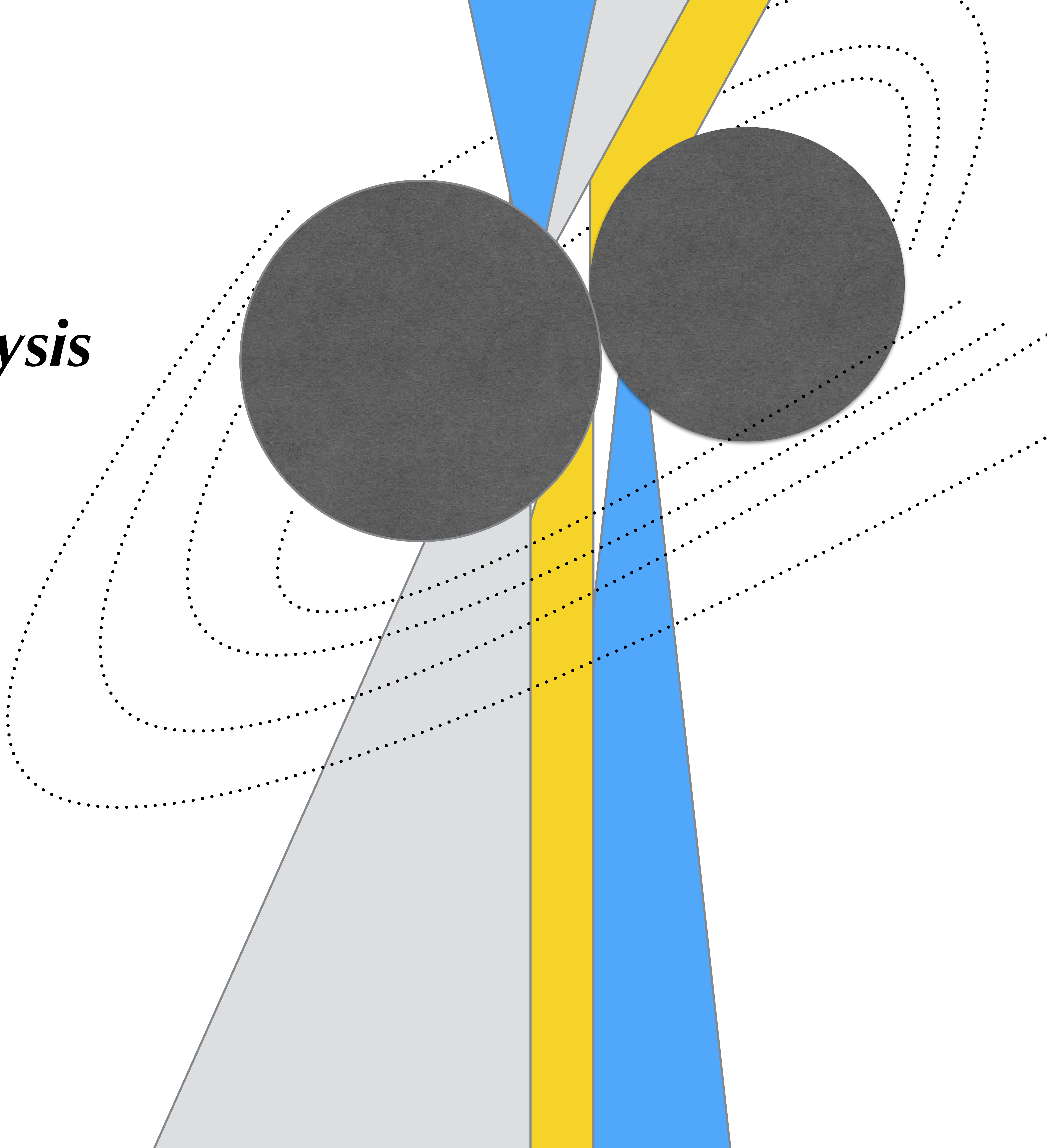
Deep learning for LISA data analysis



Natalia Korsakova
SYRTE/Observatoire de Paris

LASER

INTERFEROMETER

SPACE

ANTENNA

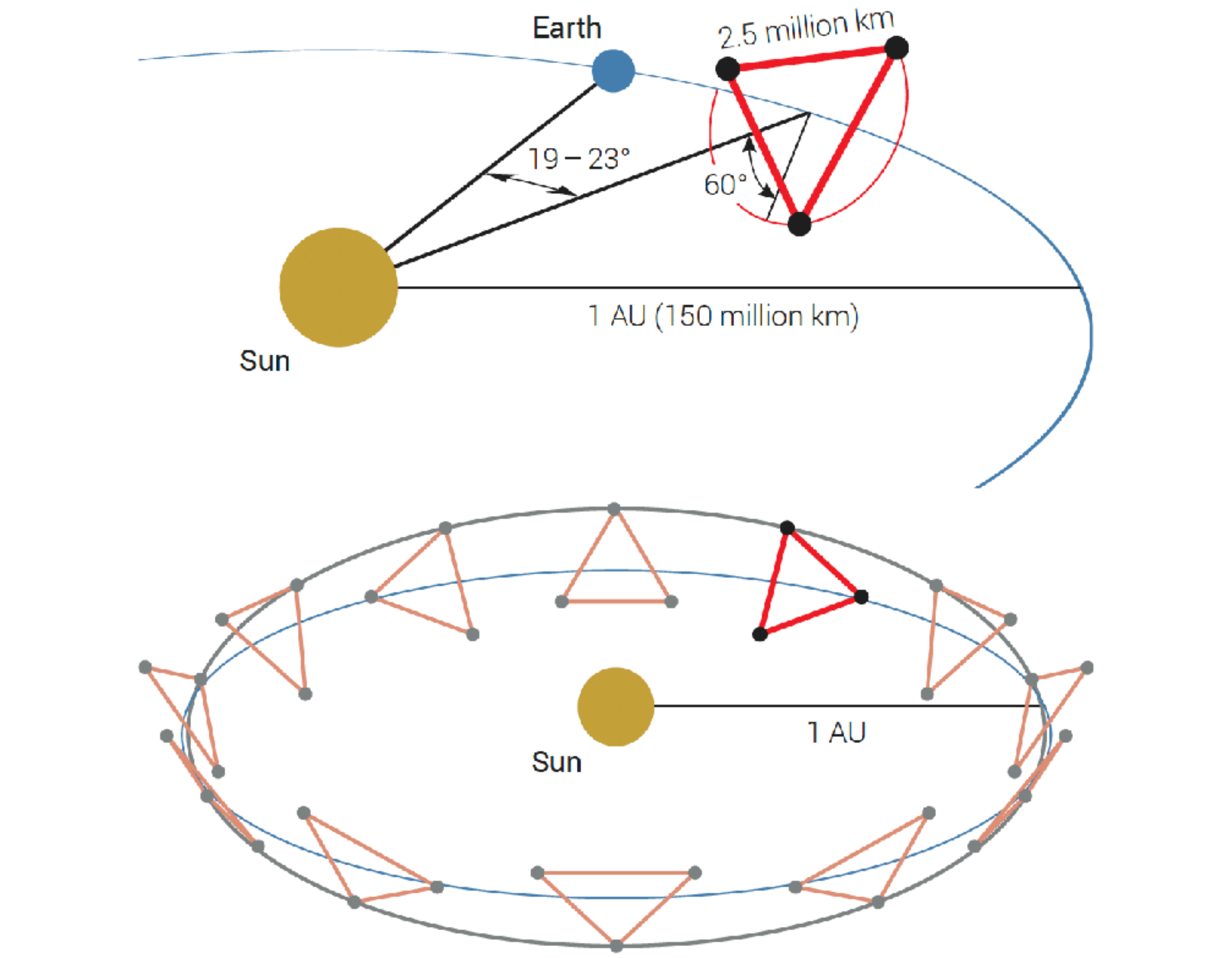


Image: LISA White paper

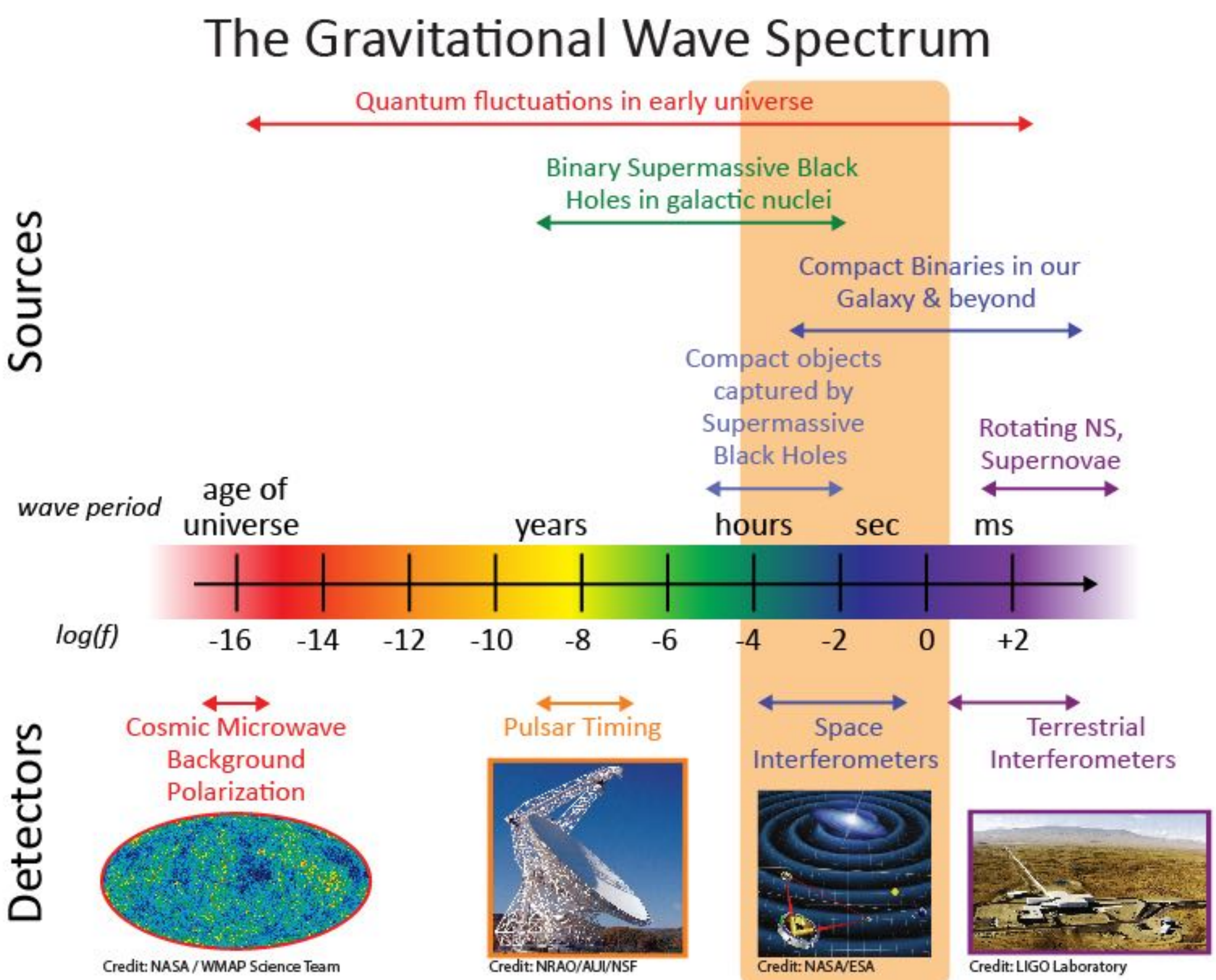
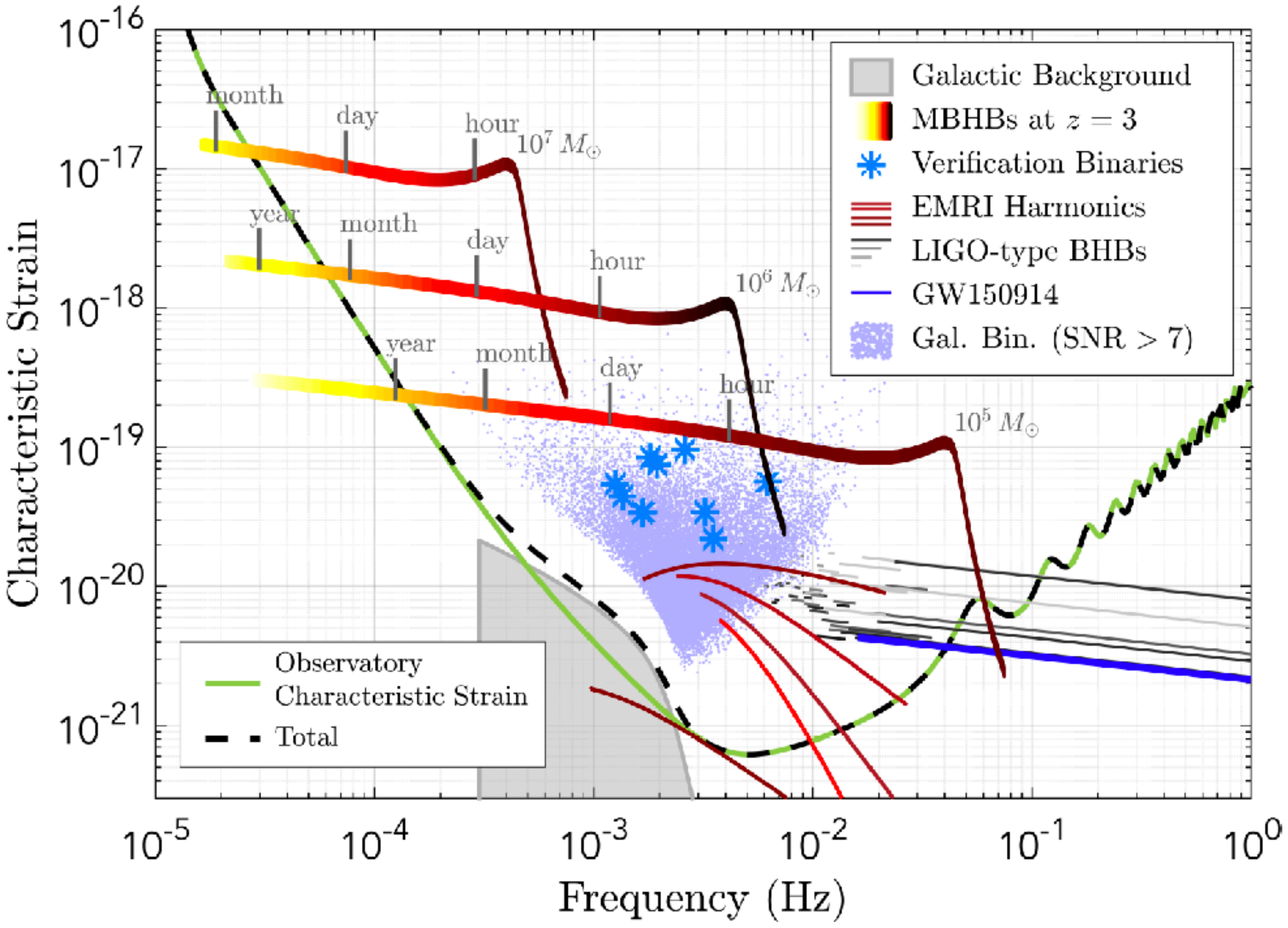

The Gravitational Wave Spectrum
Sources
Quantum fluctuations in early universe
Binary Supermassive Black Holes in galactic nuclei
Compact Binaries in our Galaxy & beyond
Compact objects captured by Supermassive Black Holes
Rotating NS, Supernovae
wave period
age of universe
years
hours
sec
ms
log(f)
-16
-14
-12
-10
-8
-6
-4
-2
0
+2
Detectors
Cosmic Microwave Background Polarization
Pulsar Timing
Space Interferometers
Terrestrial Interferometers
Credit: NASA / WMAP Science Team
Credit: NRAO/AUI/NSF
Credit: NASA/ESA
Credit: LIGO Laboratory

Image: NASA

LISA noise and sources

Signals observed:

- Massive Black Hole Binary
- Galactic Compact Binaries
- Extreme Mass Ratio Inspirals
- Stellar Origin Black Hole Binaries
- Cosmological Background
- …



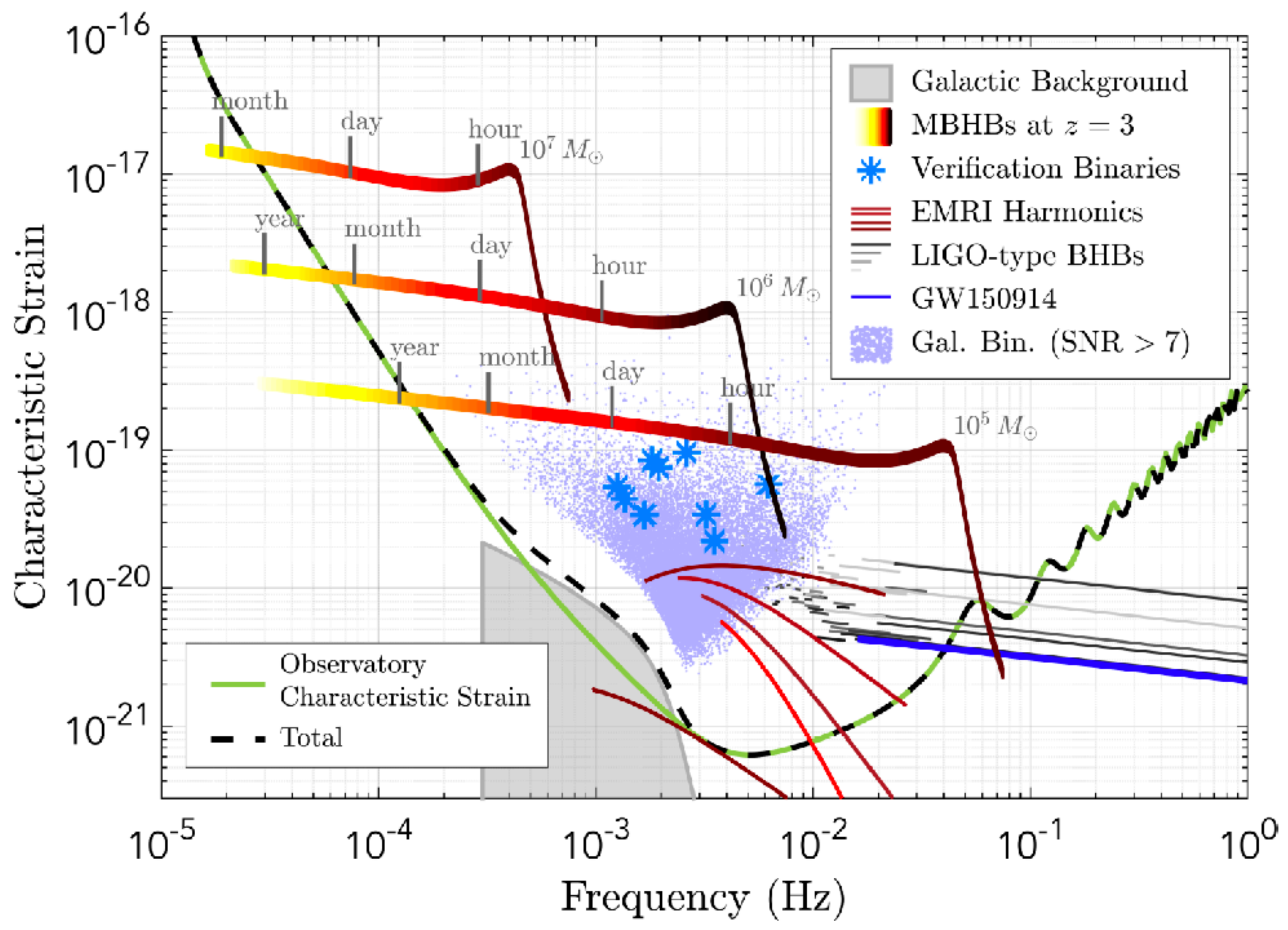
Massive Black Hole Binaries

Signals from MBHB mergers observed by LISA depend on

- assumptions regarding MBH formation,

- the recipes employed for the black hole mass growth via merger and gas accretion

— 10 to 100 sources / year

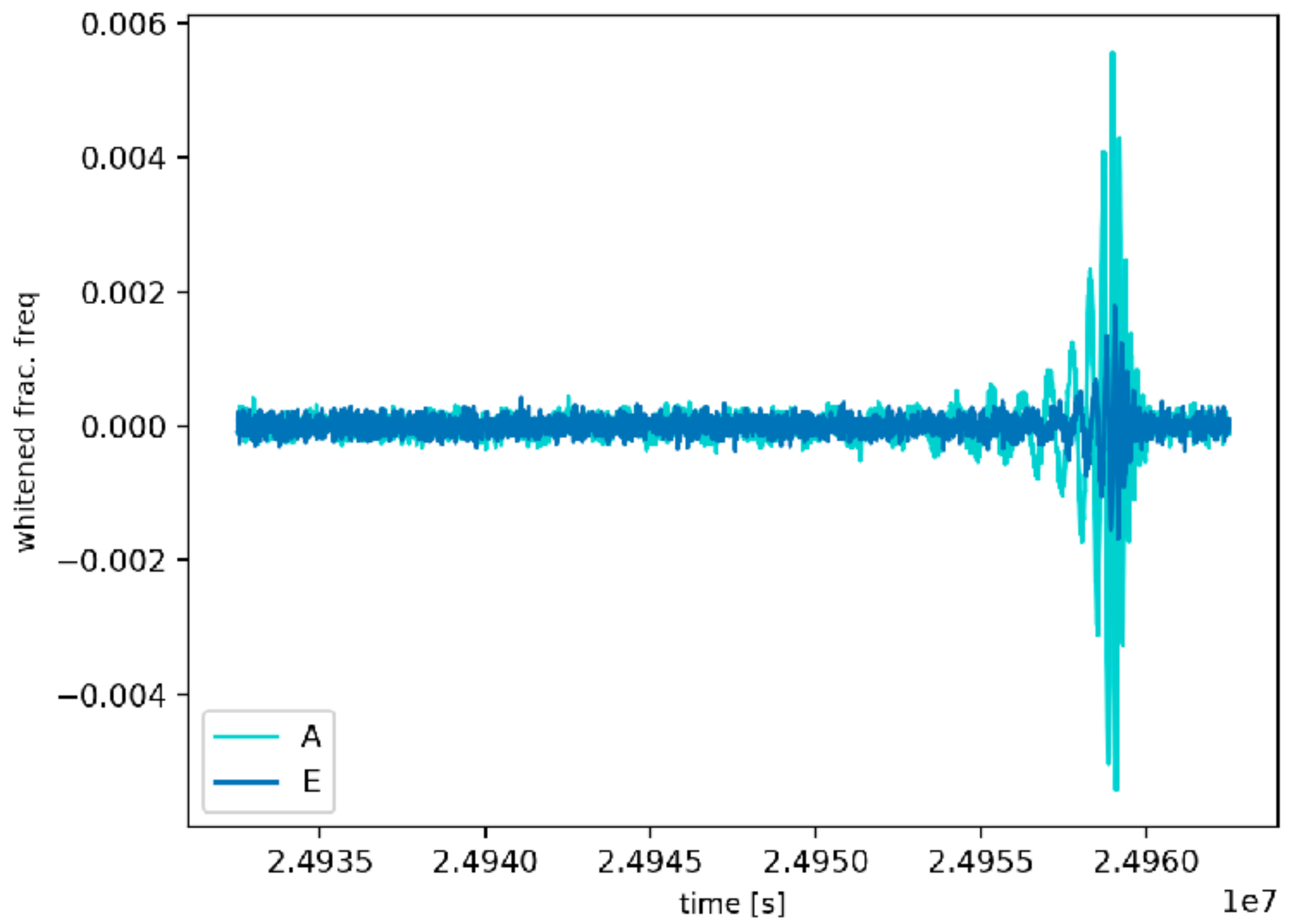


Possible electromagnetic counterparts

Multiple authors suggest that
the electromagnetic counterparts will be observed
as a transient during merger or also during inspiral and
merger.

Electromagnetic counterparts will occur
due to presence of
- matter or
- magnetic fields.

For example:
- Accretion during merger
- Jets produced by the external magnetic fields
- …



Inference

By inference we mean computing the ***posterior distribution*** *of the parameter given the observed data:*

$$p(\mathbf{z}|\mathbf{x}) = \frac{p(\mathbf{x},\mathbf{z})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p(\mathbf{x})}$$

Likelihood function

The problem is that we have to compute marginal likelihood for the observation:

$$p(\mathbf{x}) = \int p(\mathbf{x},\mathbf{z})\mathrm{d}\mathbf{z}$$

Parameter estimation

It is not possible to perform exact inference for the general problem.
We have to introduce some simplifications.

We can use approximate inference:

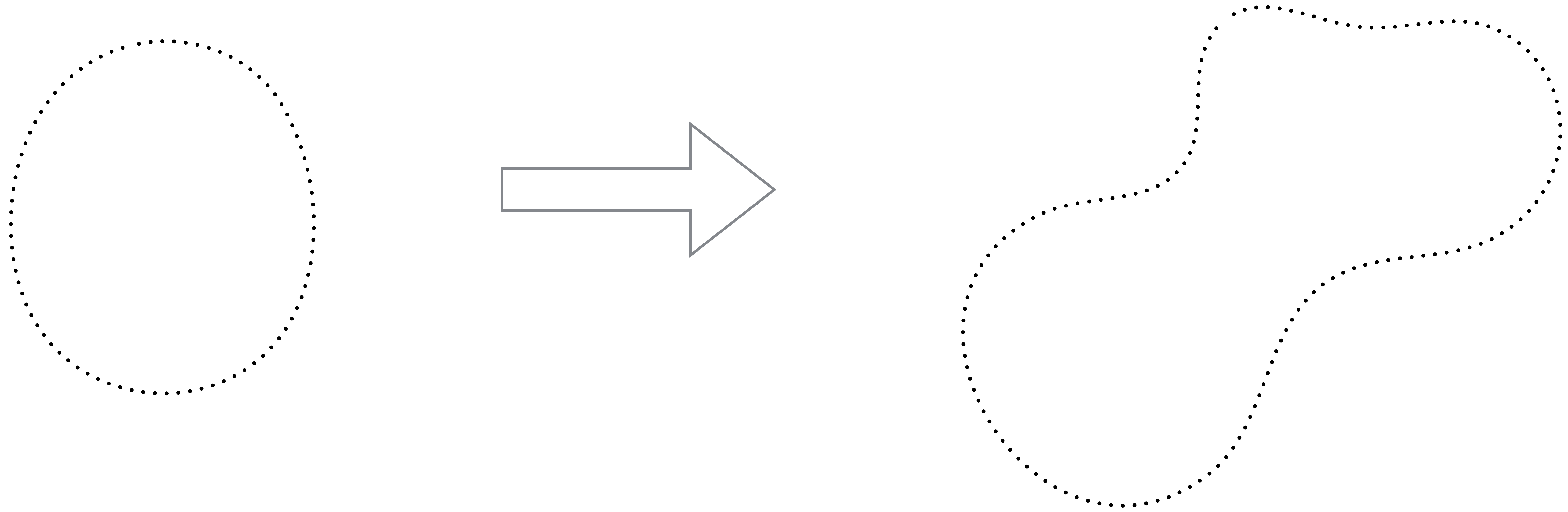
- *Markov Chain Monte Carlo/Nested Sampling: sample from the exact posterior*
- ***Variational Inference:*** *approximate the posterior distribution with a tractable distribution*

There are some exceptions for the models with some simplifications:

- *Gaussian mixture models*
- ***Invertible models***

Learning a map

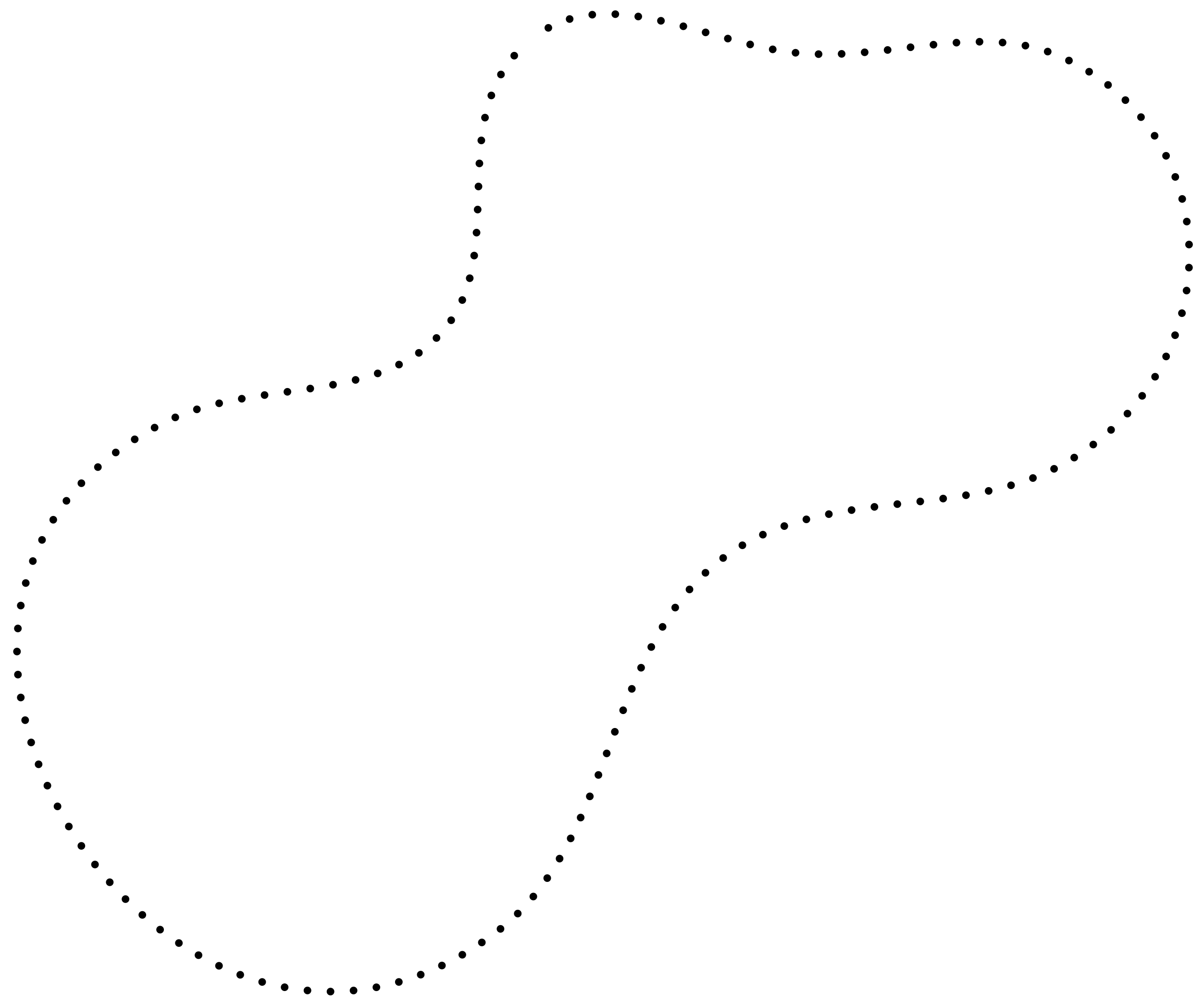
We want to make a deterministic map from the simple and easy to sample distribution to a complex one



Learning a map

The variable transformed with the mapping

$$\mathbf{x} \sim p(x)$$



target distribution

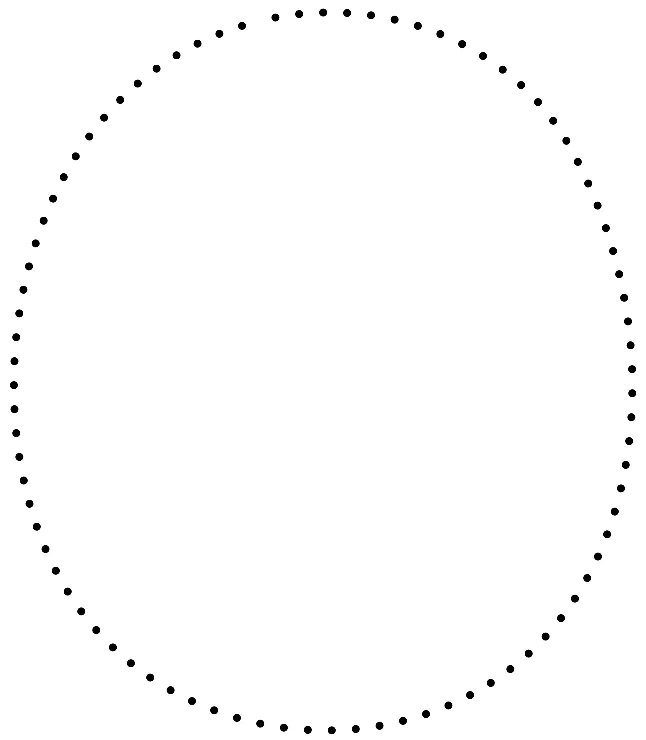
Learning a map

We take a random variable

$$\mathbf{z} \sim q(z)$$

For example:

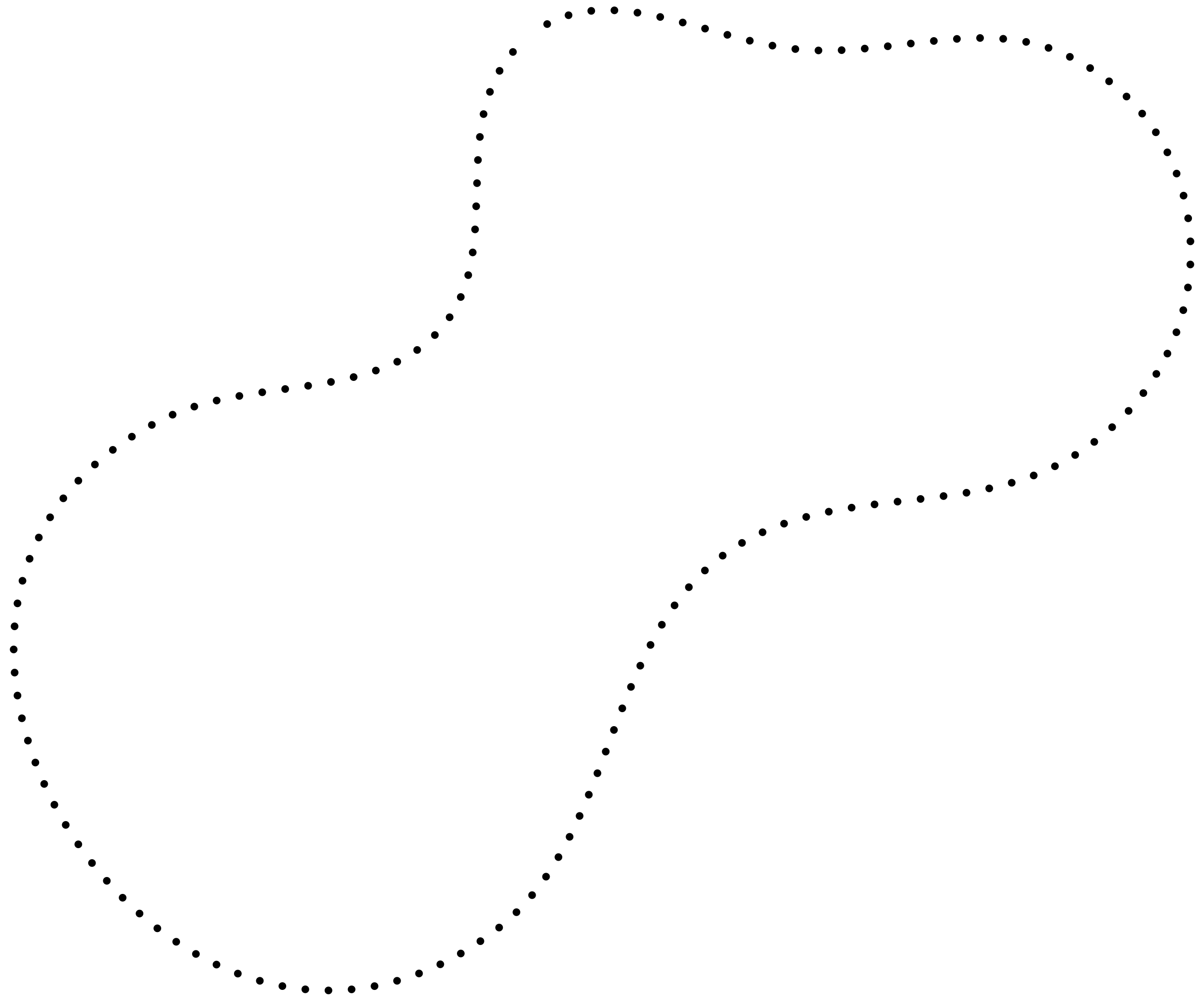
$$\mathbf{z} \sim N(0, I)$$



base distribution

The variable transformed with the mapping

$$\mathbf{x} \sim p(x)$$



target distribution

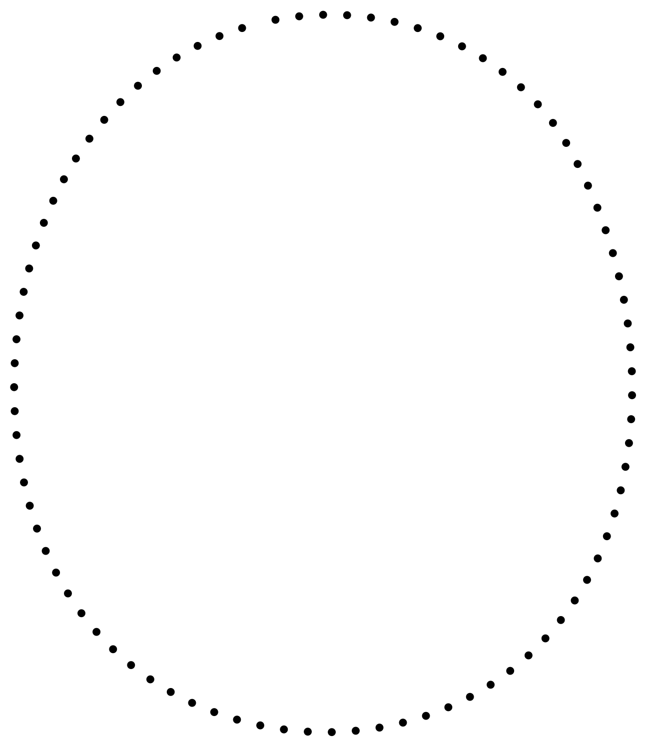
Learning a map

We take a random variable

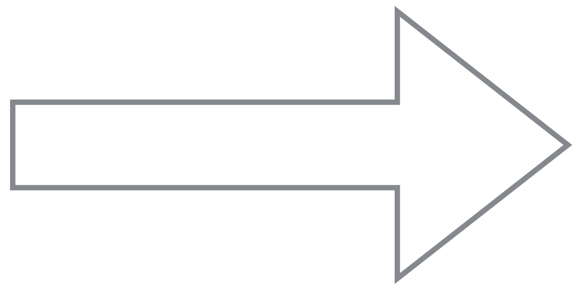
$$\mathbf{z} \sim q(z)$$

For example:

$$\mathbf{z} \sim N(0, I)$$



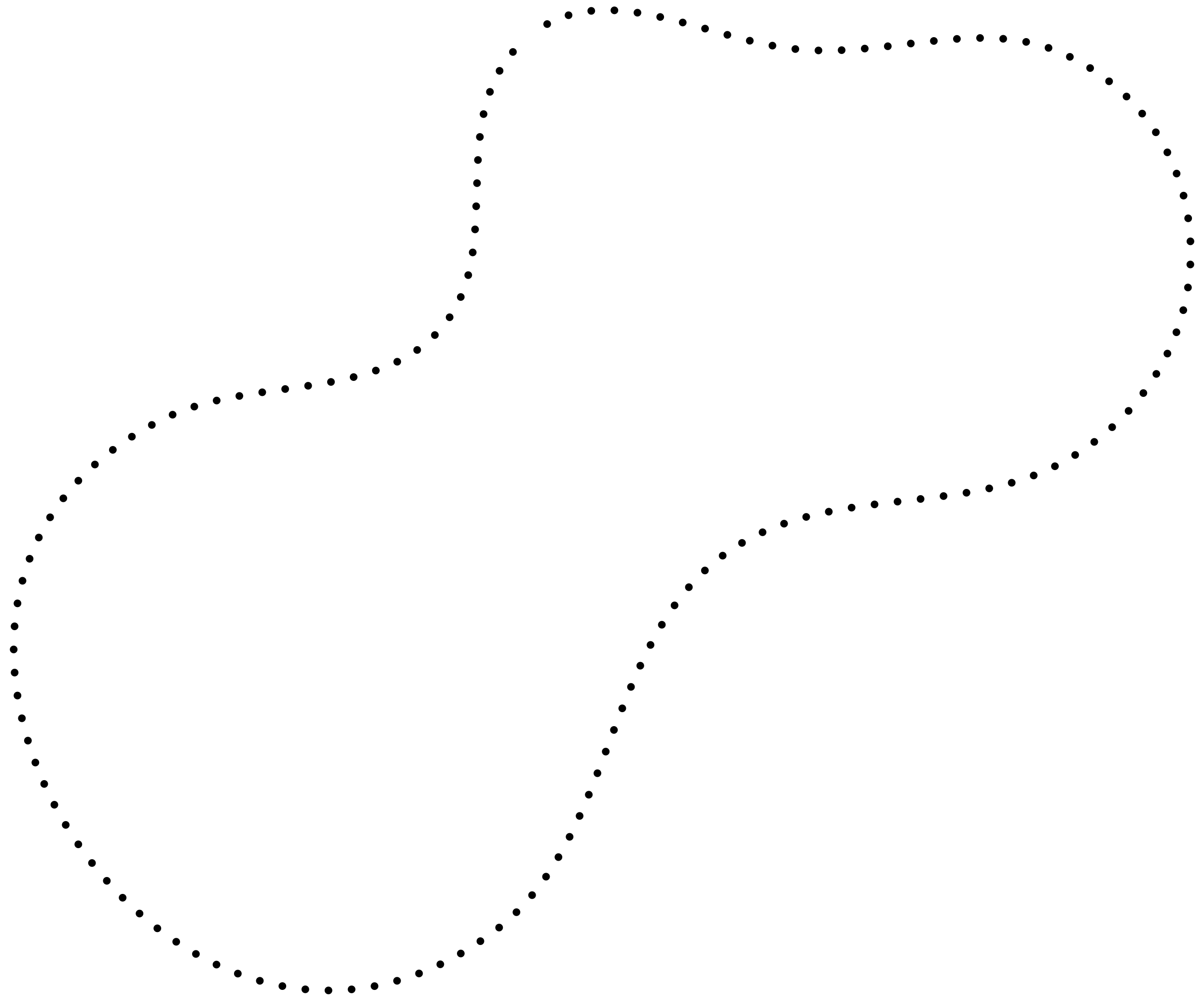
base distribution



$$\mathbf{x} = g(\mathbf{z})$$

The variable transformed with the mapping

$$\mathbf{x} \sim p(x)$$



target distribution

Learning a map

We take a random variable

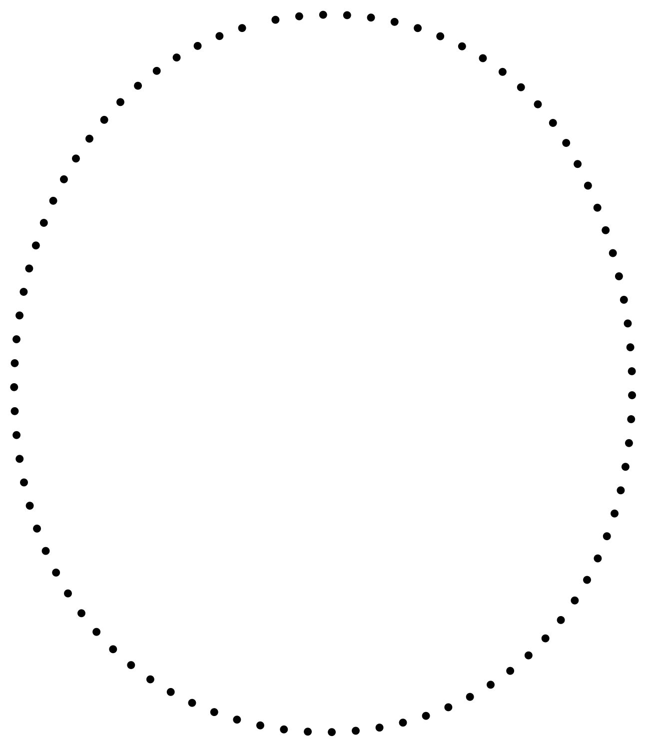
$$\mathbf{z} \sim q(z)$$

The variable transformed with the mapping

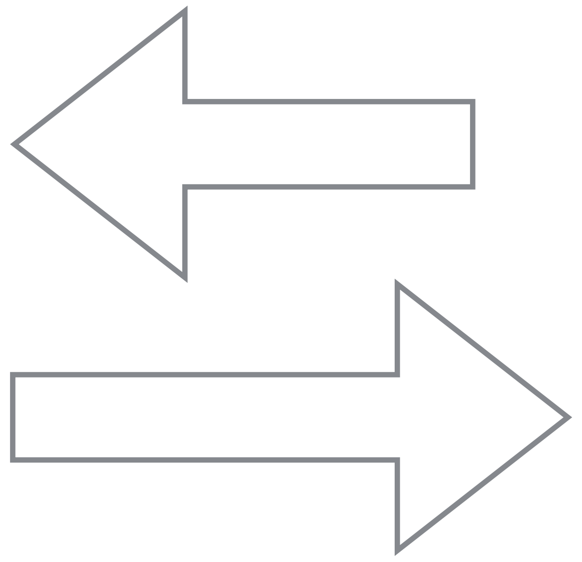
$$\mathbf{z} = f(\mathbf{x}) \qquad \mathbf{x} \sim p(x)$$

For example:

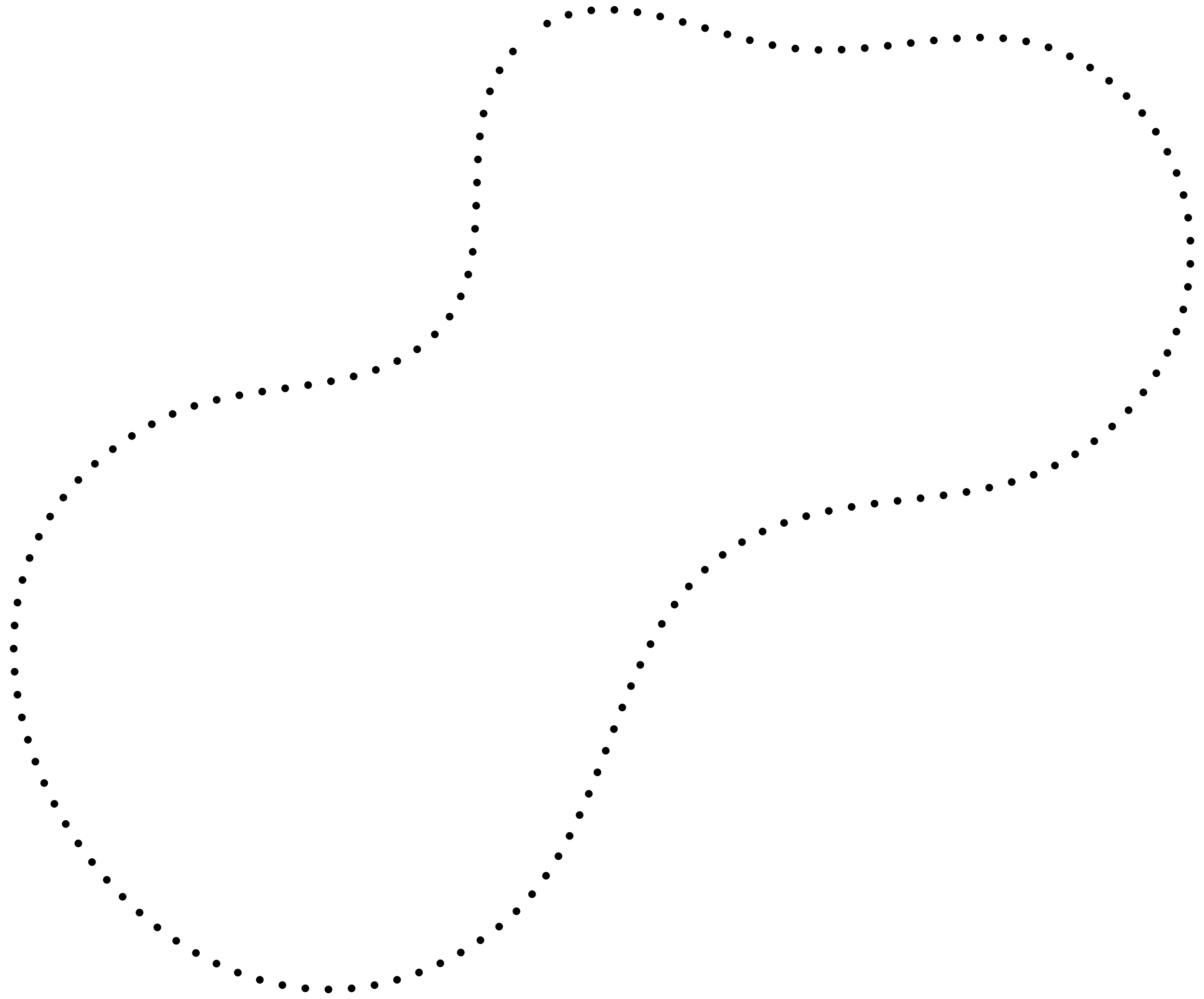
$$\mathbf{z} \sim N(0, I)$$



base distribution



$$\mathbf{x} = g(\mathbf{z}) = f^{-1}(\mathbf{z})$$



target distribution

Learning a map

How to estimate the map?

$$\mathbf{x} = f^{-1}(\mathbf{z})$$

Learning a map

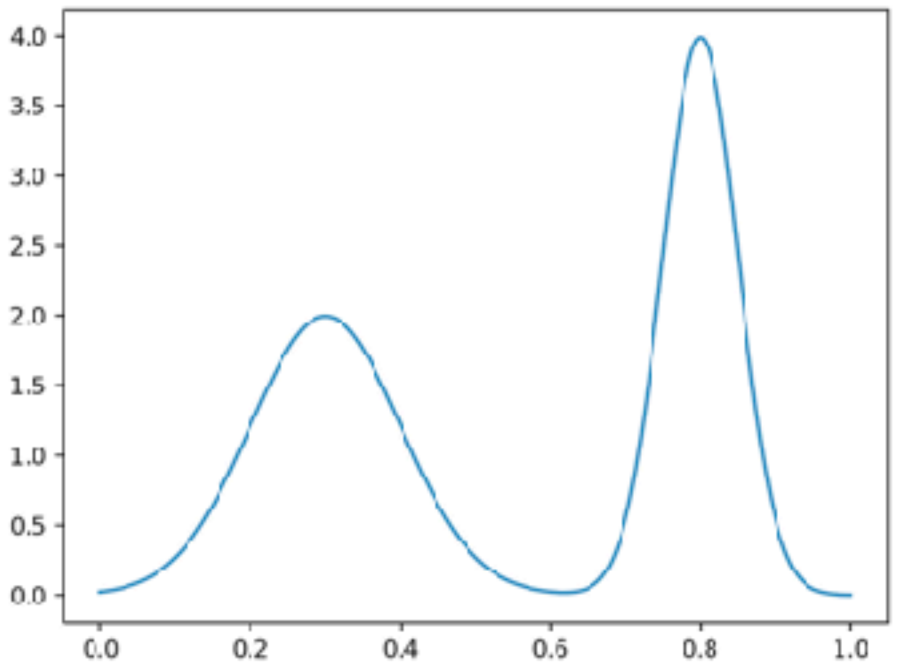
How to estimate the map?

$$\mathbf{x} = f^{-1}(\mathbf{z})$$

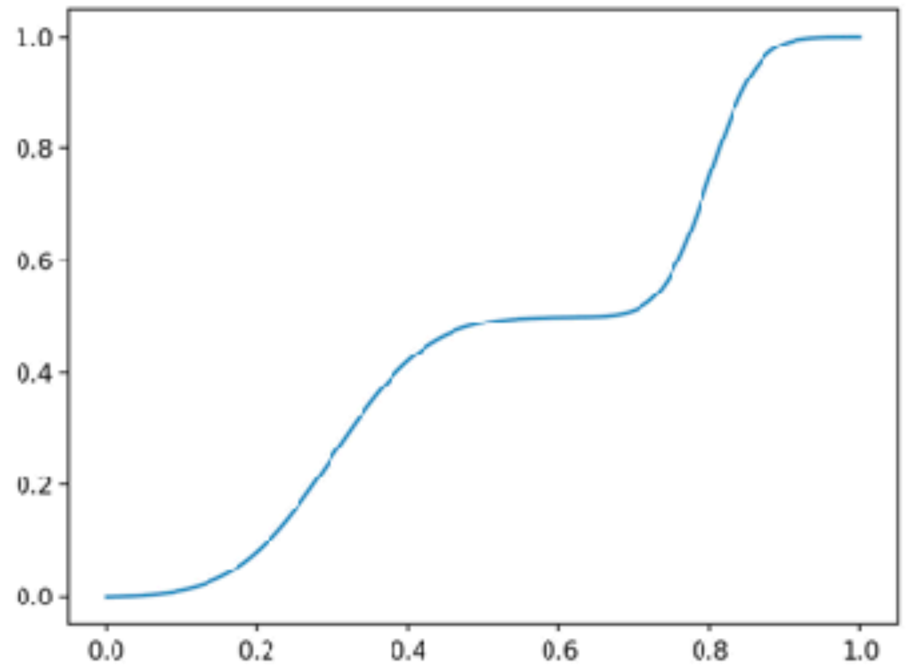
<u>Easy in 1D case:</u>

$$z \sim \text{Uniform}([0, 1])$$

$$x = f_\theta^{-1}(z)$$



$p_\theta(x)$



$f_\theta(x) = \int_{-\infty}^{x} p_\theta(t)\, dt$

$$p(x)\mathrm{d}x = q(z)\mathrm{d}z$$

Learning a map

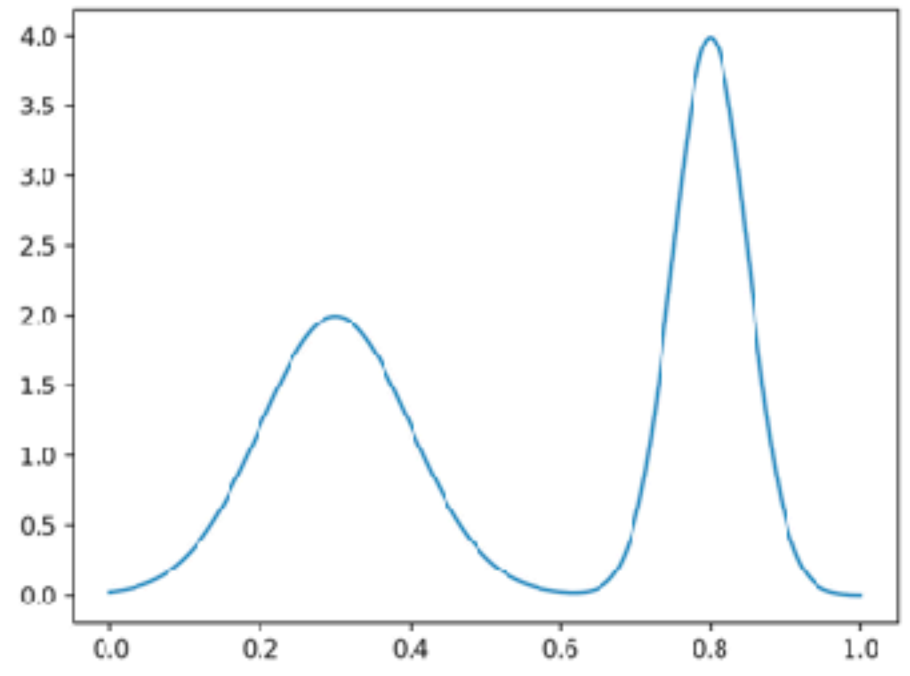
How to estimate the map?

$$\mathbf{x} = f^{-1}(\mathbf{z})$$

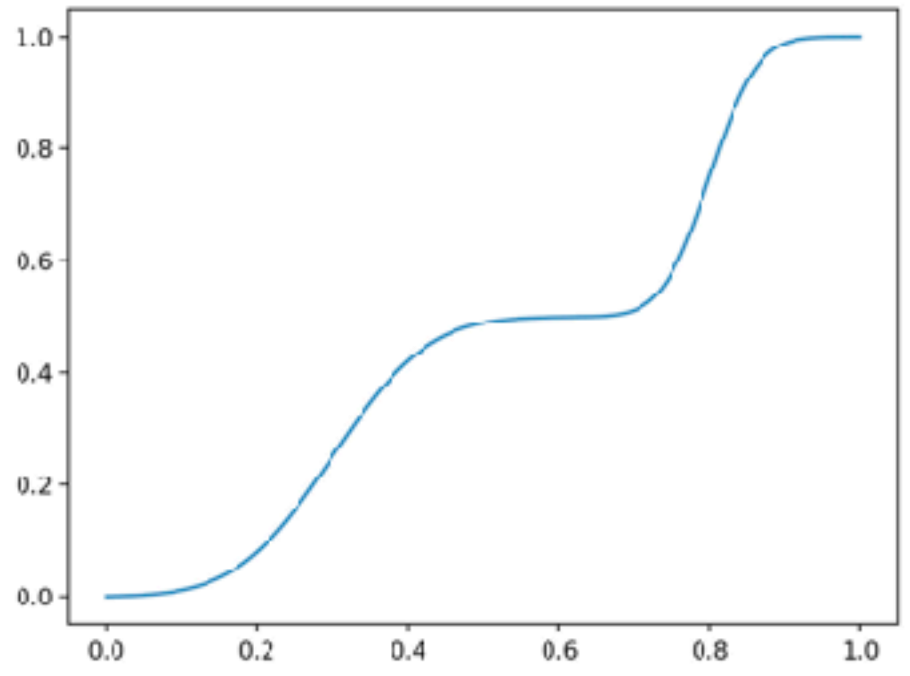
Easy in 1D case:

$$z \sim \text{Uniform}([0,1])$$

$$x = f_\theta^{-1}(z)$$



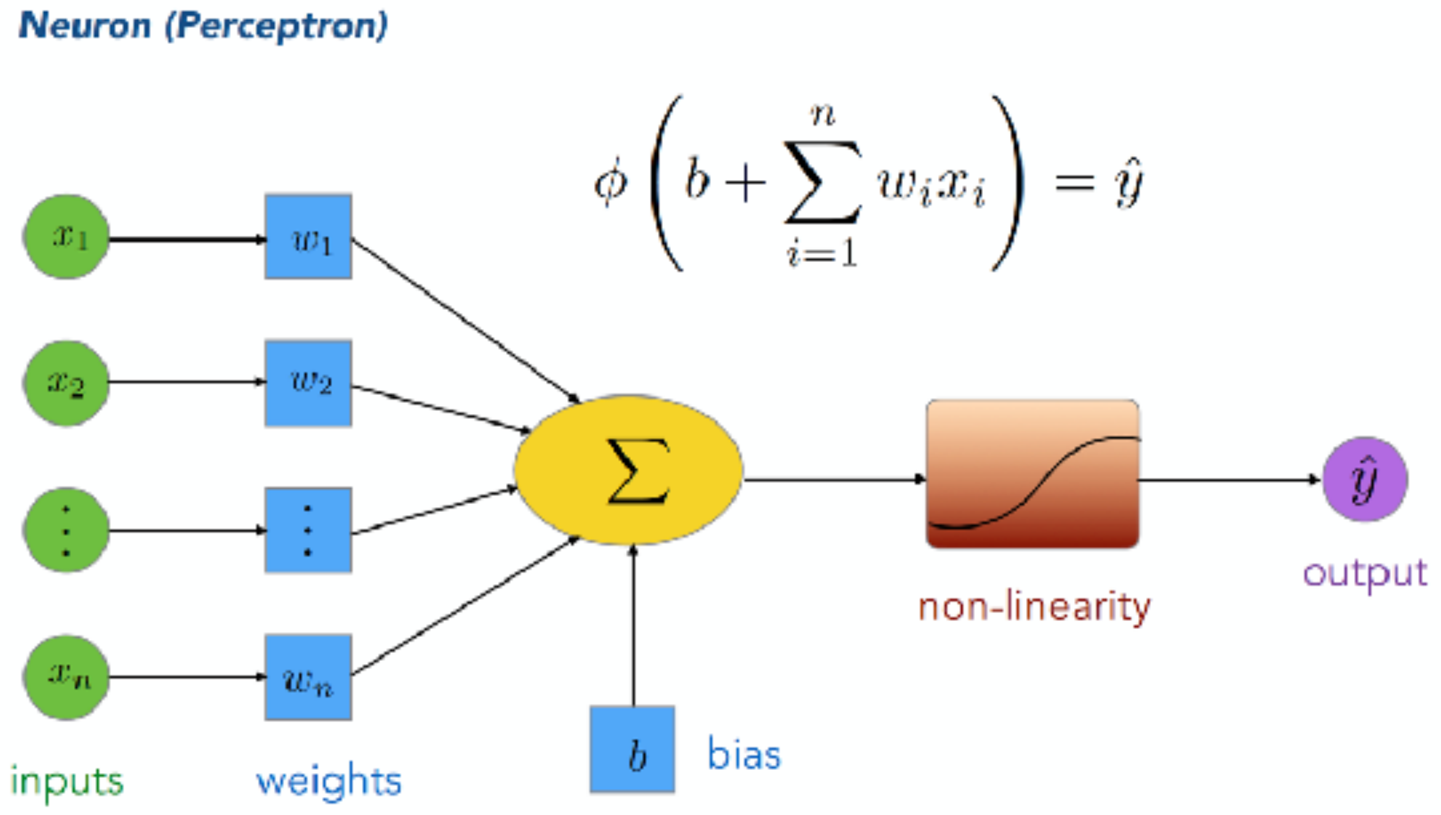
$p_\theta(x)$



$f_\theta(x) = \int_{-\infty}^{x} p_\theta(t)\,dt$

Multidimensional case:

Parameterise a map by the Neural Network

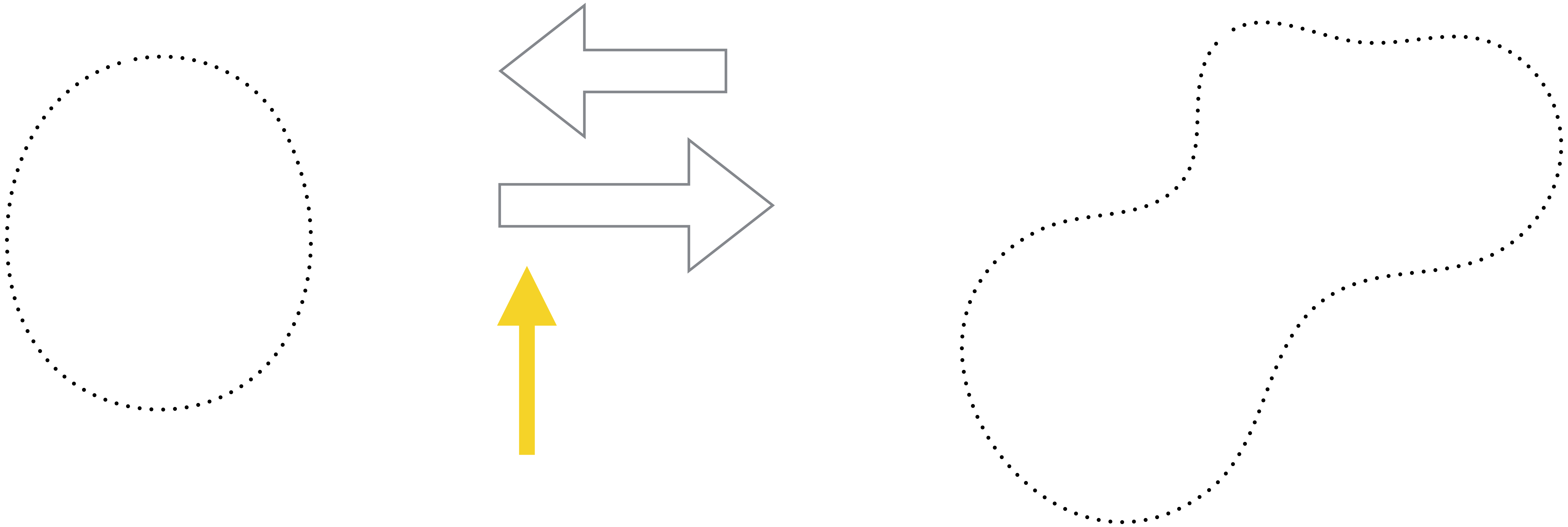


$$p(\mathbf{x}) = q(f(\mathbf{x})) \left| \det \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right|$$

Condition on the Waveform

Samples from a prior of a physical parameter

$$\mathbf{x} \sim p(x)$$



Condition map on the simulated data:

$$\mathbf{d} = h(\mathbf{x}) + \mathbf{n}$$

Therefore we have access to the joint sample: $p(\mathbf{d}, \mathbf{x}) = p(\mathbf{x})p(\mathbf{d}|\mathbf{x})$

Evaluation of Jacobian

$$J_f(\mathbf{z}) = \begin{bmatrix} \frac{\partial f_1}{\partial z_1} & \cdots & \frac{\partial f_1}{\partial z_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial z_1} & \cdots & \frac{\partial f_n}{\partial z_n} \end{bmatrix}$$

The calculation of determinant Jacobian will take $O(n^3)$

To make it faster we have to ensure that the Jacobian is triangular

Because the determinant of the triangular matrix is just a product of the diagonal elements

Affine transformations

location-scale transformation

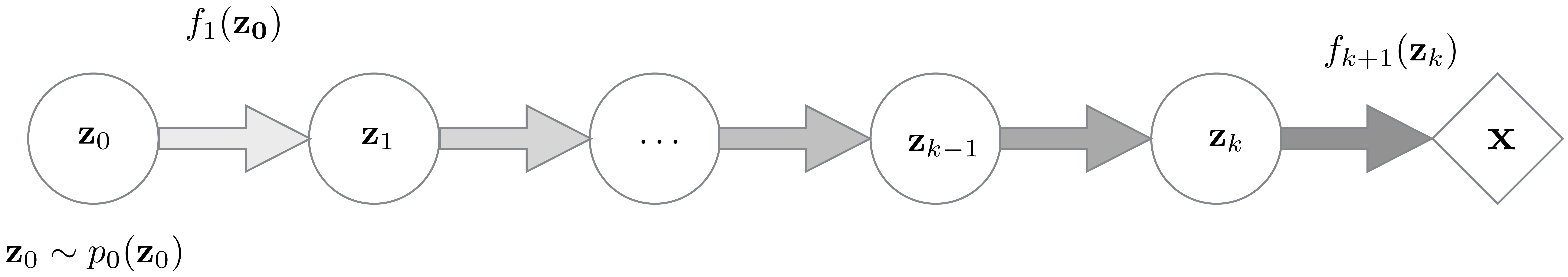
$$\tau(z_i; \mathbf{h}_i) = \alpha_i z_i + \beta_i \qquad \mathbf{h}_i = \{\alpha_i, \beta_i\}$$

Invertibility for $\alpha_i \neq 0$

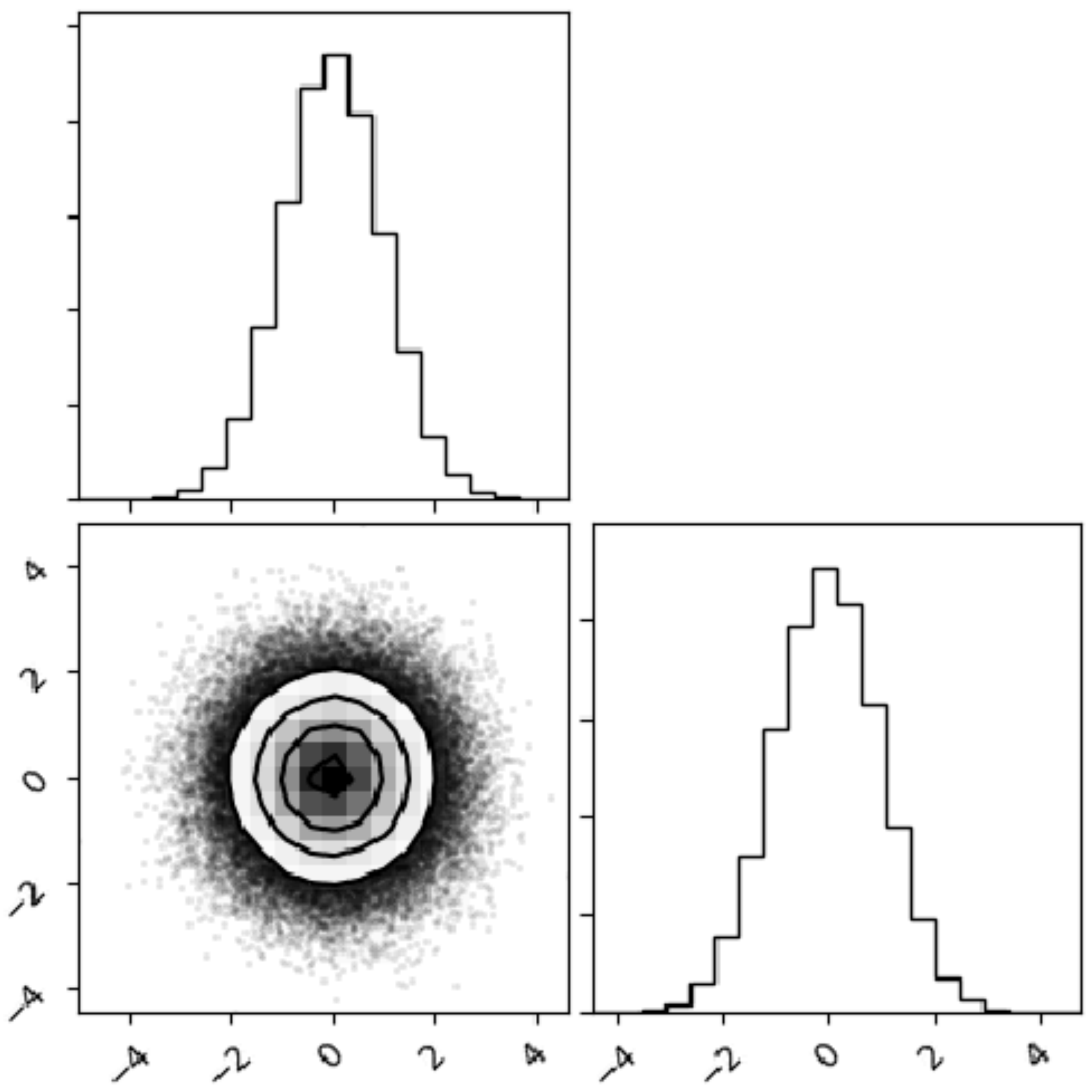
log-Jacobian becomes

$$\log|\det J_f(\mathbf{z})| = \sum_{i=1}^{N} \log|\alpha_i|$$

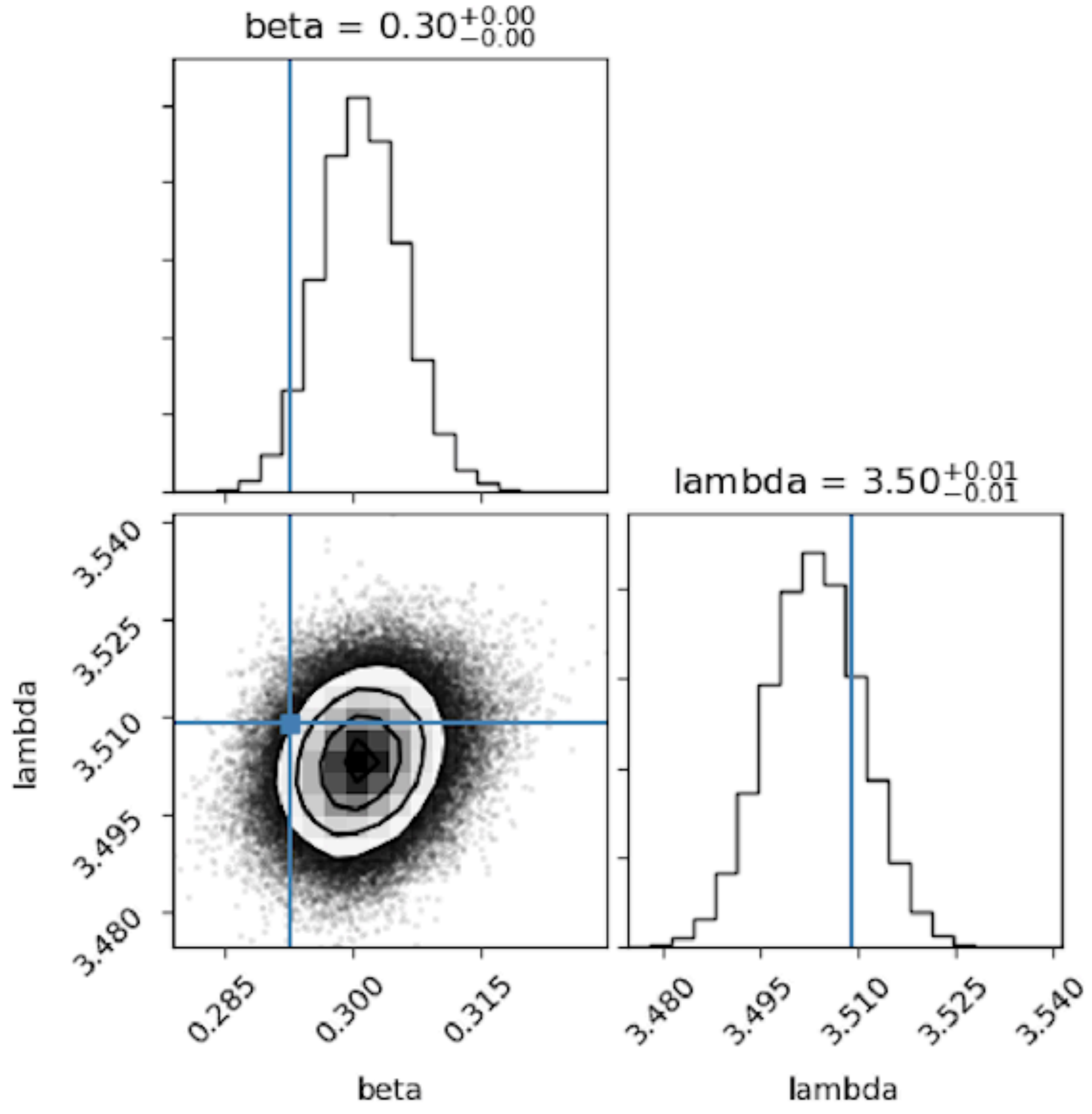
Combining transformations



Example



Learned transformation conditioned on real data (LISA Data Challenge)



Conclusions

New way to do Bayesian Inference for the Gravitational Wave data analysis

Time consuming calculations are done at the training time