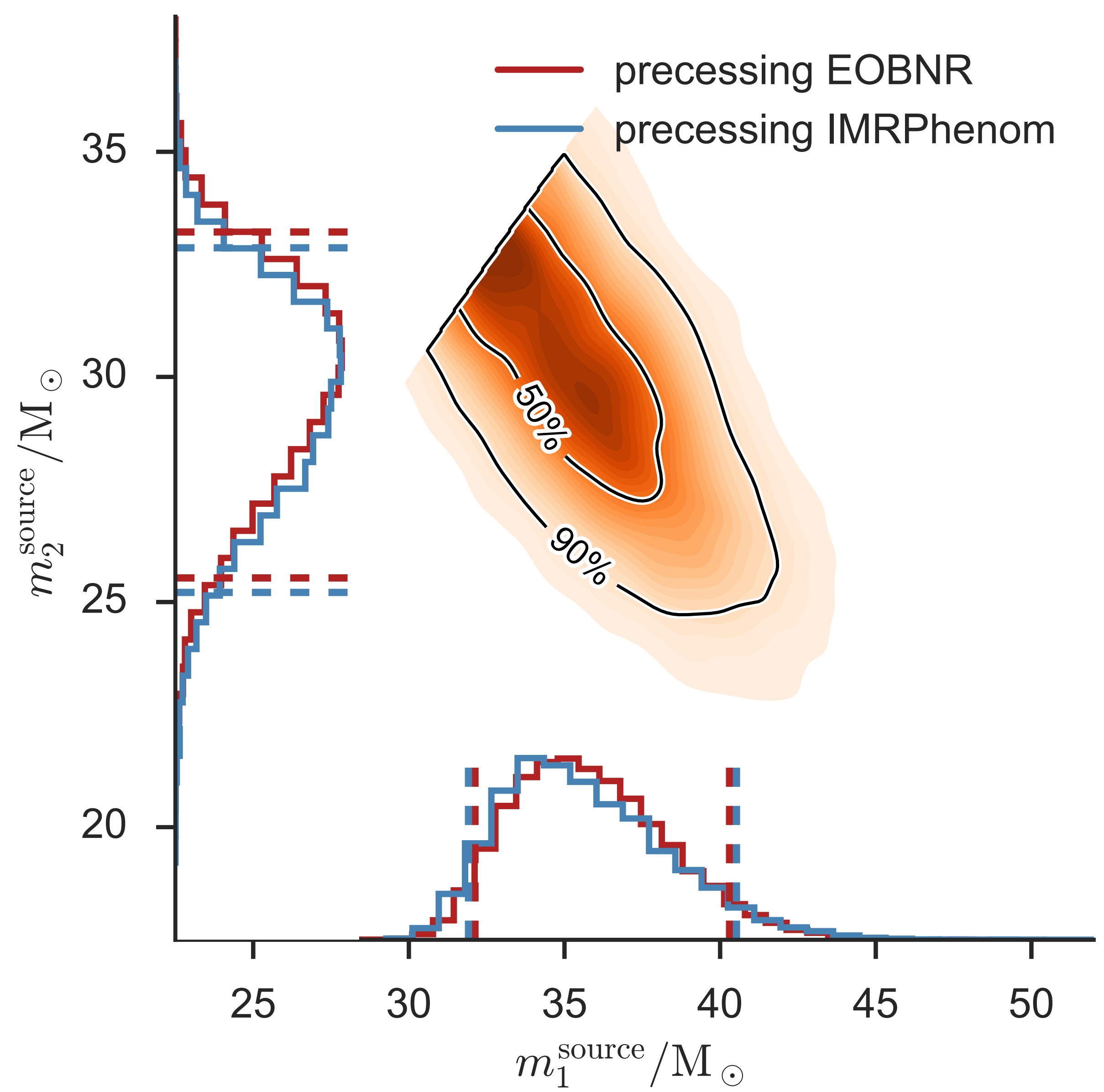


To Probability Density Function

- GW150914 masses estimates:

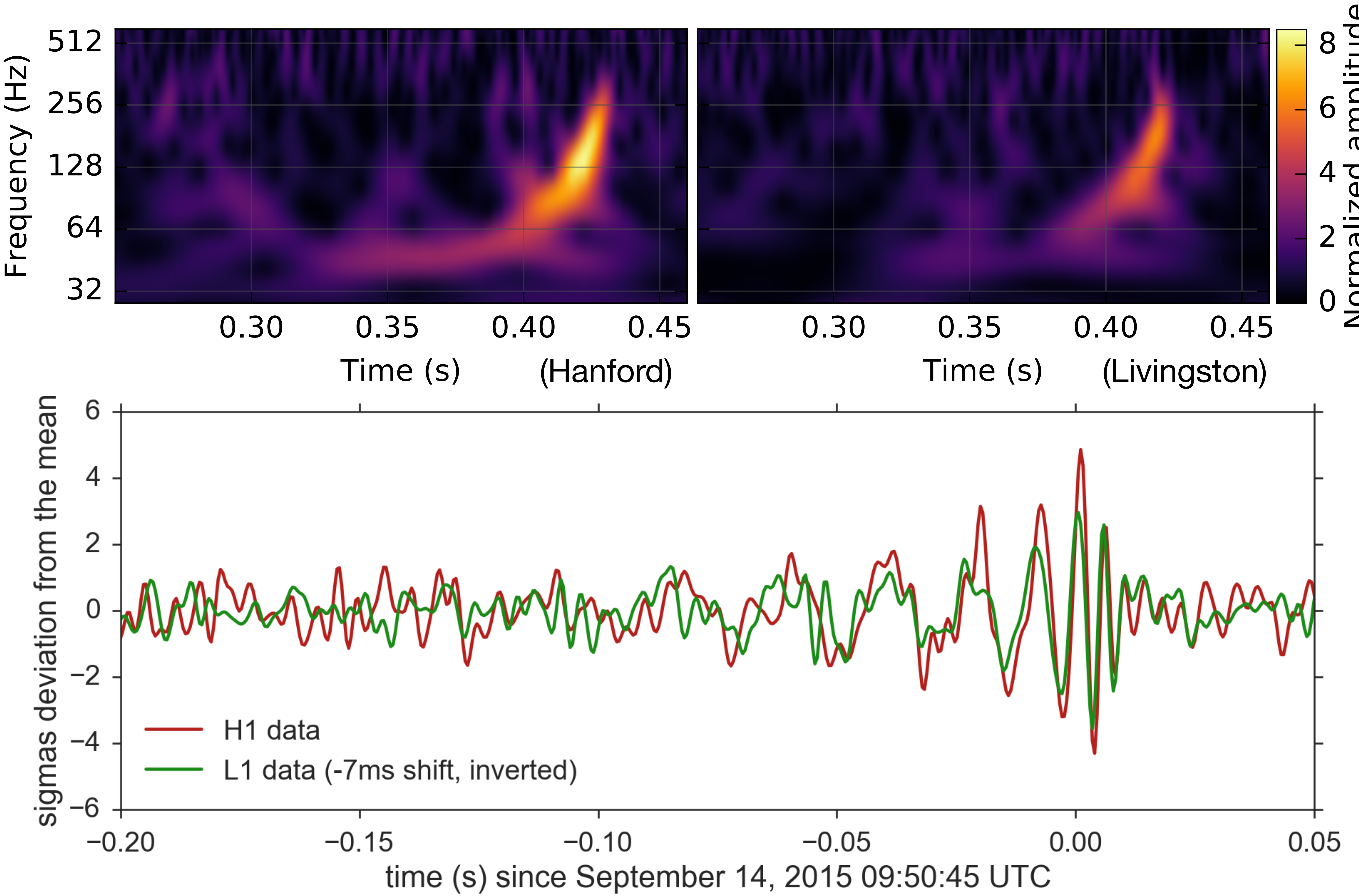
$$m_1 = 35.4^{+5.0\pm0.1}_{-3.4\pm0.3}\,\mathrm{M}_\odot$$

$$m_2 = 28.9^{+3.3\pm0.3}_{-4.3\pm0.3}\,\mathrm{M}_\odot$$



[LIGO-Virgo Collaboration, 2016]

GW150914 observation



Parameter Estimation

- We want the **posterior** probability of parameters $\vec{\lambda}$, given the data $\vec{x}$. With **Bayes'** theorem:

$$p(\vec{\lambda}\,|\,\mathbf{d}, M) = \frac{p(\vec{\lambda}\,|\,M)\,p(\mathbf{d}\,|\,\vec{\lambda}, M)}{p(\mathbf{d}\,|\,M)}$$

- Fit a **model** to the data (**noise** and **signal** models)
- Build a **likelihood** function
- Specify **prior** knowledge
- **Numerically** estimate the resulting **distribution** (**sampling** algorithms)

Parameter Estimation

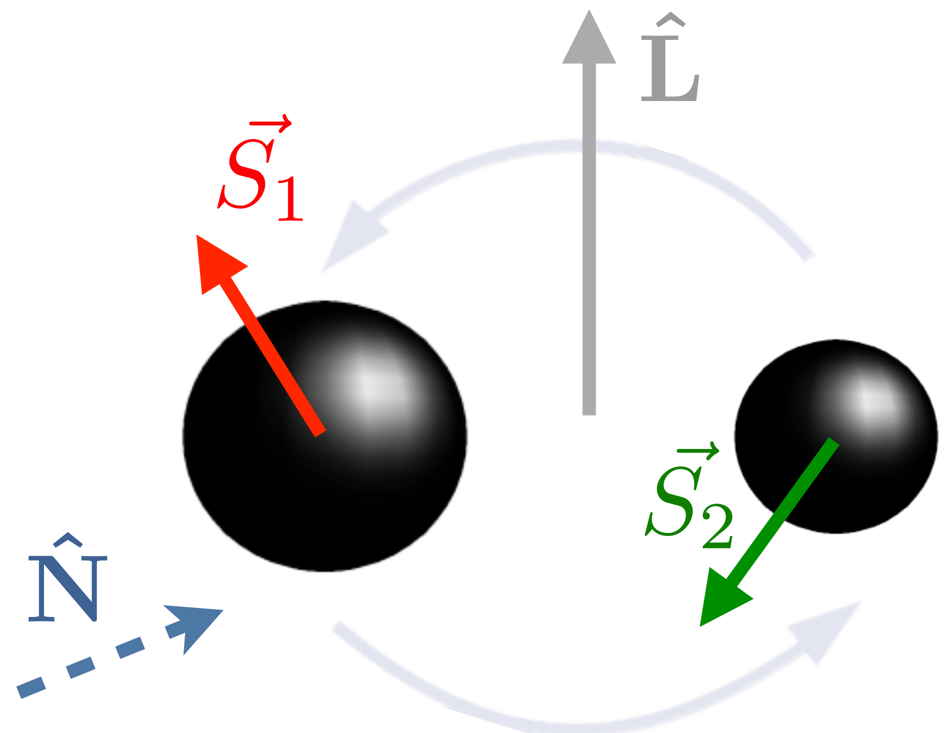
- We want the **posterior** probability of parameters $\vec{\lambda}$, given the data $\vec{x}$. With **Bayes'** theorem:

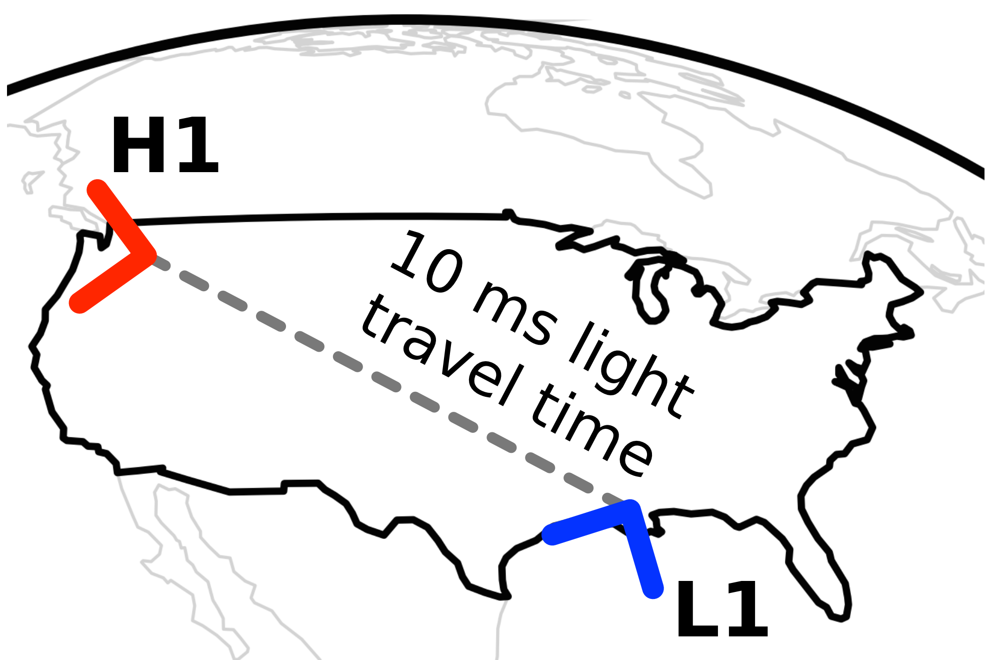
$$p(\vec{\lambda}\,|\,\mathbf{d},M) = \frac{p(\vec{\lambda}\,|\,M)\,p(\mathbf{d}\,|\,\vec{\lambda},M)}{p(\mathbf{d}\,|\,M)}$$

- Fit a **model** to the data (**noise** and **signal** models)
- Build a **likelihood** function
- Specify **prior** knowledge
- **Numerically** estimate the resulting **distribution** (**sampling** algorithms)

Compact Binary Coalescence

- **Intrinsic** parameters: primary and secondary **masses** and **spins** (and eccentricity)

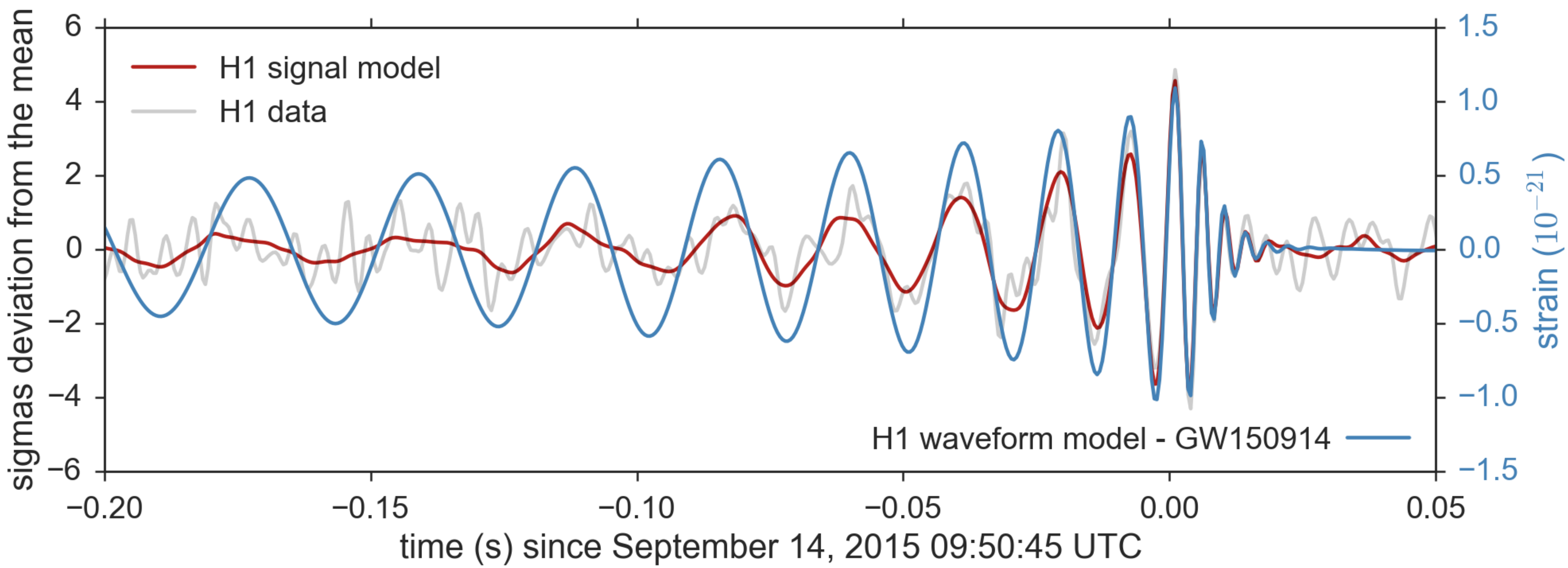




- **Extrinsic**: time, **sky-position**, distance, **orientation**, reference phase

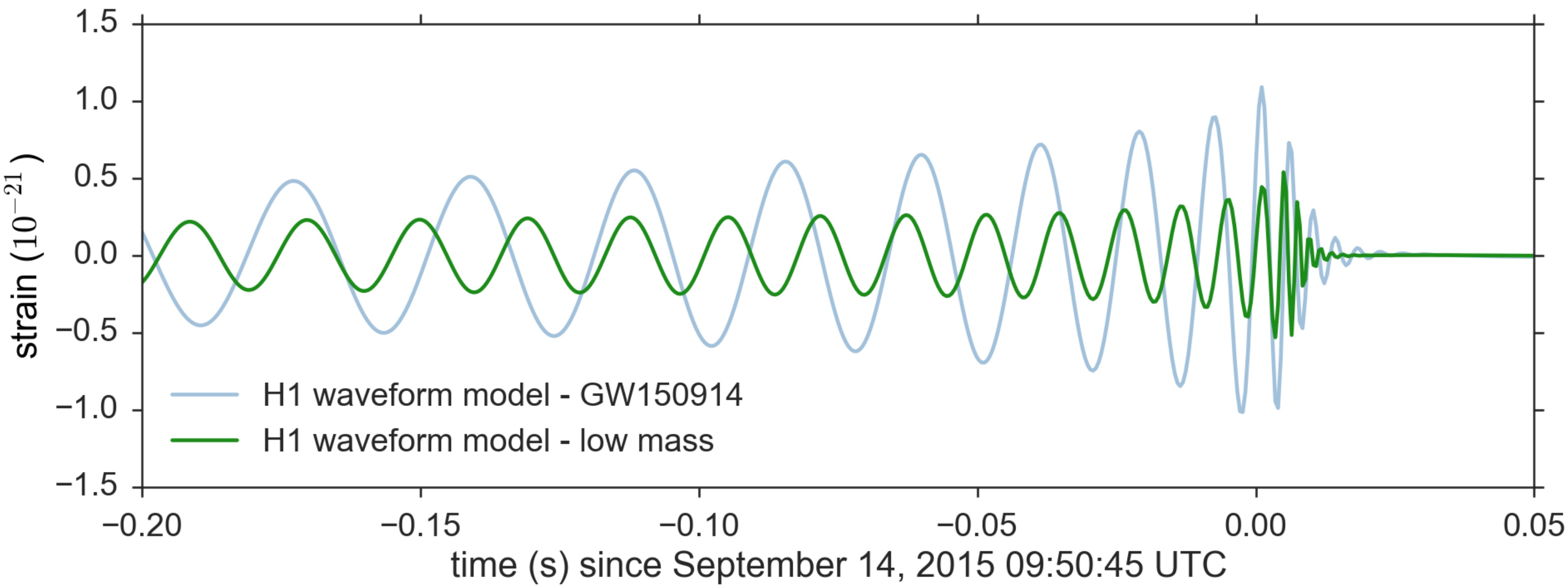
Credit: LIGO/Virgo

Gravitational waveform models

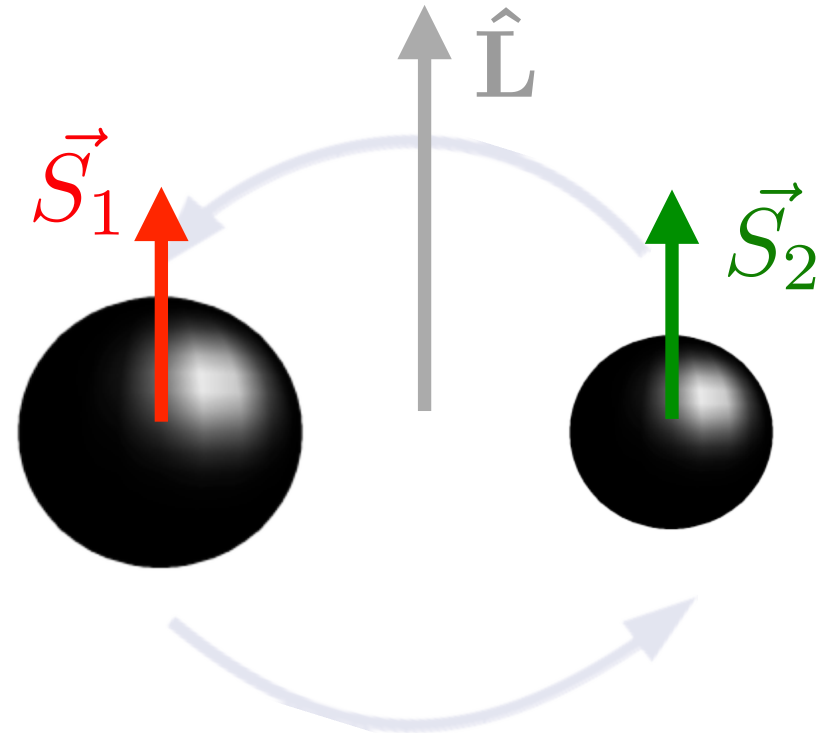


Masses from the inspiral and ringdown

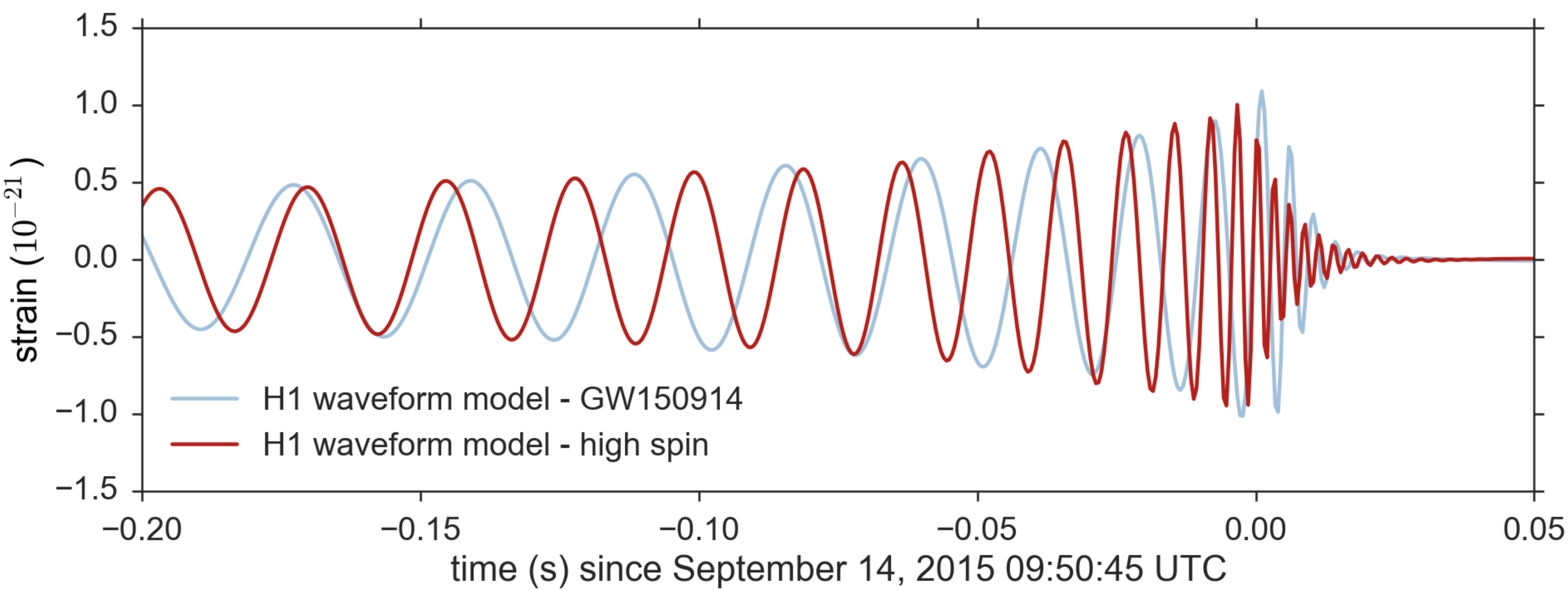
- Chirp mass: $$\mathcal{M} = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}}$$
- Mass ratio: $$q = \frac{m_1}{m_2}$$
- Total mass: **ringdown**



Effects of spins

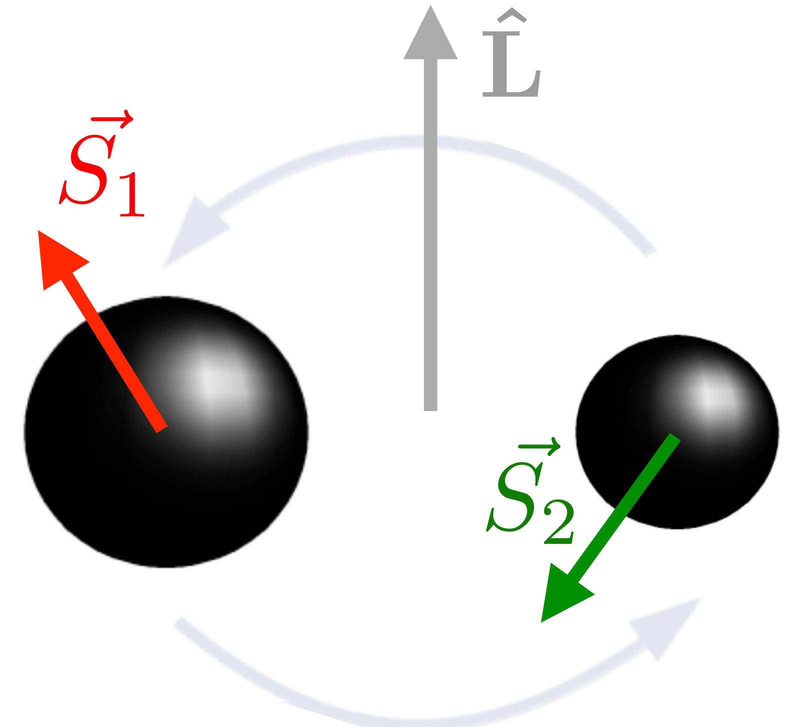


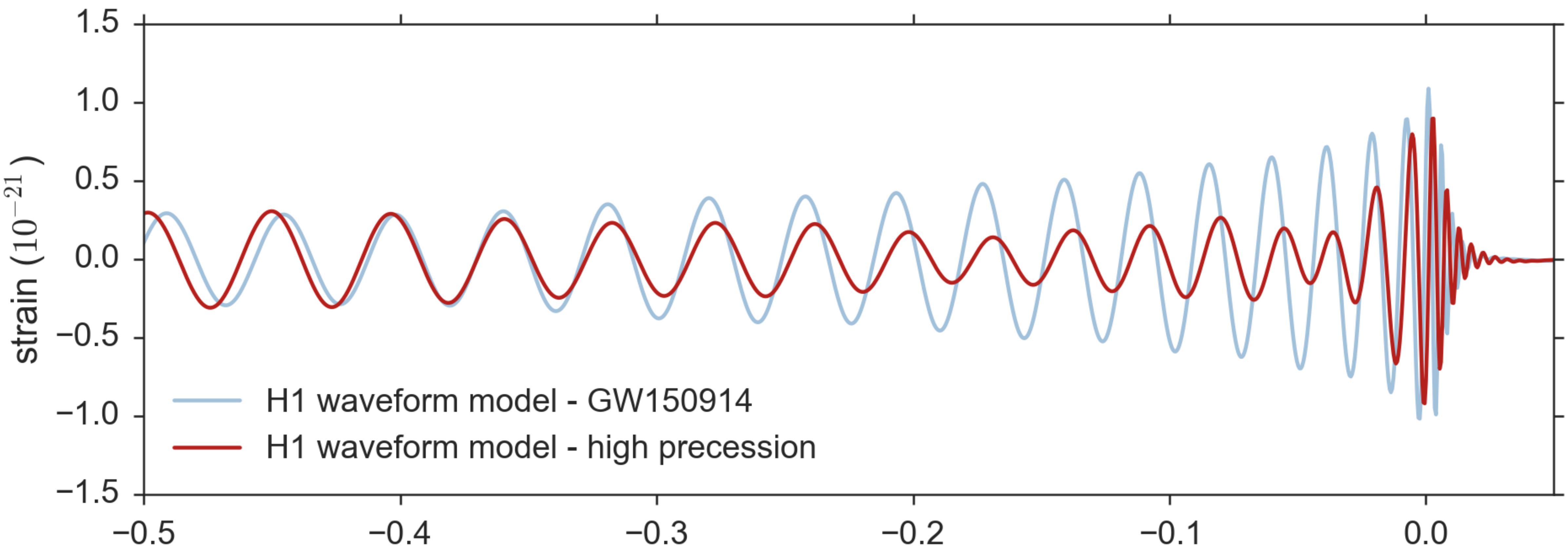
- 2 spin vectors
  - **Magnitude: orbital hang-up**
  - Mis-alignment: precession and modulations



Effects of spins

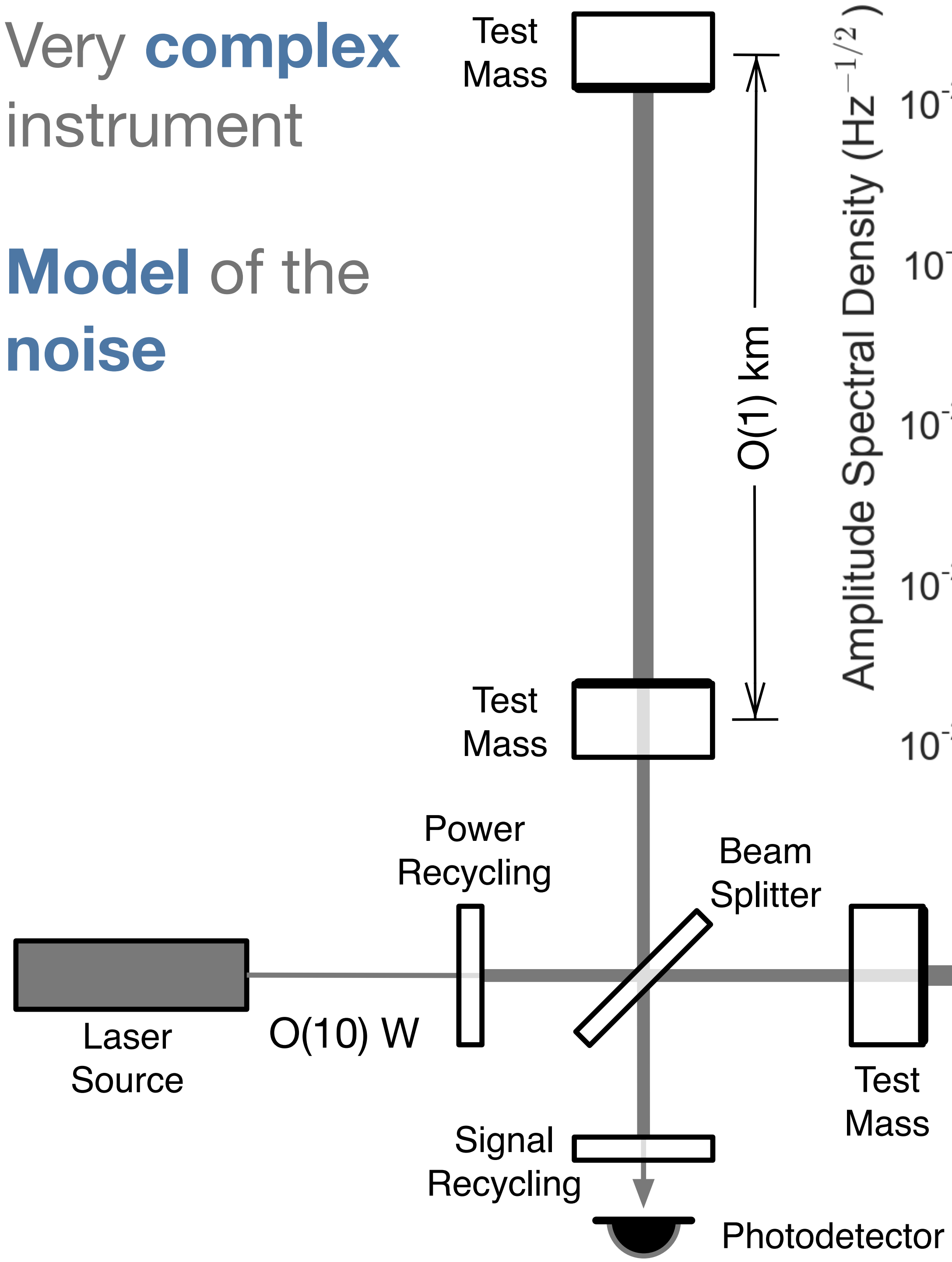
- 2 spin vectors
  - Magnitude: orbital hang-up
  - **Mis-alignment: precession and modulations**

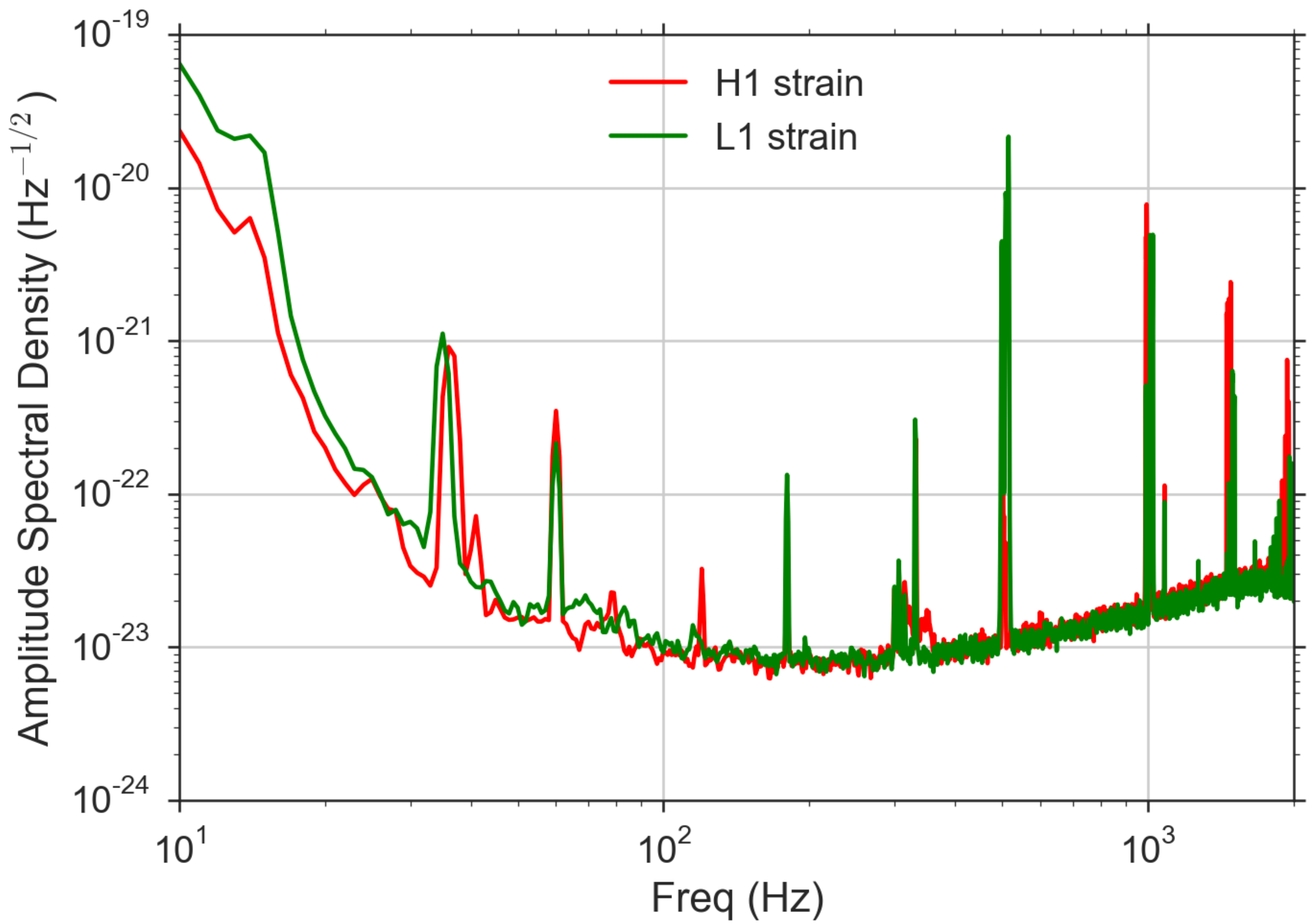




Observatories

- Very **complex** instrument
- **Model** of the **noise**

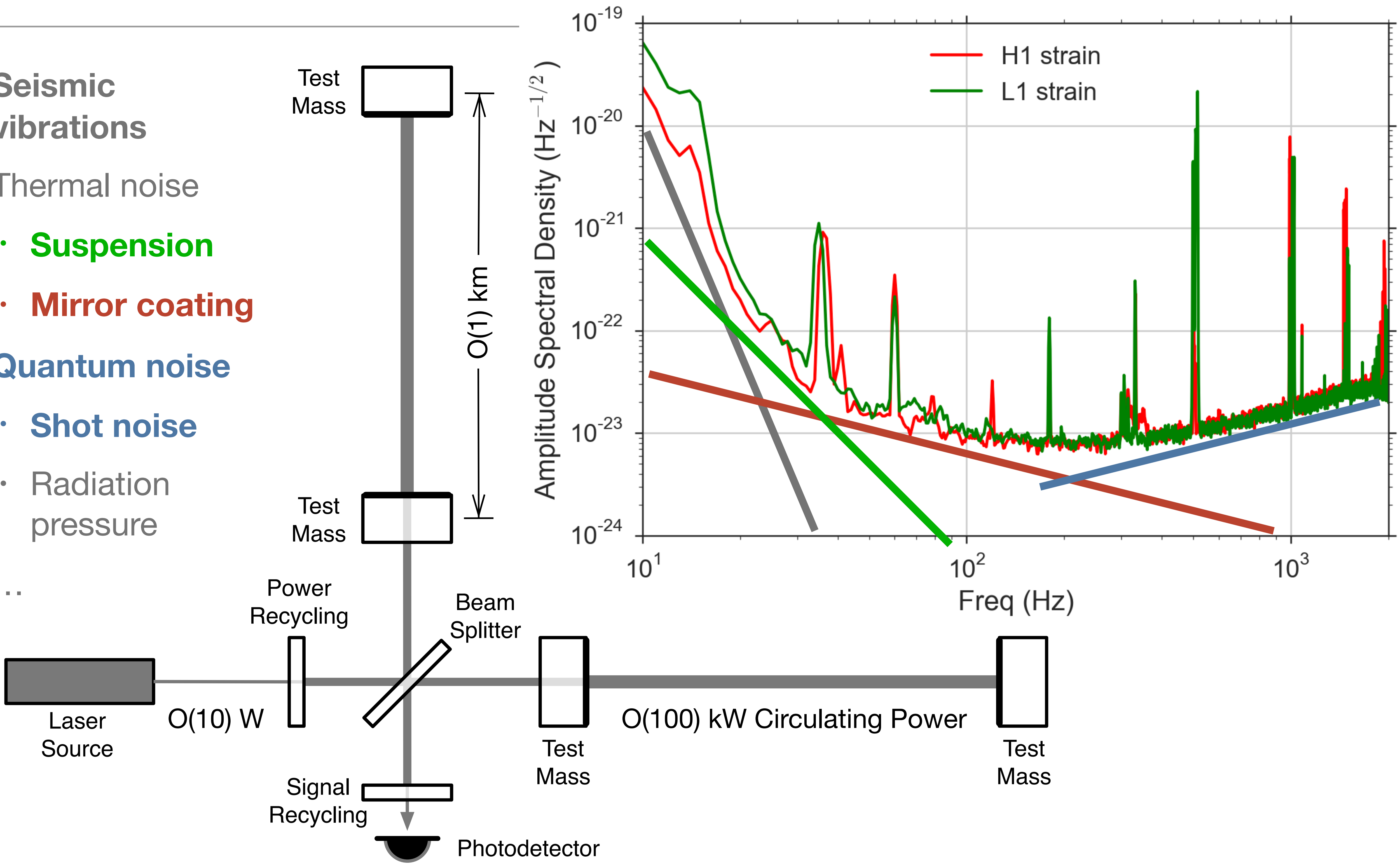




[LIGO-Virgo Collaboration, 2016]

Noise sources

- **Seismic vibrations**
- Thermal noise
  - **Suspension**
  - **Mirror coating**
- **Quantum noise**
  - **Shot noise**
  - Radiation pressure
- …



[LIGO-Virgo Collaboration, 2016]

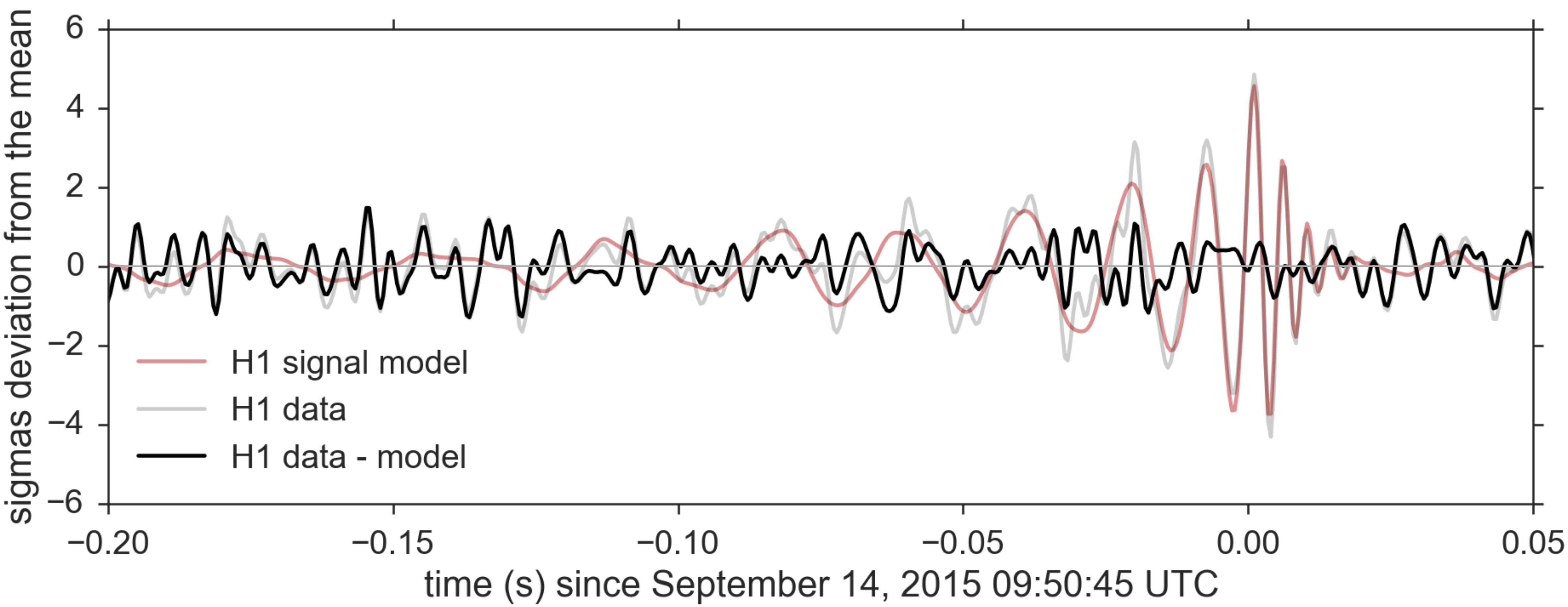
Parameter Estimation

- We want the **posterior** probability of parameters $\vec{\lambda}$, given the data $\mathbf{d}$. With **Bayes'** theorem:

$$p(\vec{\lambda} \,|\, \mathbf{d}, M) = \frac{p(\vec{\lambda} \,|\, M)\, p(\mathbf{d} \,|\, \vec{\lambda}, M)}{p(\mathbf{d} \,|\, M)}$$

- Fit a **model** to the data (**noise** and **signal** models)
- Build a **likelihood** function
- Specify **prior** knowledge
- **Numerically** estimate the resulting **distribution** (**sampling** algorithms)

The Likelihood



- How close is the **remainder** to the **mean**?
  - Assumptions: **gaussianity** and **stationarity**

The Likelihood

Gravitational wave hits the detector.

The detector **records**:

$$\mathbf{d} = \mathbf{n} + R\left[\mathbf{h}(\vec{\lambda})\right]$$

The **available data** is the **detector noise** plus the **detector response** to a **gravitational wave** of certain **parameters**.

Likelihood (single data point)

$$d_1 = n_1 + R\left[h_1(\overrightarrow{\lambda})\right]$$

- Noise probability $p(n_1)$
- Residual probability $p(d_1 - R\left[h_1(\overrightarrow{\lambda'})\right])$
- Are they compatible? $p(d_1 - R\left[h_1(\overrightarrow{\lambda'})\right]) \stackrel{?}{=} p(n_1)$

$$p(d_1 - R\left[h_1(\overrightarrow{\lambda'})\right]) = p(d_1 \mid \overrightarrow{\lambda'})$$

Probability of drawing $\mathbf{d}_1 - R\left[\mathbf{h}_1(\overrightarrow{\lambda'})\right]$ from the noise distribution under the null hypothesis

Likelihood (many data points)

- discrete frequency bins:

$$\mathbf{d} = \left\{ d_1, d_2, \ldots, d_{N_f} \right\}$$

$$= \left\{ R\left[h_1(\overrightarrow{\lambda}) \right] + n_1, R\left[h_2(\overrightarrow{\lambda}) \right] + n_2, \ldots, R\left[h_{N_f}(\overrightarrow{\lambda}) \right] + n_{N_f} \right\}$$

- Joint probability for the noise from all frequency bins:

$$p(\mathbf{d} \mid \overrightarrow{\lambda}) = p\left(R\left[h_1(\overrightarrow{\lambda'}) \right] + n_1, R\left[h_2(\overrightarrow{\lambda'}) \right] + n_2, \ldots, R\left[h_{N_f}(\overrightarrow{\lambda'}) \right] + n_{N_f} \right)$$

$$\stackrel{?}{=} p(n_1, n_2, \ldots, n_{N_f})$$

Likelihood: noise model

Noise correlation matrix

- Gaussian noise

$$p(n_1, n_2, \ldots, n_{N_f}) \approx e^{-\frac{1}{2} n_i C_{ij}^{-1} n_j}$$

- Stationary noise

$$C_{ij} = \frac{1}{2} S_n(f_i)\, \delta_{ij}$$

n^{th} detector PSD

Likelihood

- Probability of obtaining data $\mathbf{d}$ assuming signal $h(\overrightarrow{\lambda})$ and that the noise is **stationary** and **gaussian**:

$$p(\mathbf{d}\,|\,\overrightarrow{\lambda}, M) \approx \exp\left(-\frac{1}{2}(\mathbf{d} - R[h(\overrightarrow{\lambda})]\,|\,\mathbf{d} - R[h(\overrightarrow{\lambda})])\right)$$

- With the discrete bins now continuous:

$$(a\,|\,b) = 2\int_0^{+\infty} df\, \frac{a^*(f)\,b(f) + a(f)\,b^*(f)}{S_n(f)}$$

Likelihood

- In practice, we use:

$$p(\mathbf{d} \mid \overrightarrow{\lambda}, M) \approx \exp\left(-2 \sum_{n=1}^{N_{det}} \int_{f_{low}}^{f_{high}} df \, \frac{\left| \mathbf{d}_n(f) - R_n \left[h(f; \overrightarrow{\lambda}), f; \overrightarrow{\lambda} \right] - g_n(f; \overrightarrow{\lambda}) \right|^2}{S_n(f; \overrightarrow{\lambda})} \right)$$

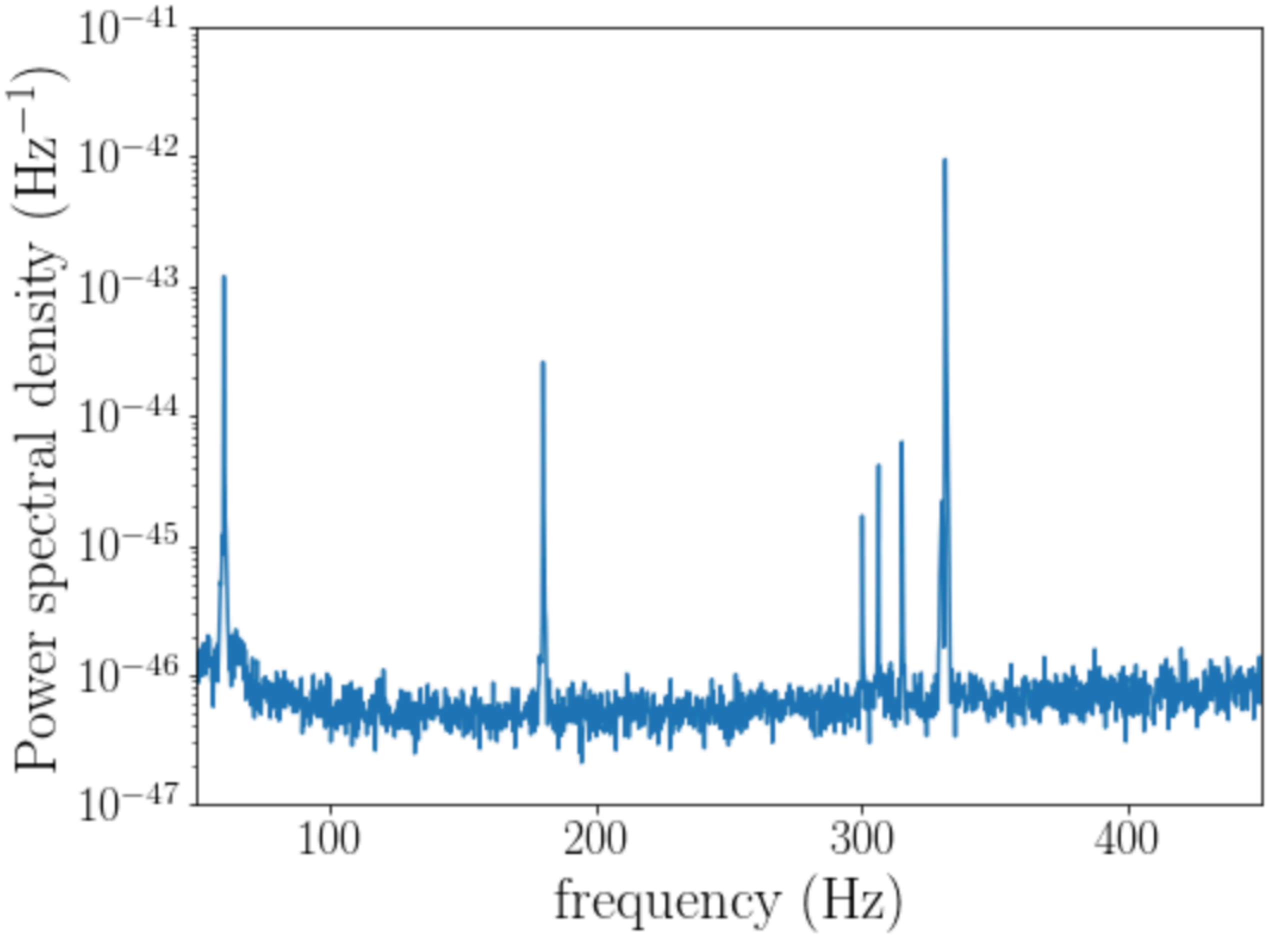
Likelihood

- In practice, we use:

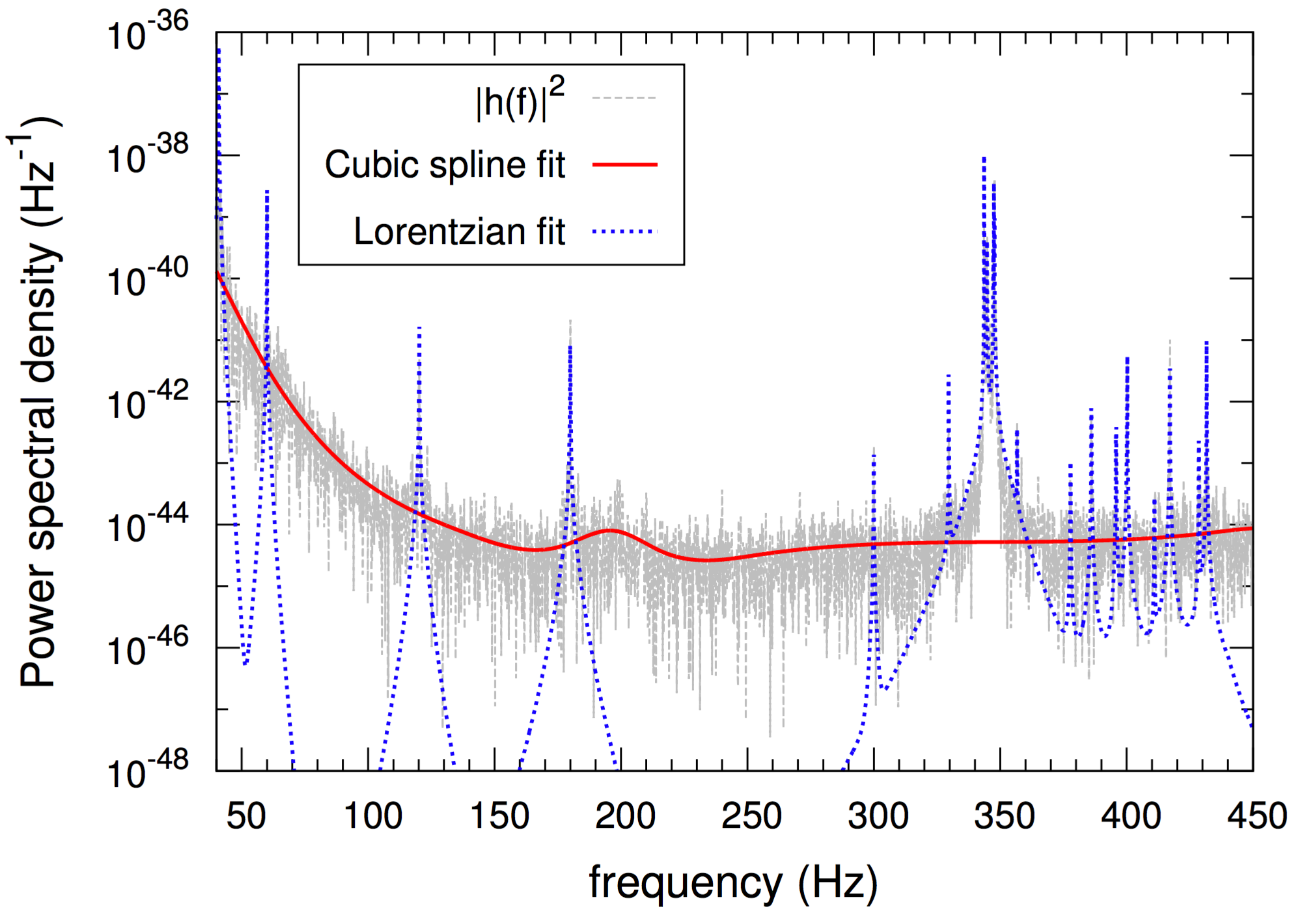
$$p(\mathbf{d} \mid \overrightarrow{\lambda}, M) \approx \exp\left(-2 \sum_{n=1}^{N_{det}} \int_{f_{low}}^{f_{high}} df \, \frac{\left| \mathbf{d}_n(f) - R_n \left[h(f; \overrightarrow{\lambda}), f; \overrightarrow{\lambda} \right] - g_n(f; \overrightarrow{\lambda}) \right|^2}{S_n(f; \overrightarrow{\lambda})} \right)$$

Noise model: PSD

Off-source (average)



On-source



[Littenberg and Cornish, 2014]

Likelihood

- In practice, we use:

$$p(\mathbf{d} \,|\, \overrightarrow{\lambda}, M) \approx \exp\left(-2 \sum_{n=1}^{N_{det}} \int_{f_{low}}^{f_{high}} df \, \frac{\left| \mathbf{d}_n(f) - R_n\left[h(f;\,\overrightarrow{\lambda}), f;\,\overrightarrow{\lambda}\right] - g_n(f;\,\overrightarrow{\lambda}) \right|^2}{S_n(f;\,\overrightarrow{\lambda})} \right)$$

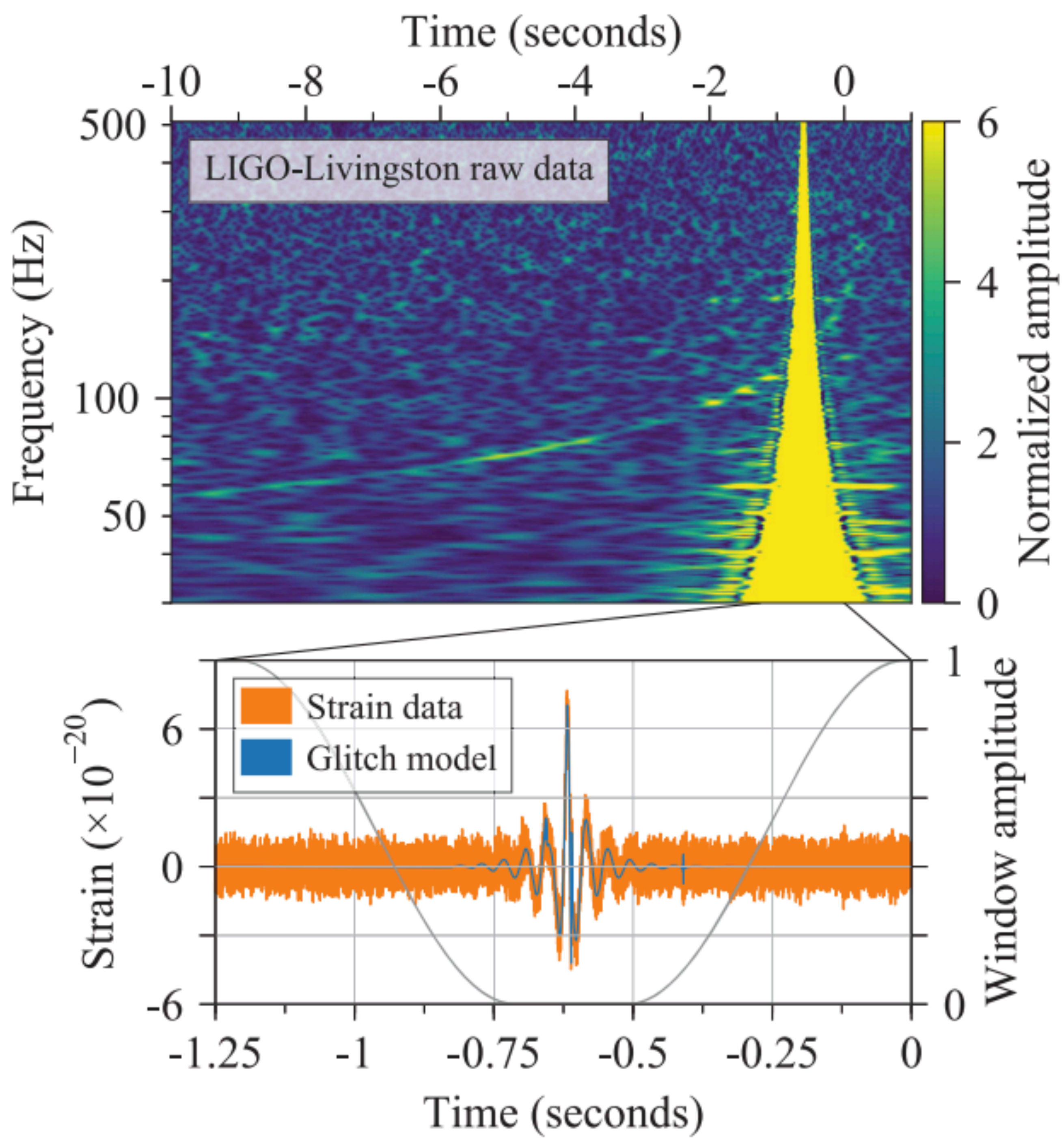
Noise model: Gaussianity

- Include glitch in the likelihood

or

- remove it from the data

$$\mathbf{d} = \mathbf{n} + R\left[\mathbf{h}(\vec{\lambda})\right] + g$$



[LIGO-Virgo Collaboration, 2017]

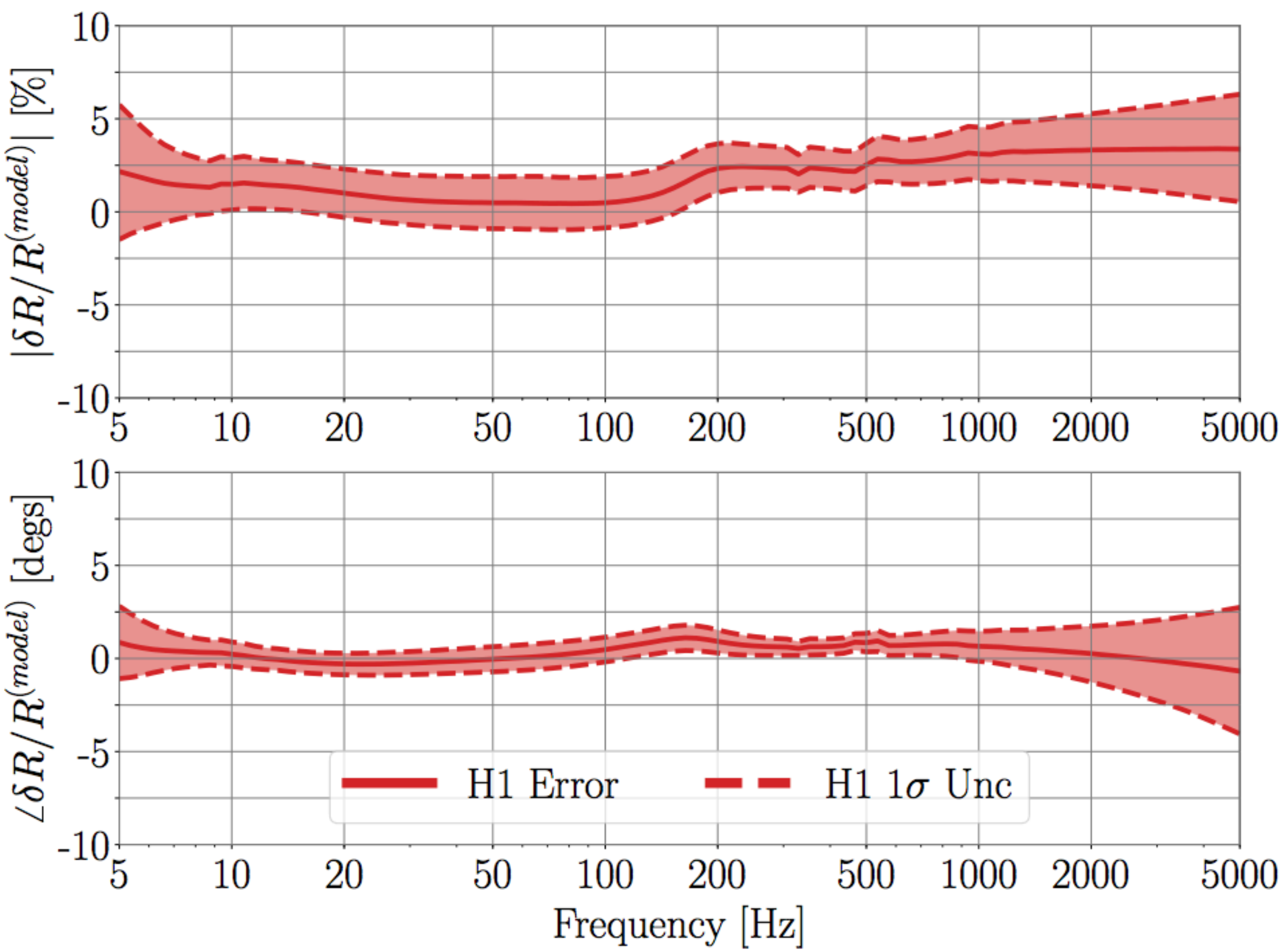
Likelihood

- In practice, we use:

$$p(\mathbf{d}\,|\,\vec{\lambda}, M) \approx \exp\left(-2\sum_{n=1}^{N_{det}}\int_{f_{low}}^{f_{high}} df\,\frac{\left|\mathbf{d}_n(f) - R_n\left[h(f;\,\vec{\lambda}), f;\,\vec{\lambda}\right] - g_n(f;\,\vec{\lambda})\right|^2}{S_n(f;\,\vec{\lambda})}\right)$$

Noise model: Calibration

- Interpolate with cubic splines and marginalise over the calibration error.



$$h' \to h'(1 + \delta A)e^{i\delta\phi}$$

$$\delta A(f) = p_s(f; \{f_i, \delta A_i\})$$
$$\delta\phi(f) = p_s(f; \{f_i, \delta\phi_i\})$$

[Cahillane et al., 2017]

Parameter Estimation

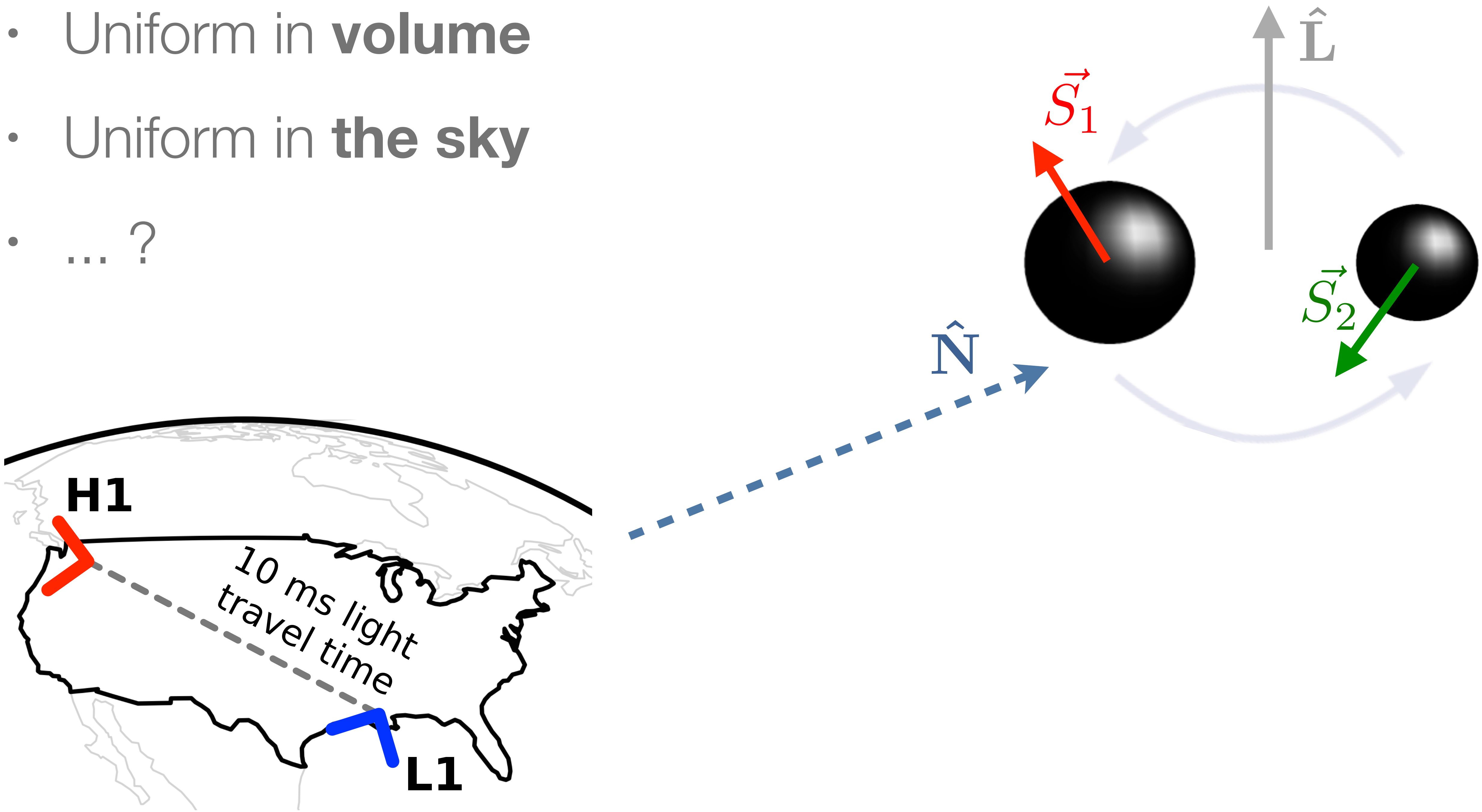
- We want the **posterior** probability of parameters $\vec{\lambda}$, given the data $\vec{x}$. With **Bayes'** theorem:

$$p(\vec{\lambda}\,|\,\mathbf{d}, M) = \frac{p(\vec{\lambda}\,|\,M)\,p(\mathbf{d}\,|\,\vec{\lambda}, M)}{p(\mathbf{d}\,|\,M)}$$

- Fit a **model** to the data (**noise** and **signal** models)
- Build a **likelihood** function
- Specify **prior** knowledge
- **Numerically** estimate the resulting **distribution** (**sampling** algorithms)

Prior: Compact Binary Coalescence

- Uniform in **volume**
- Uniform in **the sky**
- ... ?



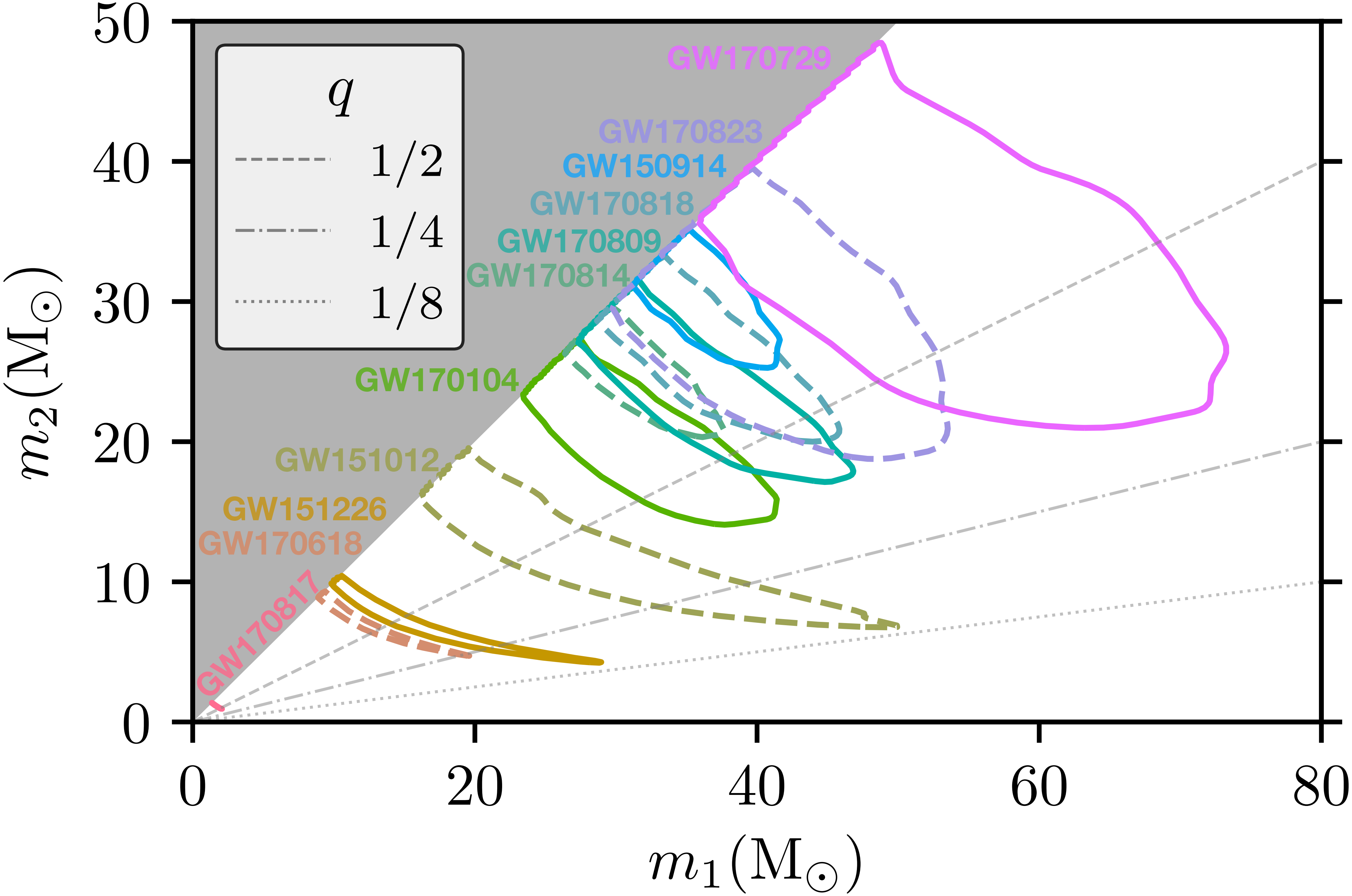
Parameter Estimation

- We want the **posterior** probability of parameters $\vec{\lambda}$, given the data $\vec{x}$. With **Bayes'** theorem:

$$p(\overrightarrow{\lambda}\,|\,\mathbf{d}, M) = \frac{p(\overrightarrow{\lambda}\,|\,M)\,p(\mathbf{d}\,|\,\overrightarrow{\lambda}, M)}{p(\mathbf{d}\,|\,M)}$$

- Fit a **model** to the data (**noise** and **signal** models)
- Build a **likelihood** function
- Specify **prior** knowledge
- **Numerically** estimate the resulting **distribution** (**sampling** algorithms)

Gravitational-wave observations



[LIGO-Virgo Collaboration, 2018]