MAX PLANCK INSTITUTE
FOR INTELLIGENT SYSTEMS

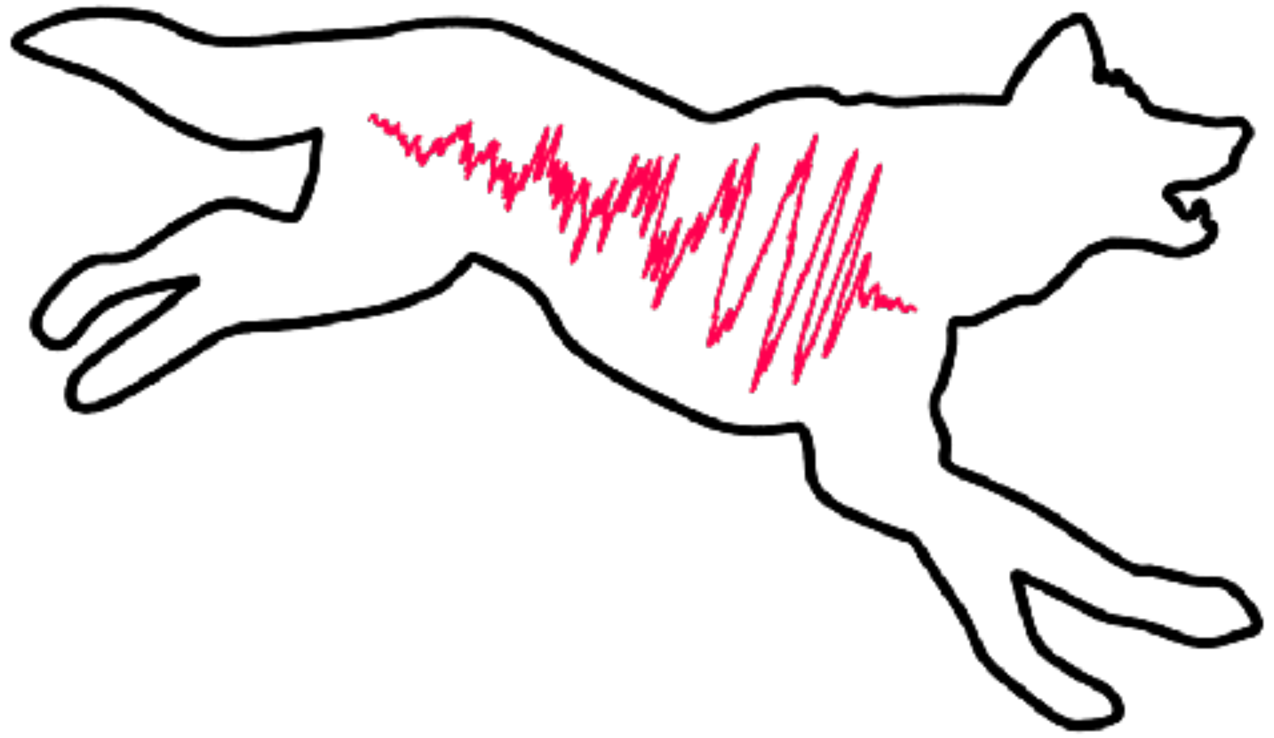
Importance sampling with DINGO

An overview

Annalena Kofler, 10.07.2025

TianQin Seminar

Overview



- How does Neural Posterior Estimation work?
- What is DINGO?
- How does importance sampling work?
- How is importance sampling included in the DINGO library?

Mainly based on these papers:

- Dax+, Real-Time Gravitational Wave Science with Neural Posterior Estimation. PRL 127, 2021
- Dax+, Group Equivariant Neural Posterior Estimation, ICLR 2022
- Dax+, Neural Importance Sampling for Rapid and Reliable Gravitational Wave Inference, PRL 130, 2023

The Dingo Pack


Maximilian Dax


Stephen Green


Annalena Kofler


Nihar Gupte


Michael Pürrer


Alex Roussopoulos


Samuel Clyne


Ashwin Girish


Cecilia Fabbri


Jonas Wildberger


Vincent Berenz


Jonathan Gair


Jakob Macke


Bernhard Schölkopf


Alessandra Buonanno

Posterior Estimation

Why care about the Posterior?

Bayes Formula:

d: Data

θ: Parameters

Likelihood

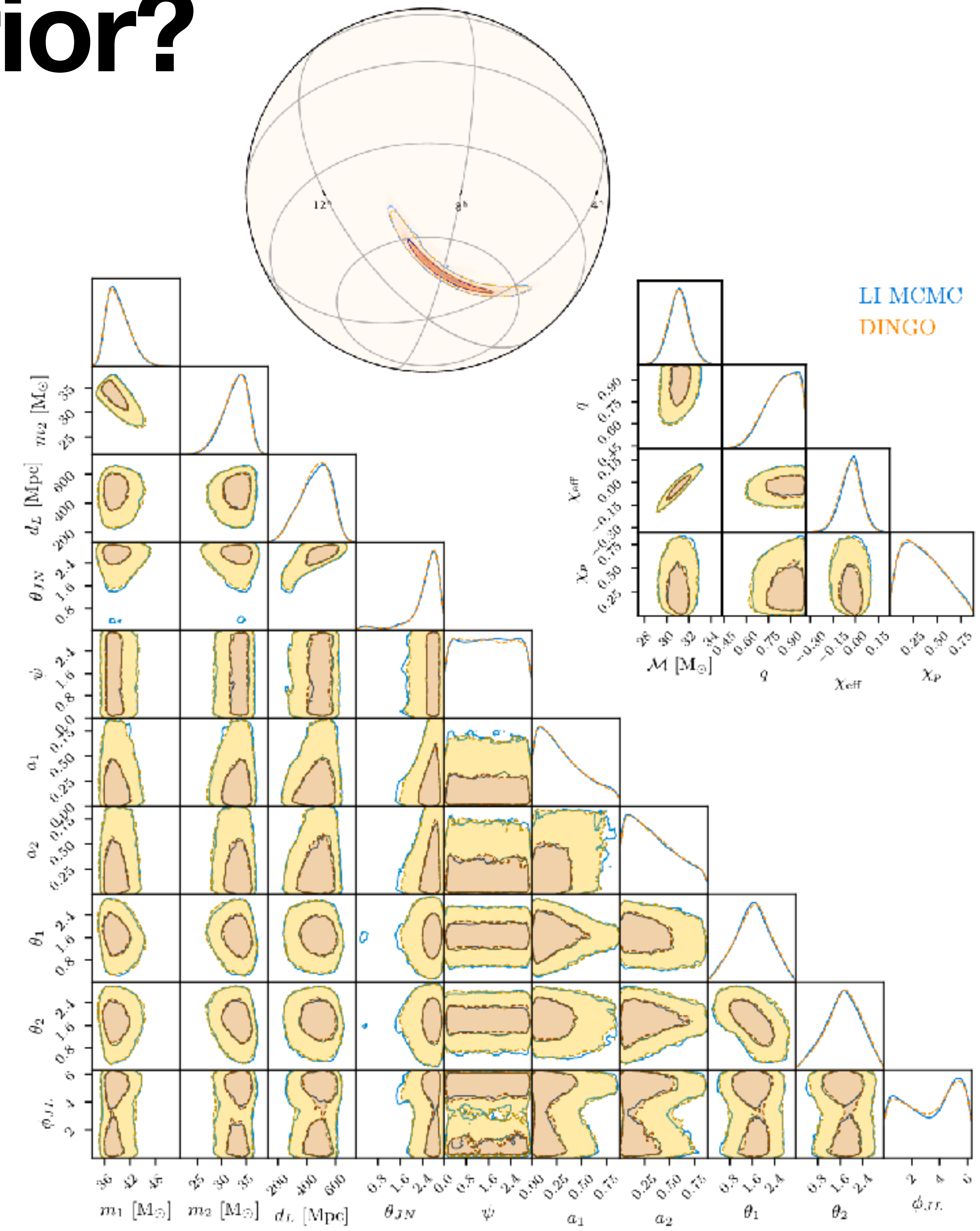
$$p(\theta\,|\,d) = \frac{p(d\,|\,\theta)}{p(d)}\,p(\theta)$$

Posterior

Evidence

Prior

→ allows us to interpret gravitational wave signals



Dax, et al.,Real-time gravitational-wave science with neural posterior estimation, PRL2021

How does Posterior Estimation usually work?

Bayes Formula:

Likelihood

$$p(\theta \,|\, d) = \frac{p(d \,|\, \theta)}{p(d)} \, p(\theta)$$

Posterior Prior

Evidence

d: Data

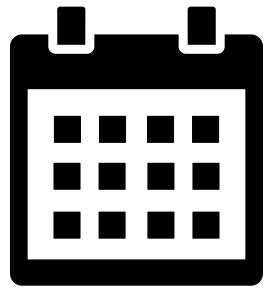
θ: Parameters

Steps:

1. Specify Prior, e.g. $d_l \in [100,\ 1000]$ Mpc
2. Run Monte Carlo Method, e.g. Nested Sampling
   - Simulate waveforms
   - Calculate likelihood
3. Obtain converged posterior

Why is this problematic in the future?

- Standard Bayesian methods: millions - billions of parameter estimations before converging
- Evaluating one event: Minutes - Weeks



- In the future: detector upgrades → more signals to analyze

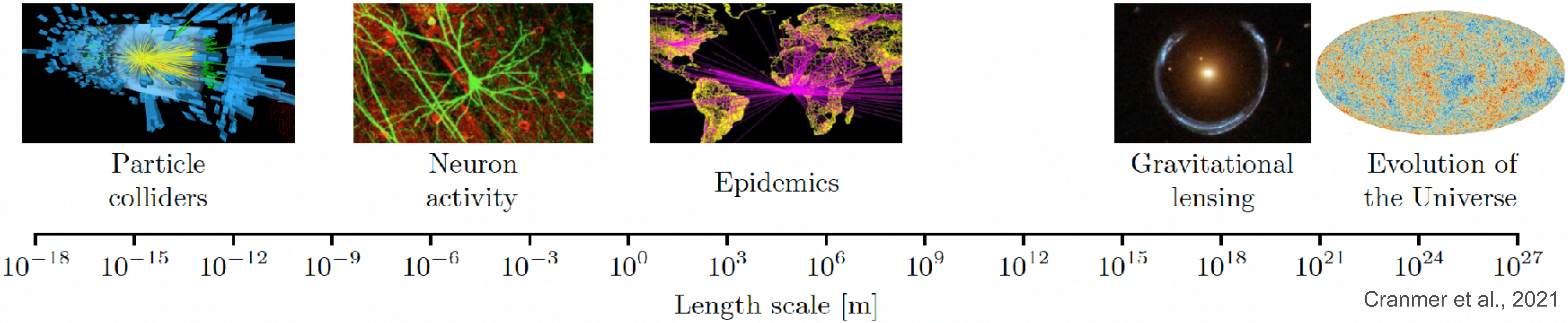
→ Solve problem by using alternative methods such as simulation-based inference

Simulation-Based Inference and Neural Posterior Estimation

What is Simulation-Based Inference?

- SBI = statistical method to obtain information about a distribution based on simulated samples

- Why based on simulations? → We have simulators for a lot of things:

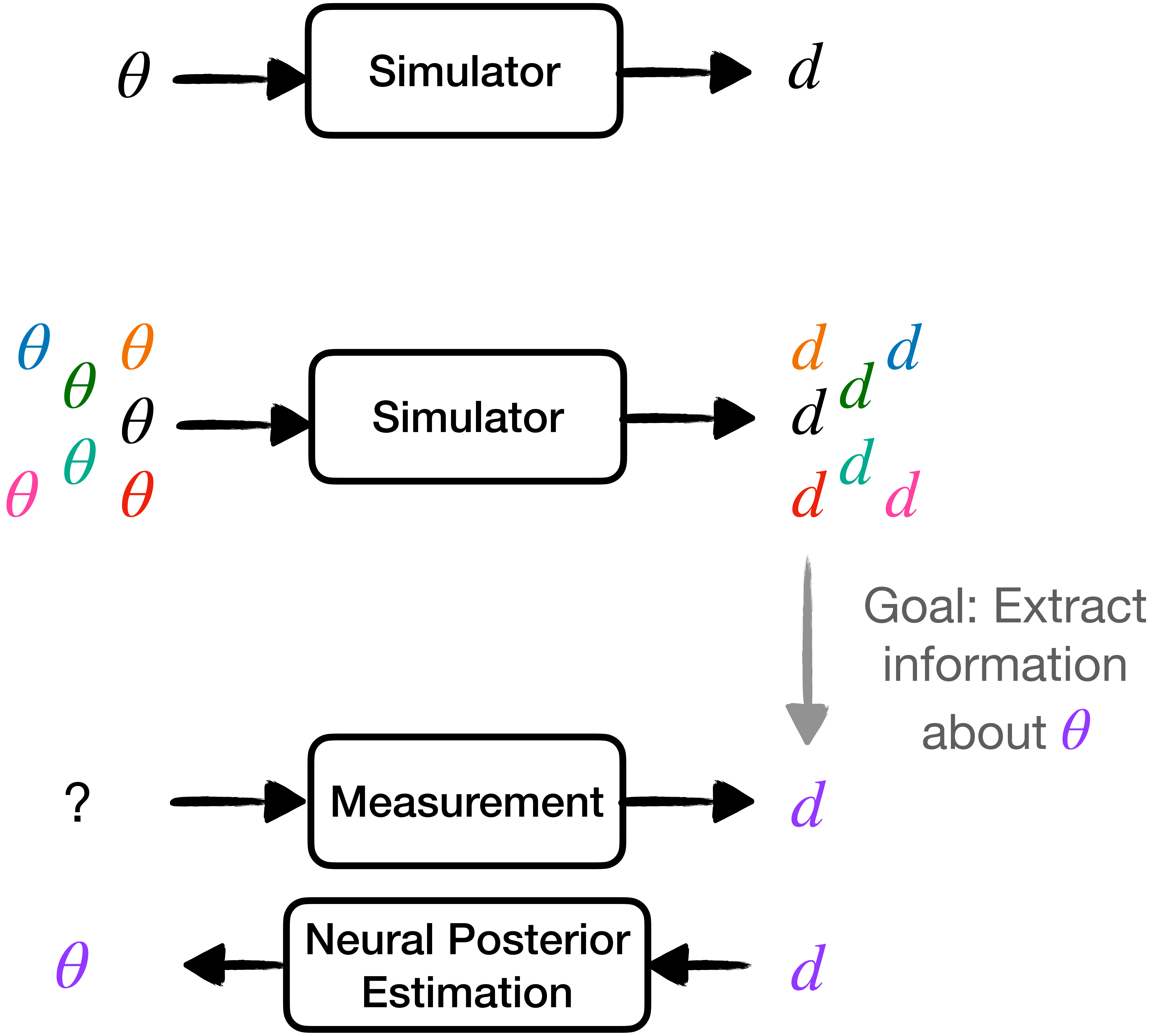


Cranmer et al., 2021

- Especially for gravitational waves!

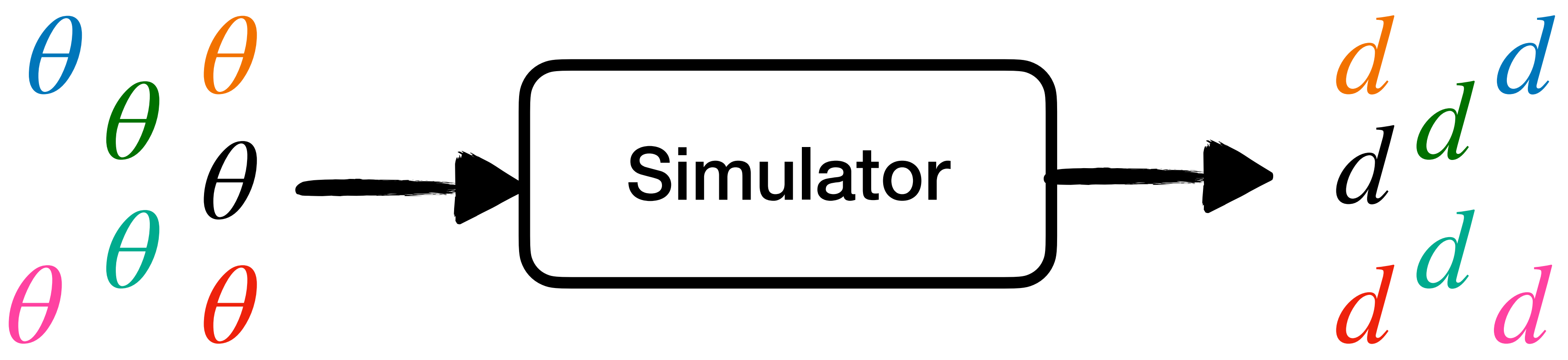
How does SBI work?

- Requirement: Simulator
- Generate dataset



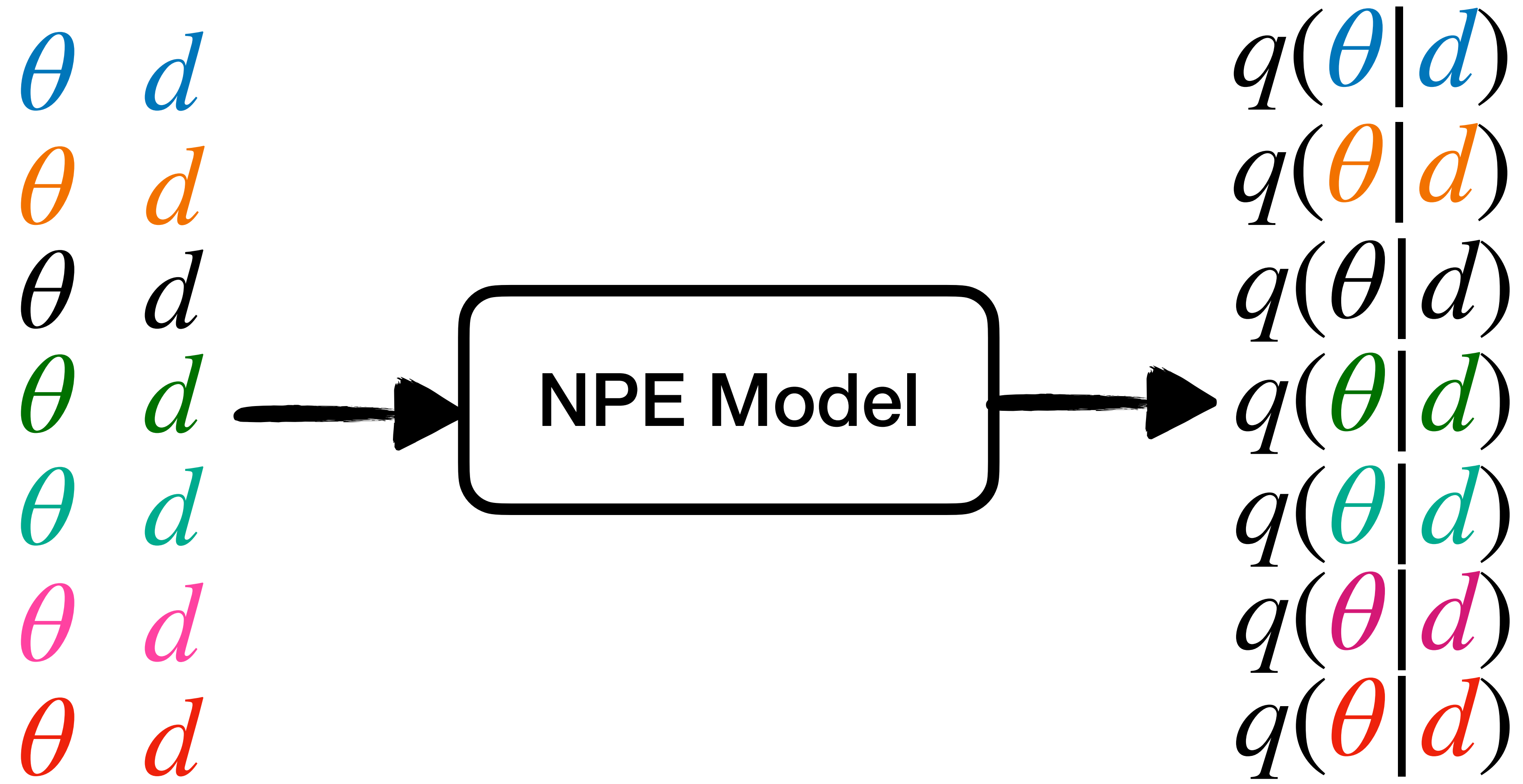
How does Neural Posterior Estimation work?

1. Generate large training data set with simulator

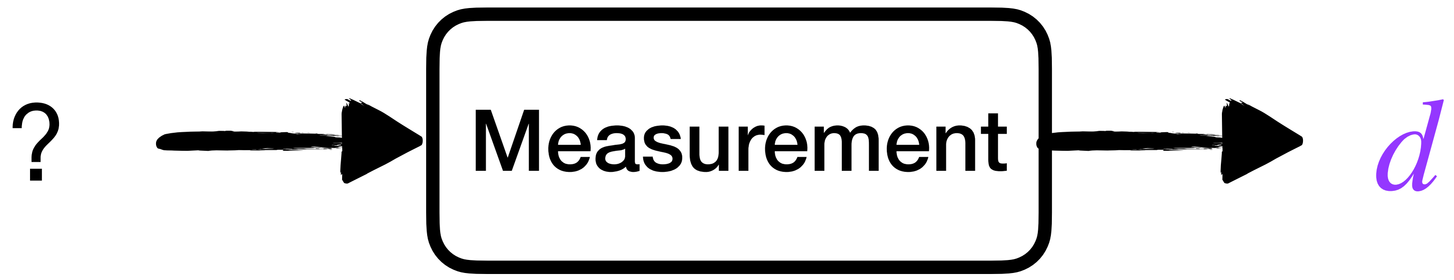


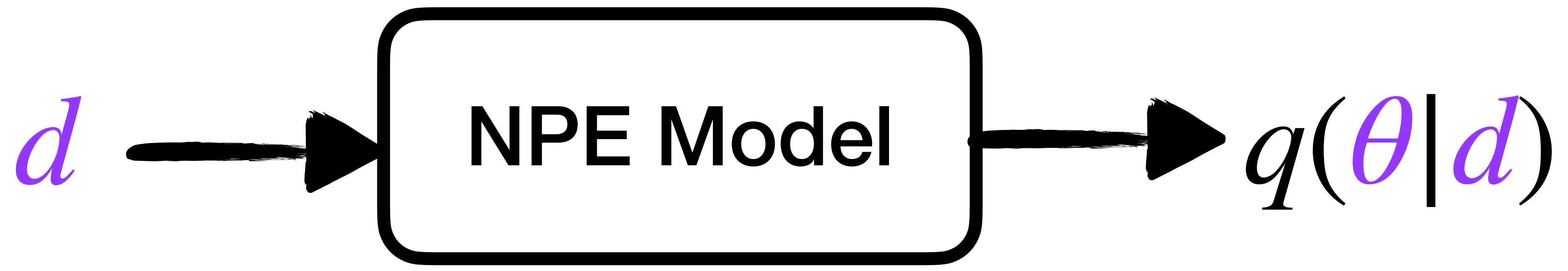
2. Train machine learning model q to approximate posterior

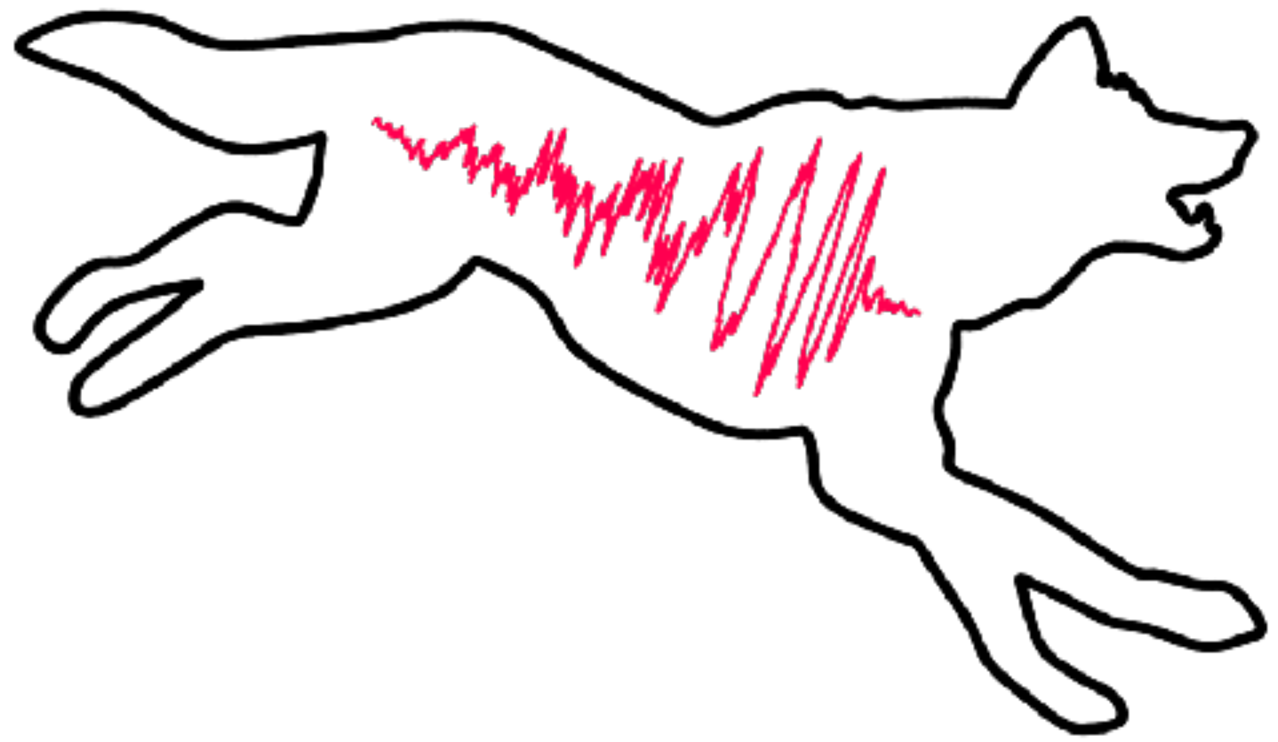
$$p(\theta\,|\,d) = \frac{p(d\,|\,\theta)}{p(d)}\, p(\theta) \approx q(\theta\,|\,d)$$



3. Evaluate model on measurement to obtain $q(\theta|d)$





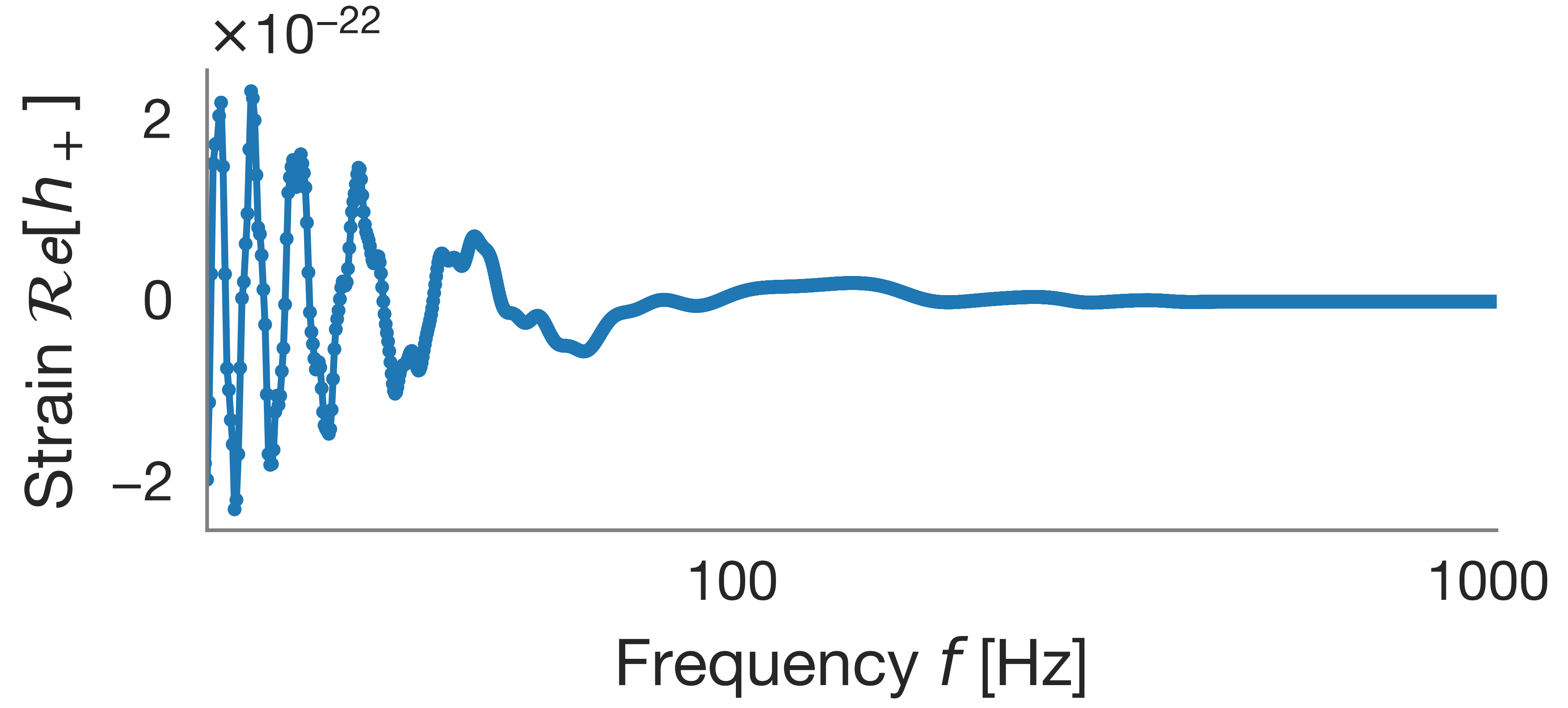


NPE for gravitational waves with DINGO

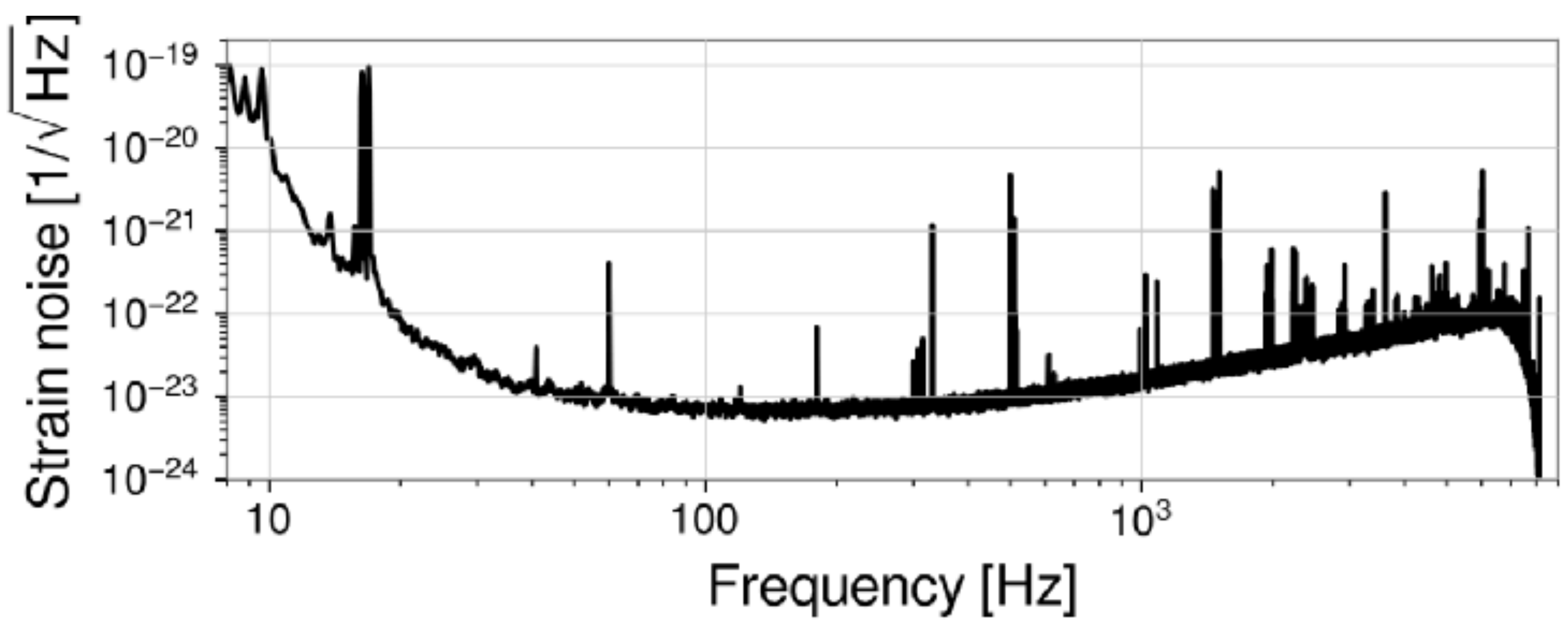
DINGO = <u>D</u>eep <u>IN</u>ference for <u>G</u>ravitational wave <u>O</u>bservations

NPE overview

- Generate simulated waveforms: $\theta \sim p(\theta), \;\; h = \text{simulator}(\theta) \;\rightarrow \{\theta, h\}$



- Add realistic noise to the waveform
  $S_n(f) \rightarrow \{S_n^{(i)}(f)\}$
  1. Sample noise $n^{(i)} \sim \mathcal{N}(0, \, S_n^{(i)})$
  2. Add to waveform $d^{(i)} = h(\theta^{(i)}) + n^{(i)}$



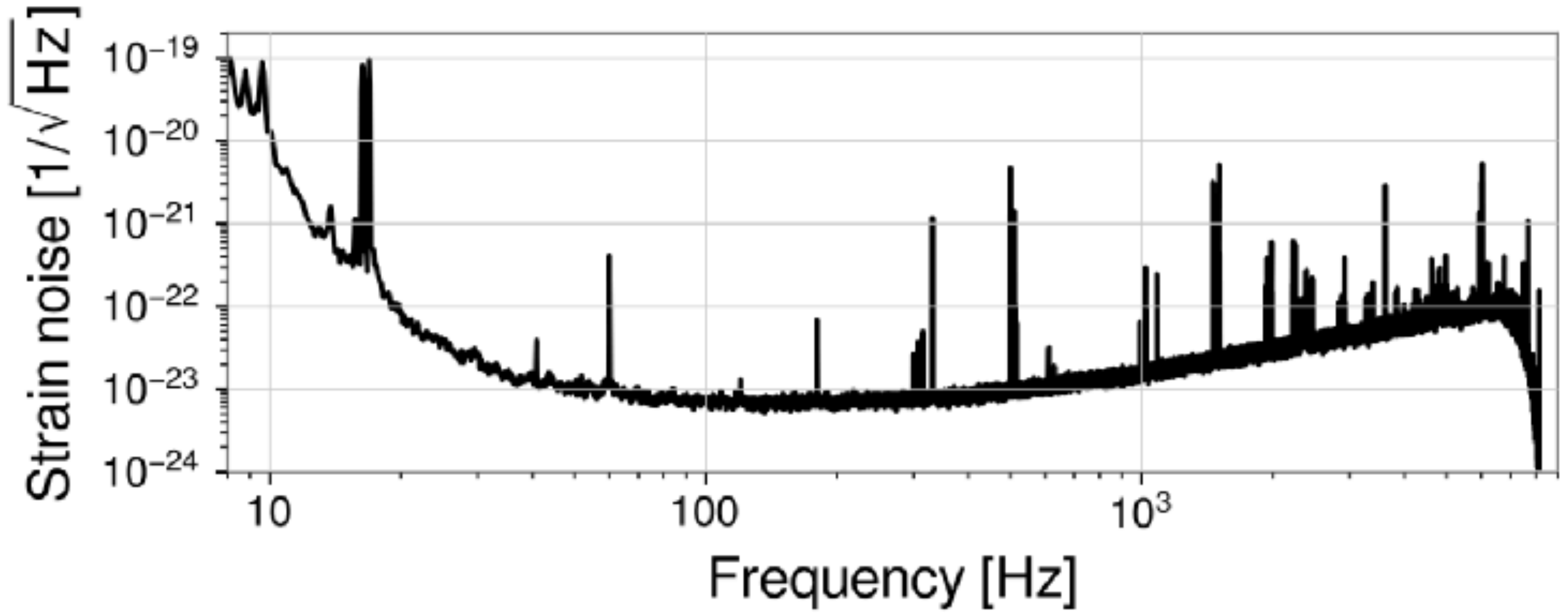
- Train conditional normalizing flow

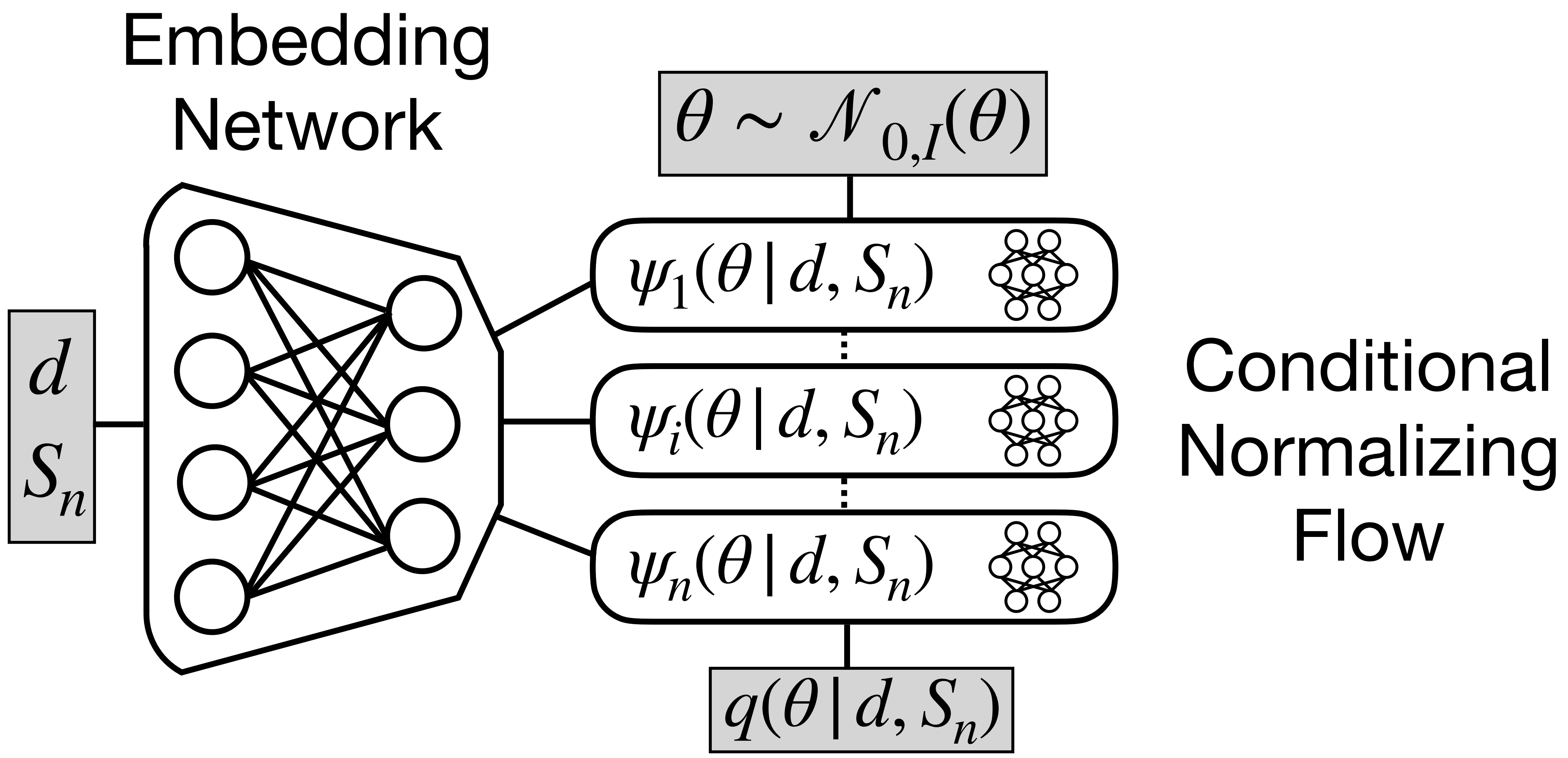
Dax+, PRL 2021

Changing PSDs

- Detector noise $S_n(f)$ varies from event to event
  → augment training to include collection of PSDs $S_n(f) \rightarrow \{S_n^{(i)}(f)\}$

1. Sample PSD $S_n^{(i)} \sim p(S_n)$
2. Generate noise $n^{(i)} \sim \mathcal{N}(0,\, S_n^{(i)})$
3. Add signal $d^{(i)} = h(\theta^{(i)}) + n^{(i)}$

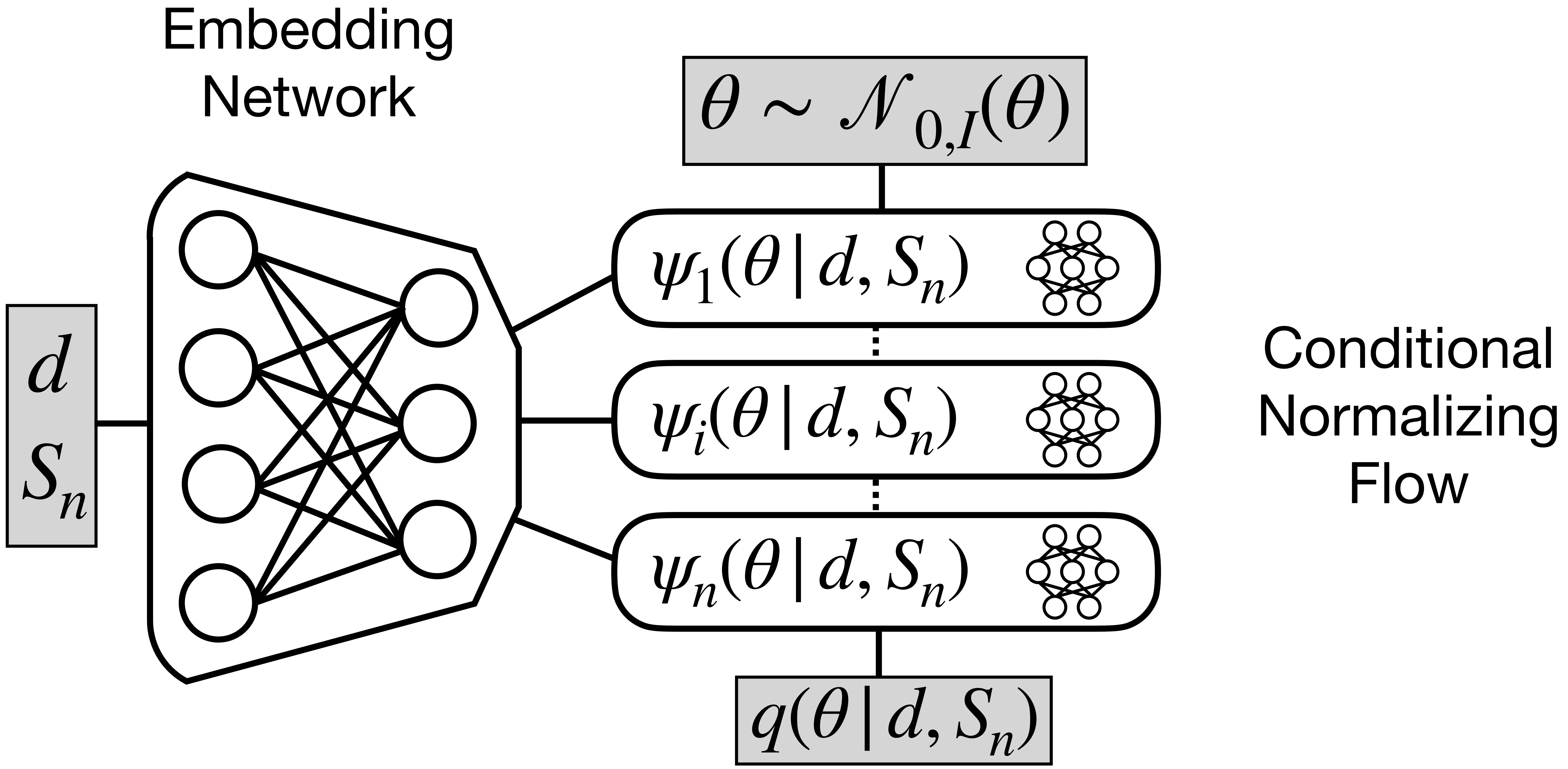




Training the model

- Provide data d and noise curve S_n to embedding network
- Train with negative log-likelihood loss

$$\mathscr{L} = -\frac{1}{N} \sum_{\substack{\theta^{(i)} \sim p(\theta) \\ d^{(i)} \sim p(d \mid \theta^{(i)})}} \log q(\theta^{(i)} \mid d^{(i)})$$



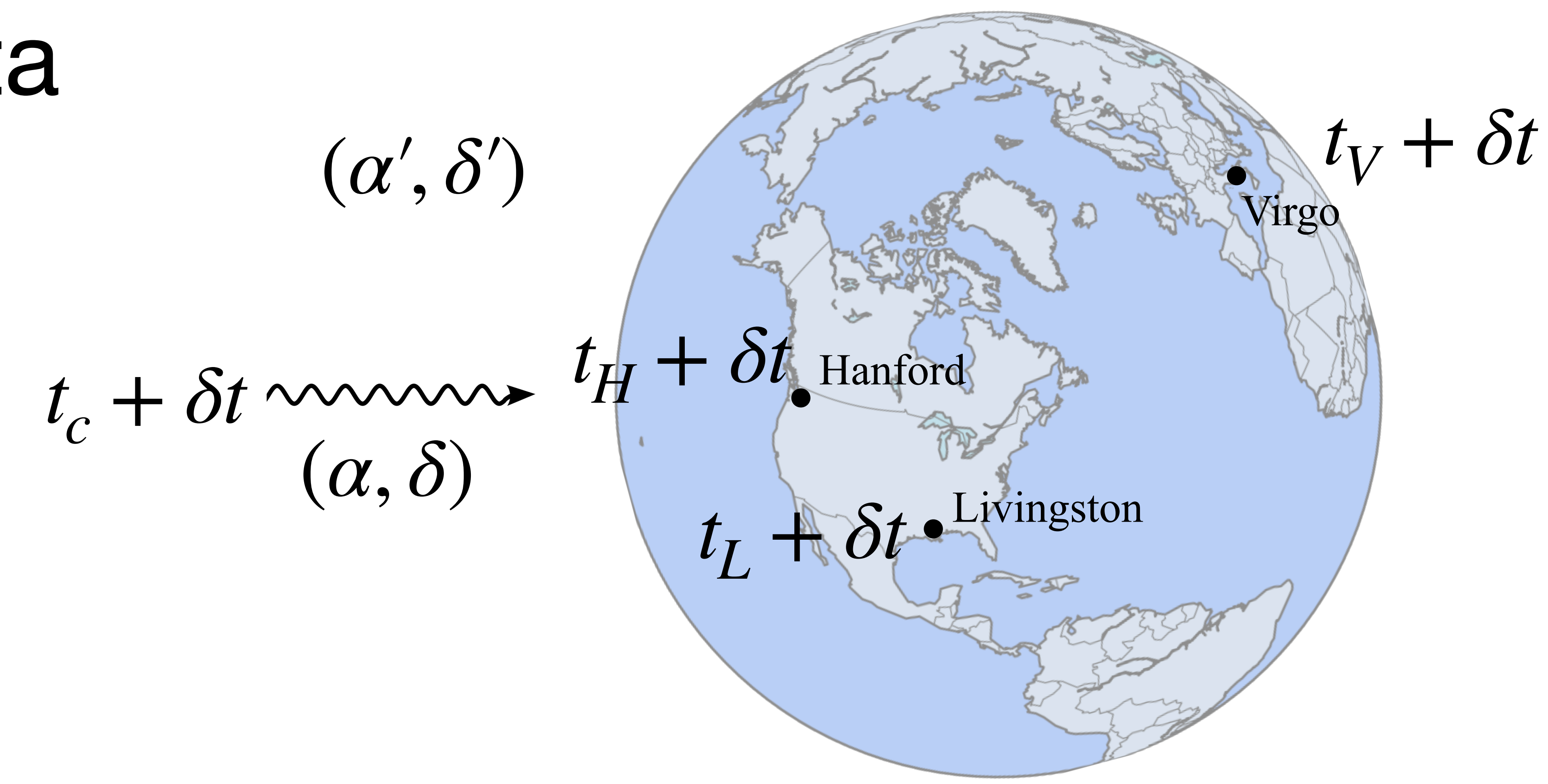
Dax+, PRL 2021

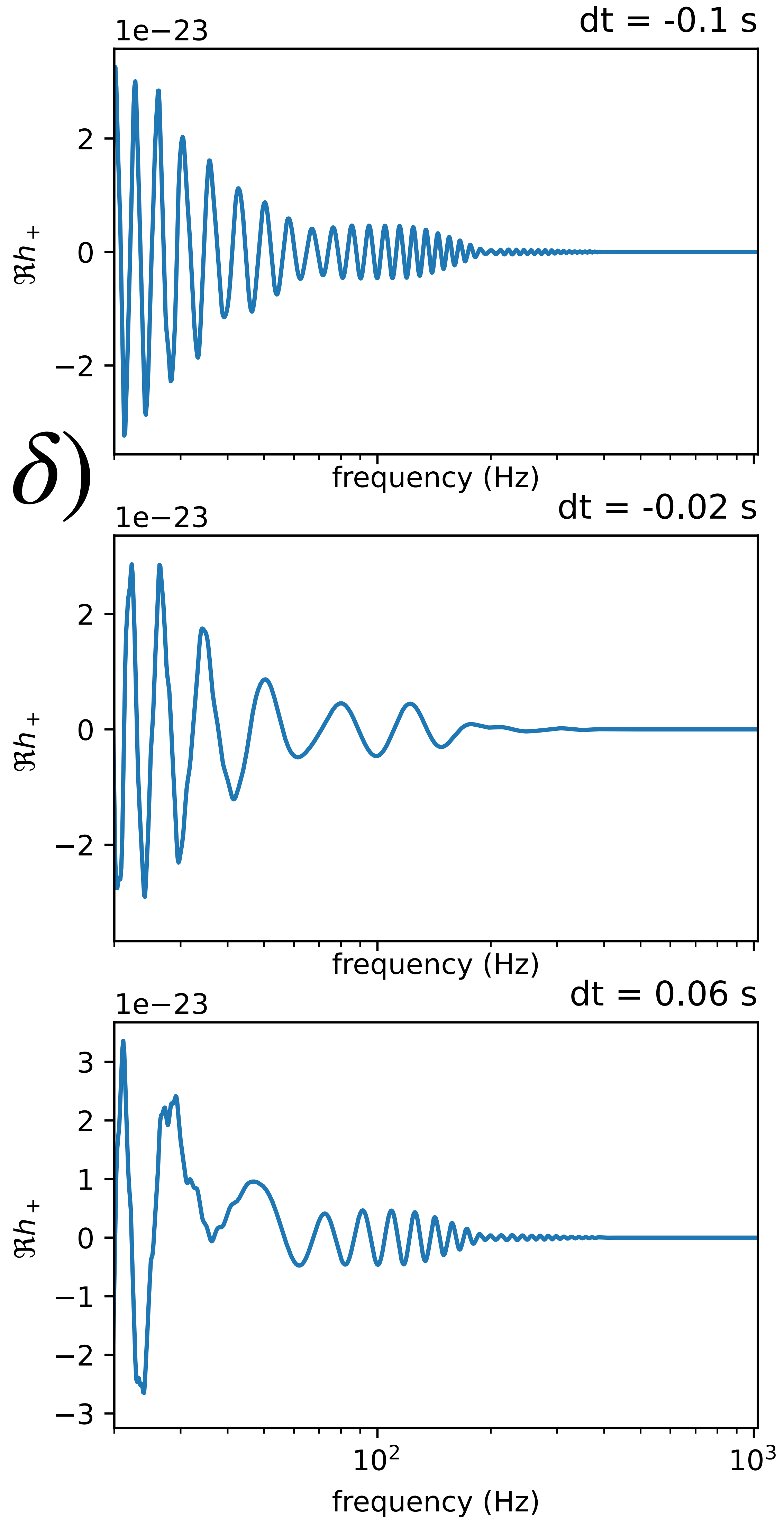
Simplifying Data using Symmetries

- Posterior should be …
  … equivariant wrt. overall coalescence time t_c
  … approximately equivariant wrt. changes in sky position (α, δ)

- Frequency-domain data
  $\rightarrow$ time shift = multiplication by $e^{2\pi i f \cdot \delta t}$

- Hard to learn just from data





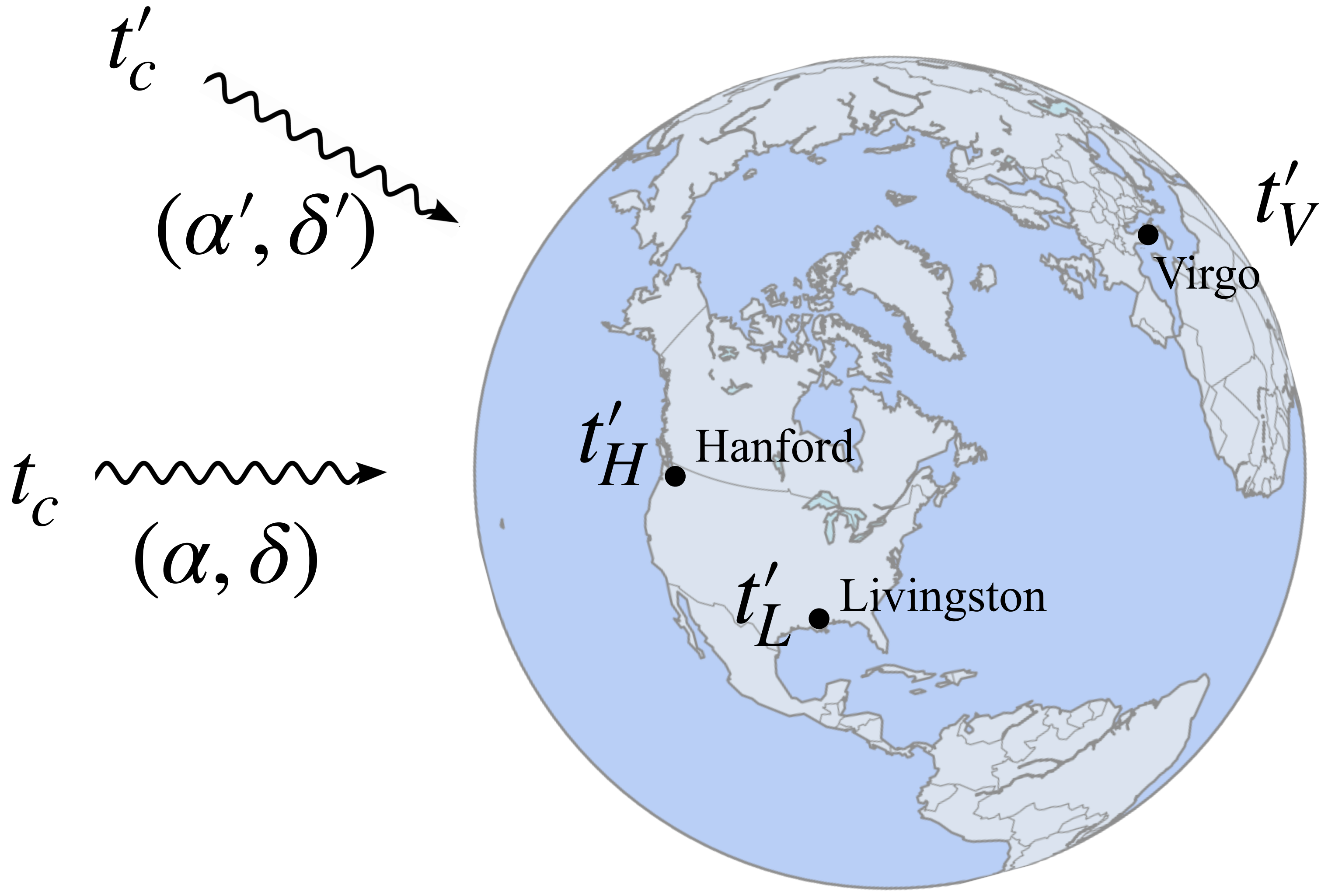
Dax+, ICLR 2022

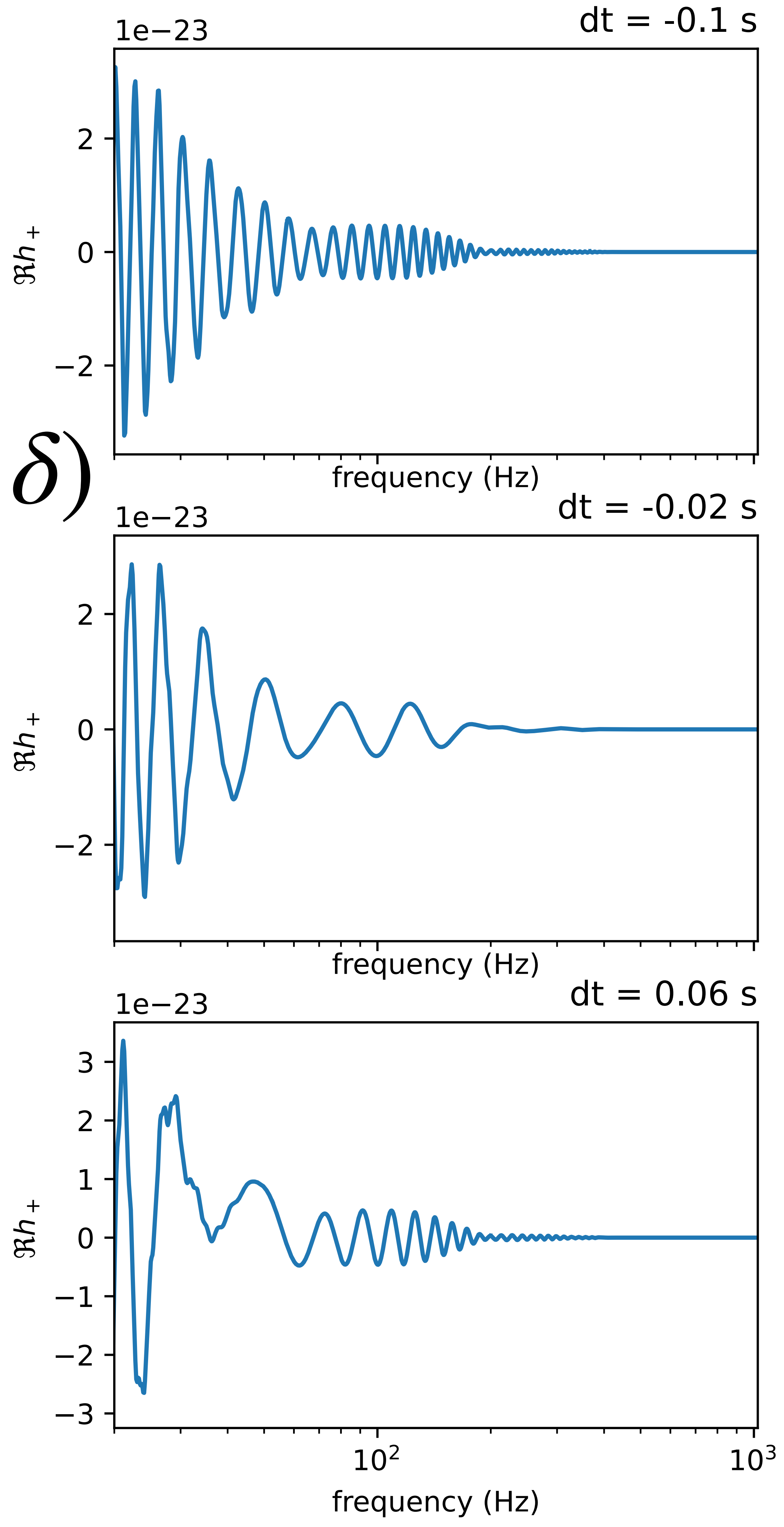
Simplifying Data using Symmetries

- Posterior should be …
  … equivariant wrt. overall coalescence time t_c
  … approximately equivariant wrt. changes in sky position (α, δ)

- Frequency-domain data
  → time shift = multiplication by $e^{2\pi i f \cdot \delta t}$

- Hard to learn just from data





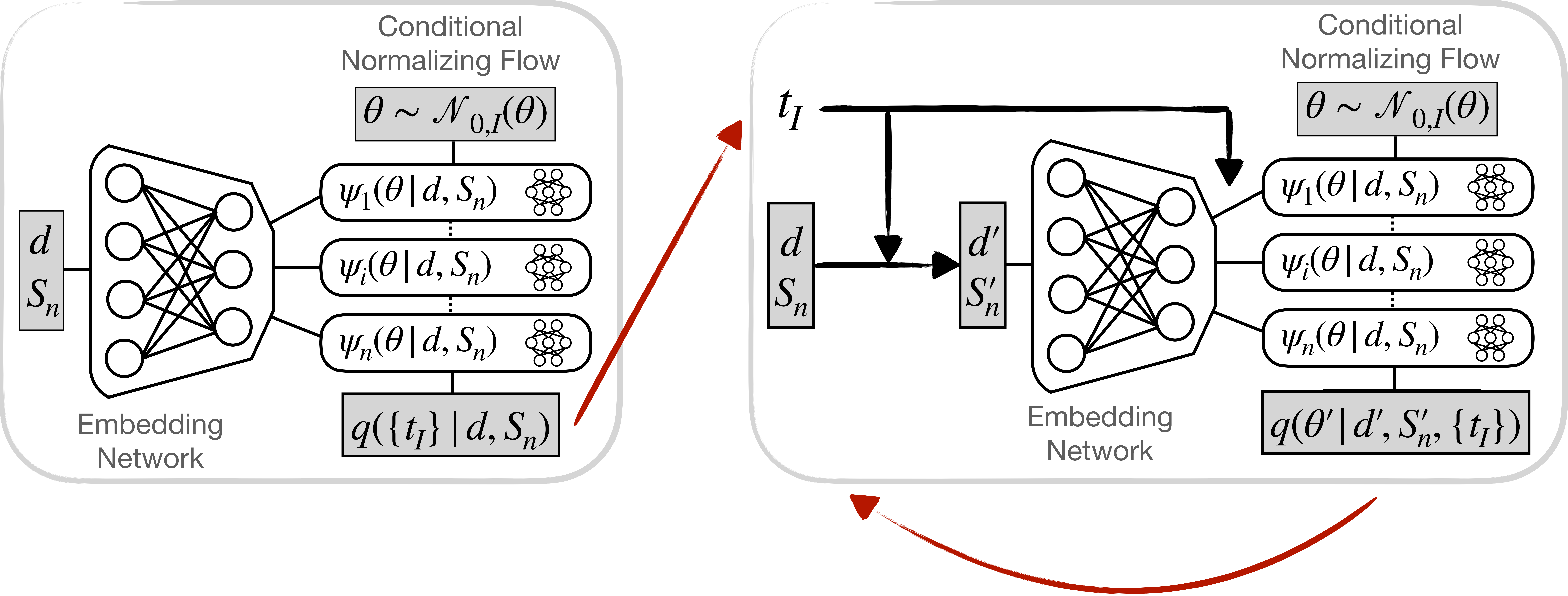
GNPE = Group-equivariant NPE

- Standardize data within band around $t_I = 0$
- Gibbs Sampling:
  - Start with blurred proxy $t_I + \epsilon_I$ with $\epsilon_I \sim \kappa(\epsilon_I)$
  - Learn approximate distribution $q_{\text{init}}(\{t_I\}_{I=H,L,V} \,|\, d)$ for time shift
  - Time-translate data $d' = d(t_I = 0)$
  - Train conditional density estimator $q(\theta' \,|\, d', t_I + \epsilon_I)$

Dax+, ICLR 2022

GNPE = Group-equivariant NPE

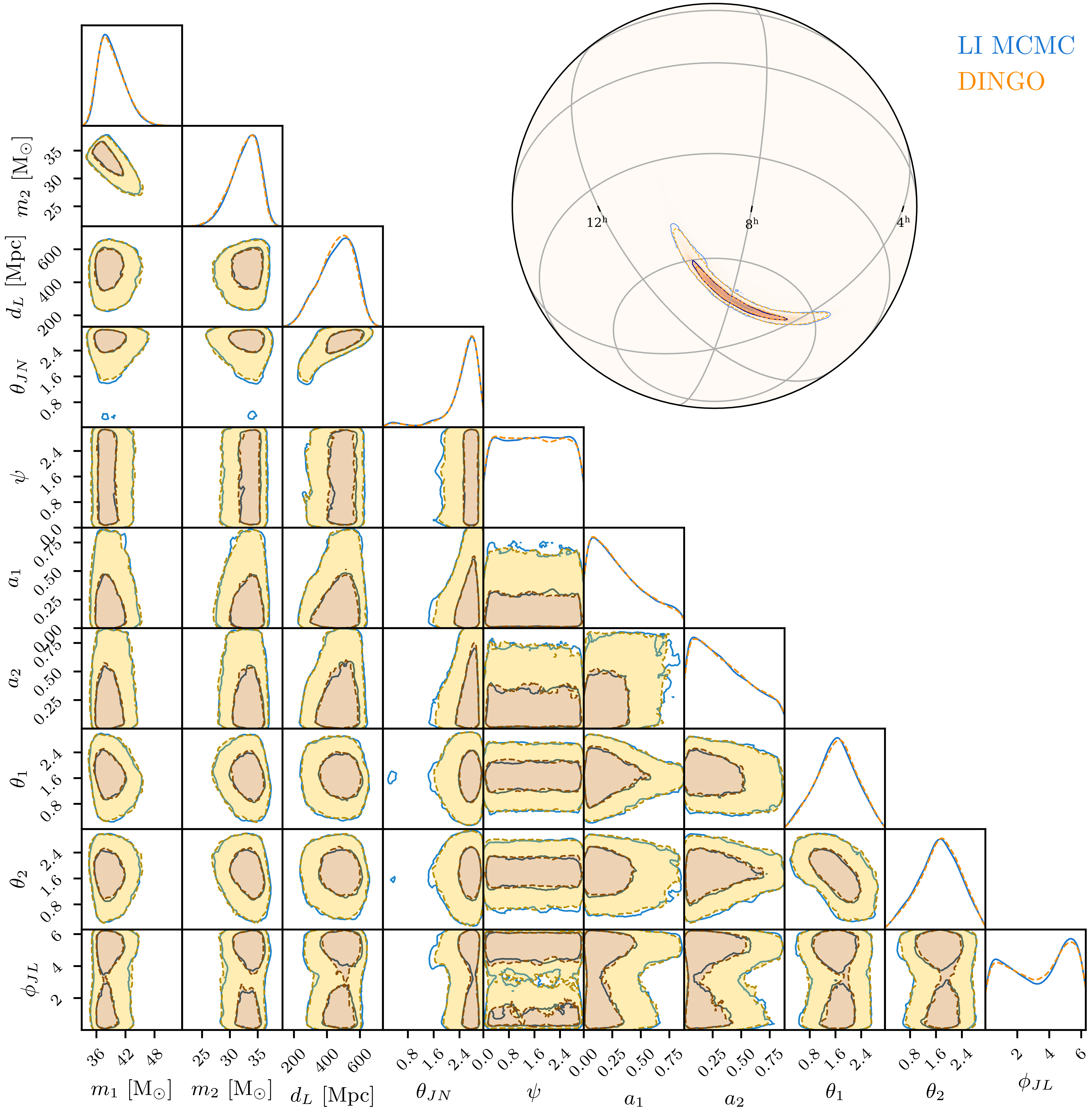
- Include condition on t_I in DINGO



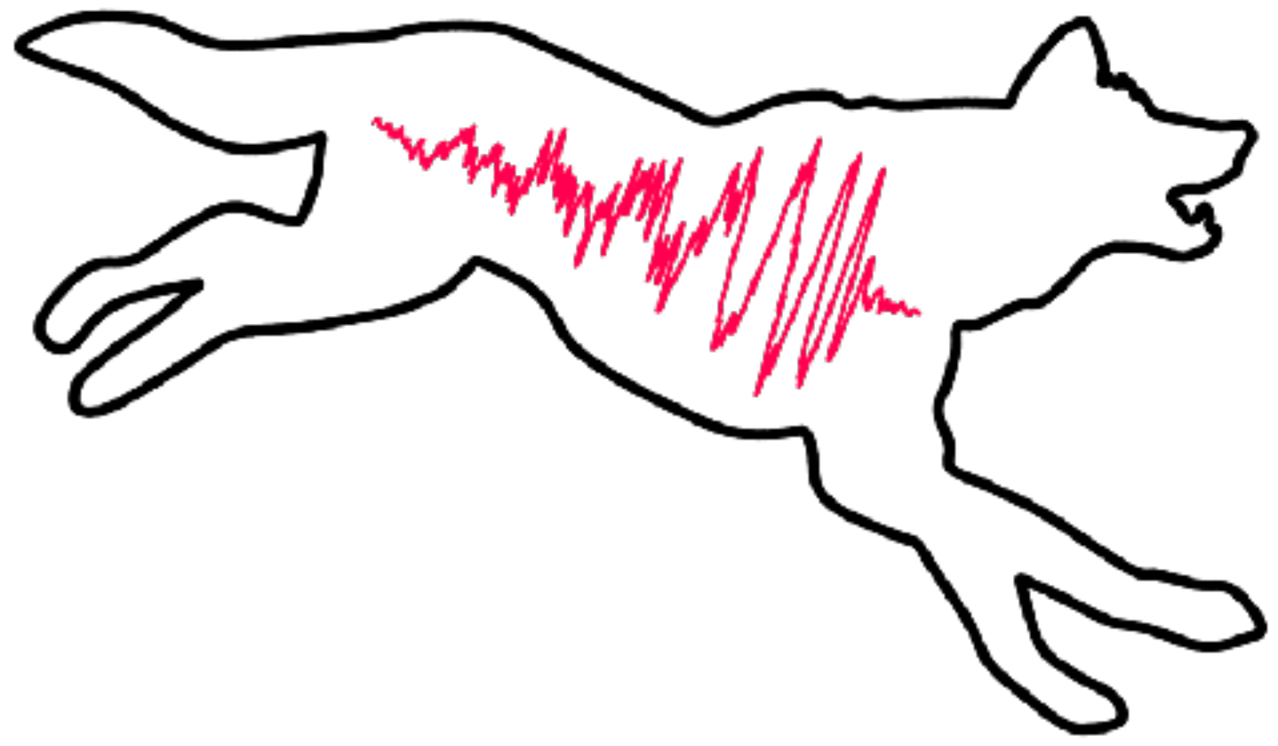
 Dax+, ICLR 2022

Results with GNPE

- Very good agreement with standard techniques
- $\sim 10^7$ training examples
  $\sim 10^8$ network parameters
- Training: $\sim$ few days
  Inference: $\sim$ few minutes
- Amortized method:
  - Train once
  - Perform inference on many events with minimal cost



Dax+, PRL 2021



Importance Sampling with DINGO

DINGO-IS

Verifying Results: DINGO-IS

- Why is it important to verify the results of any ML model (like DINGO)?
  - Mistrust of physics community towards ML methods
  - Scientific discovery statements require extra care
- Why is importance sampling (IS) ideal for this?
  - Sampling method used in almost all standard samplers
  - You don’t have to trust the DINGO model, you just have to trust IS!

How does importance sampling work?

- Two distributions
  - Target distribution = true posterior $p(\theta | d)$
  - Proposal = learned posterior $q(\theta | d)$
- Compute importance weight

$$\theta_i \sim q(\theta_i | d) \qquad w_i = \frac{p(\theta_i | d)}{q(\theta_i | d)}$$

- (Assume for now: $p(\theta | d)$ is known)

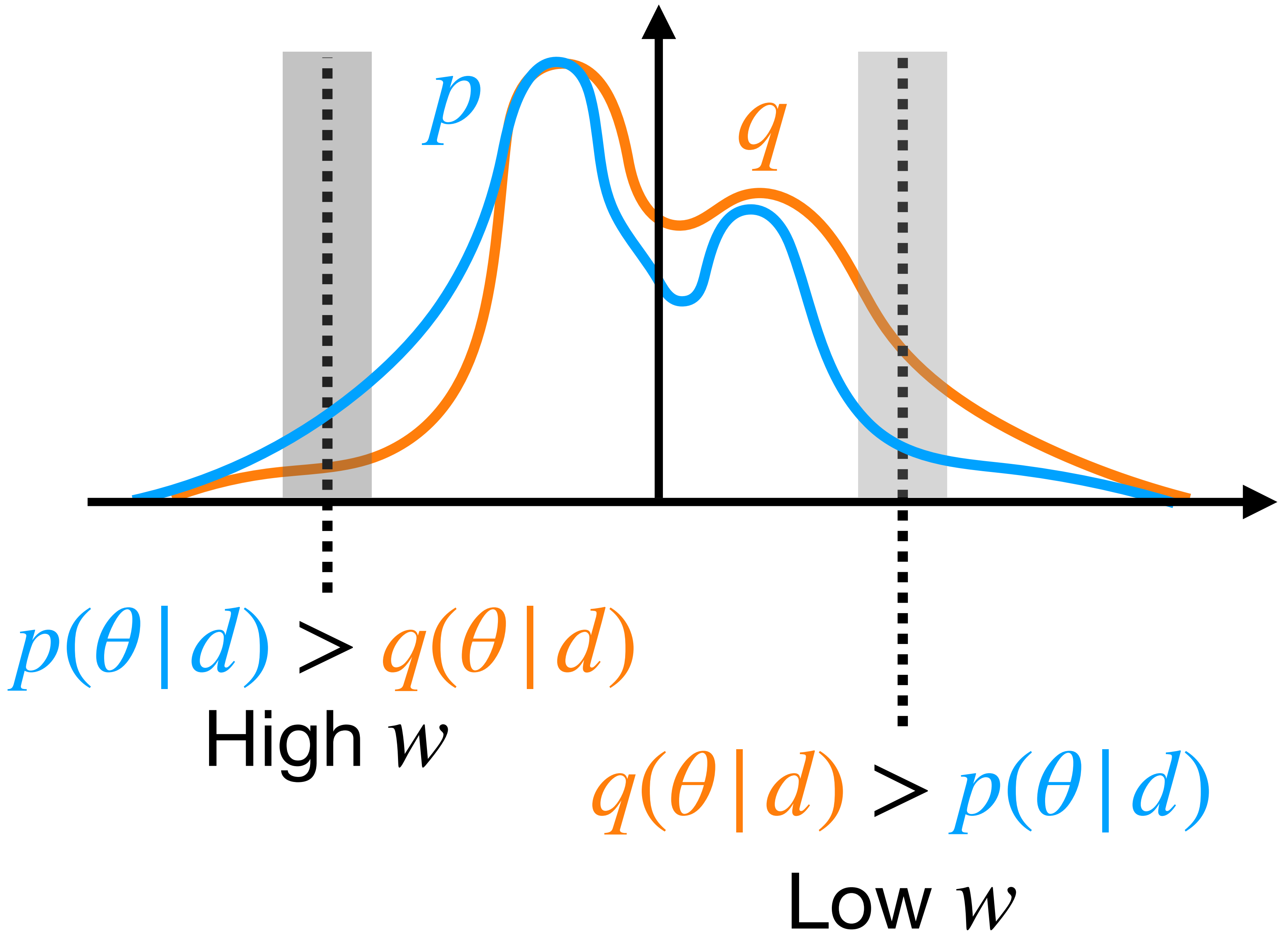
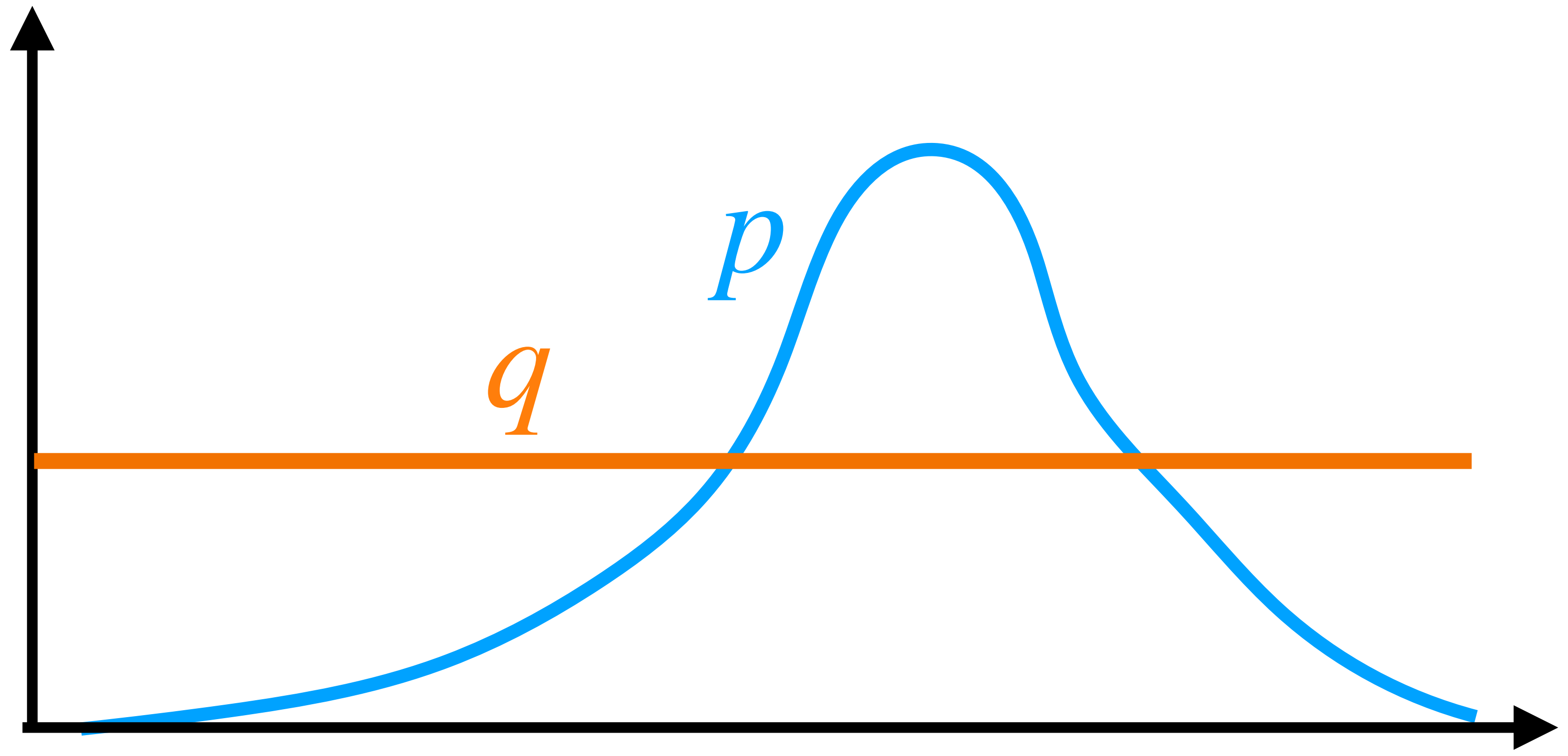


Illustration of importance weights



- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Uniform $q(x)$
- Importance weight $$w(x) = \frac{p(x)}{q(x)}$$

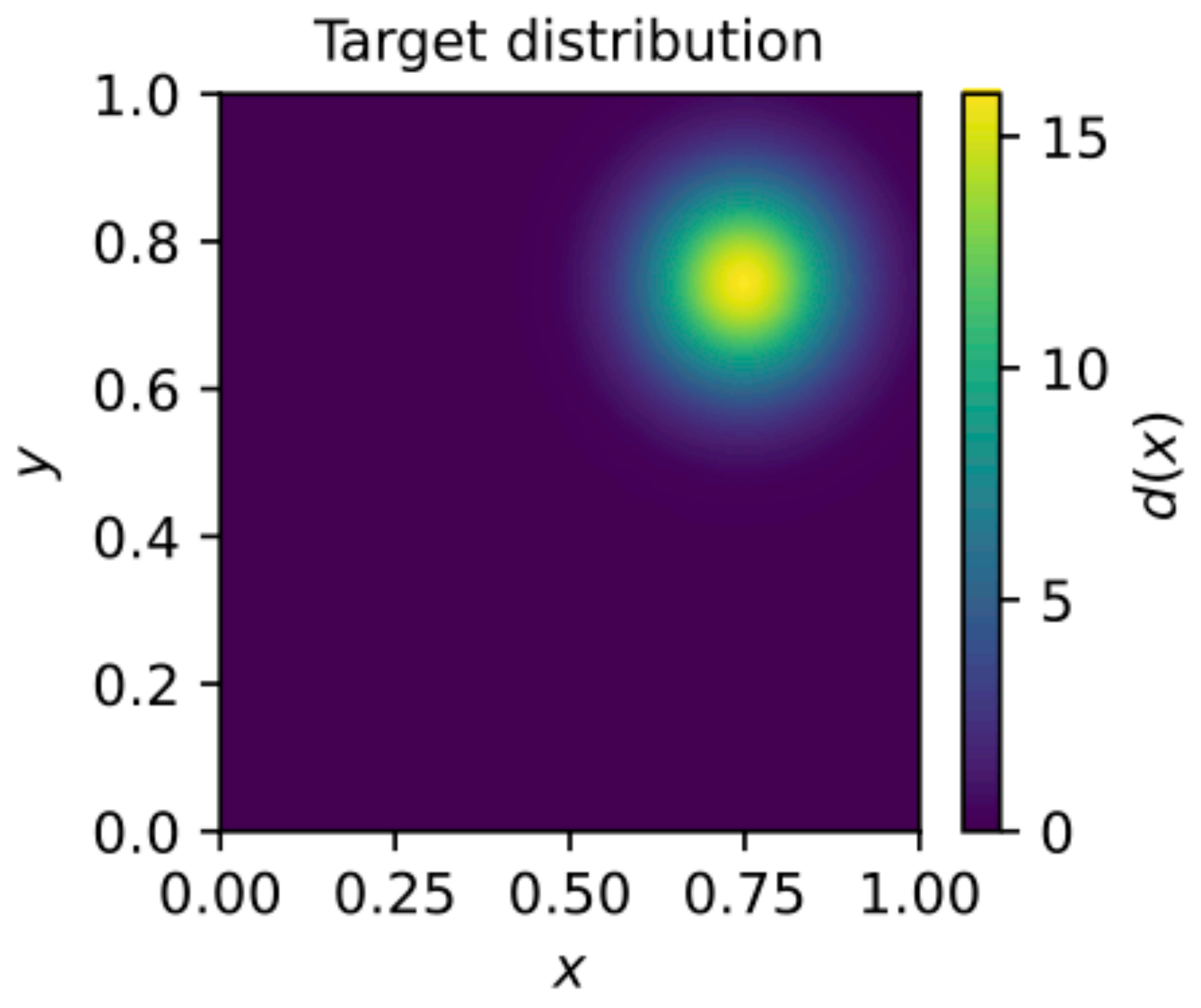
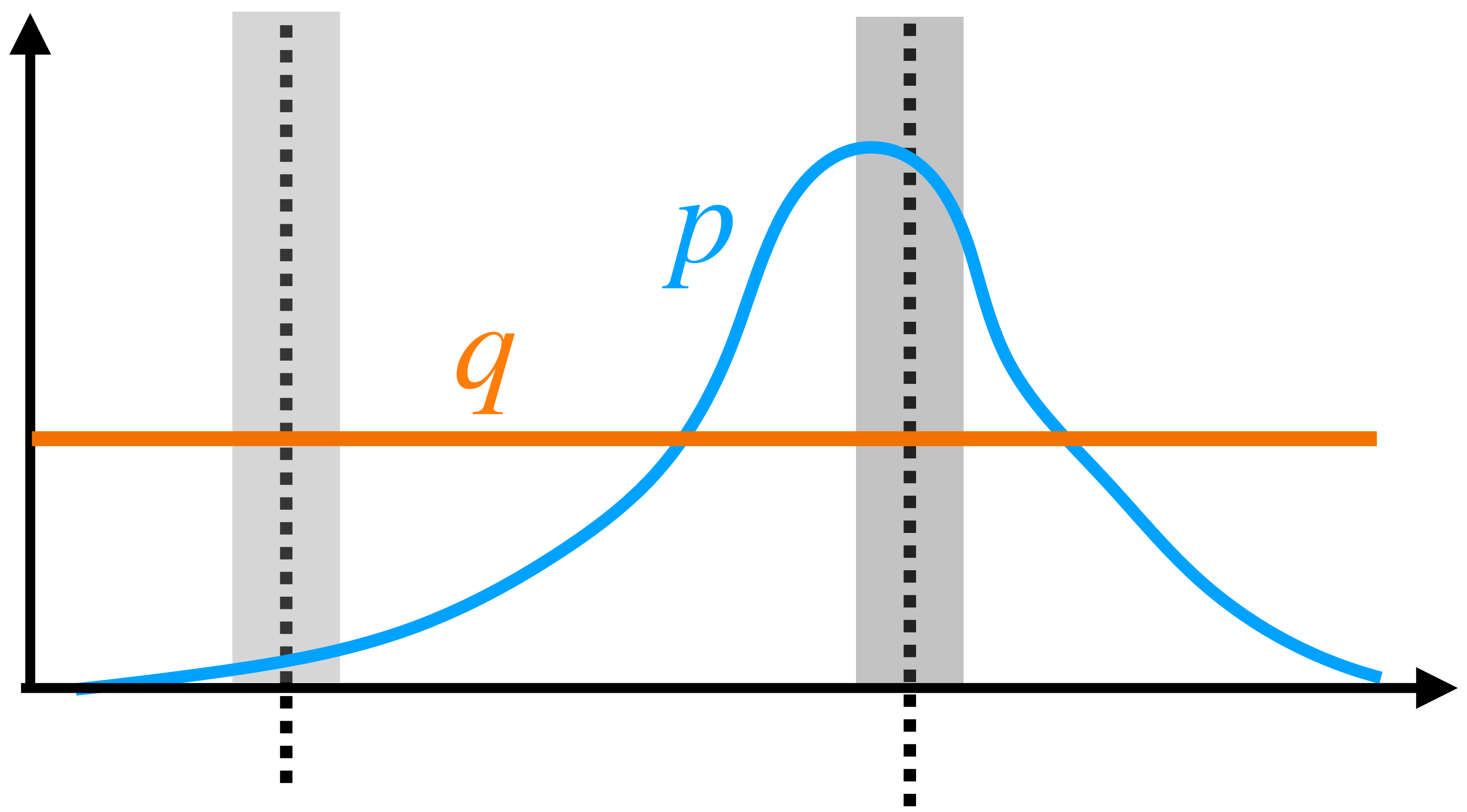


Illustration of importance weights

- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Uniform $q(x)$
- Importance weight $\quad w(x) = \dfrac{p(x)}{q(x)} = \dfrac{p(x)}{\text{const.}}$



$q(x) > p(x)$ Low w

$p(x) > q(x)$ High w

$x \sim q(x)$

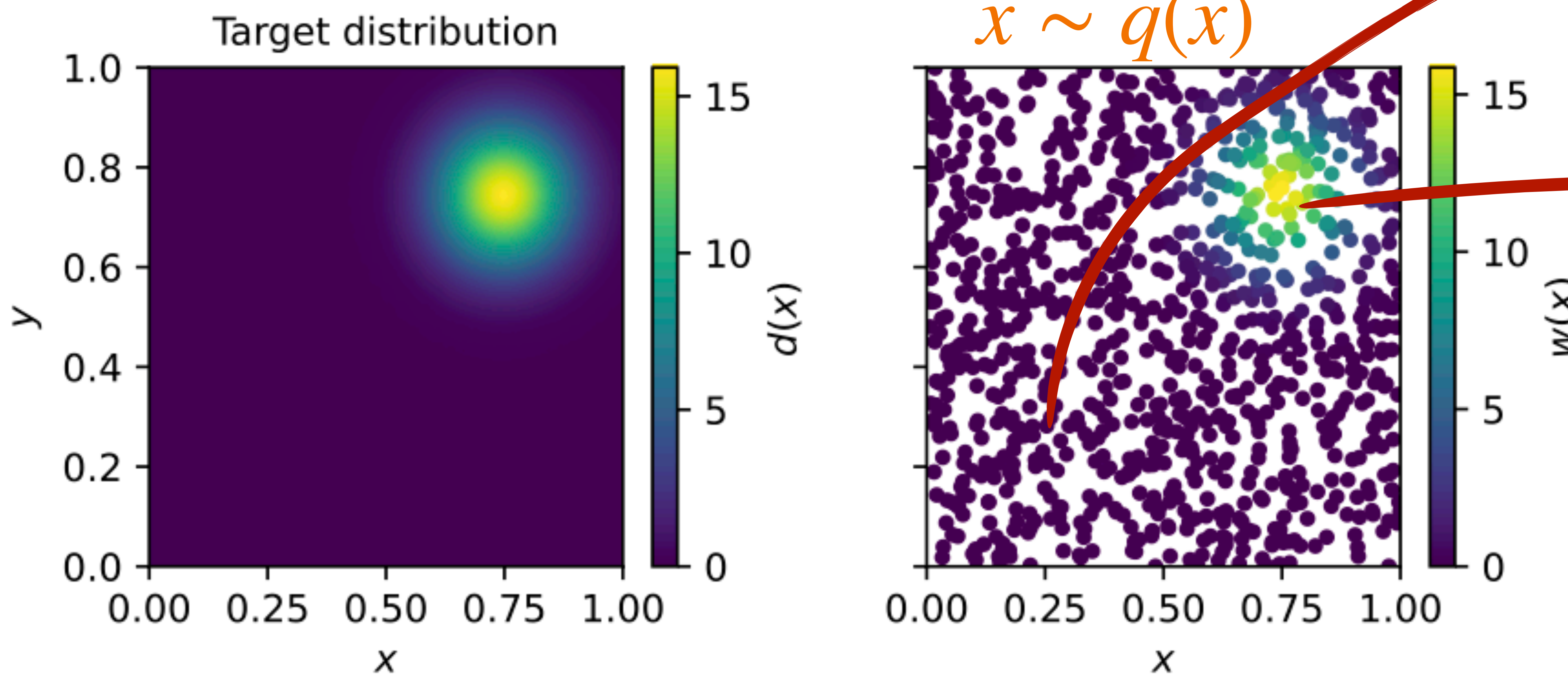
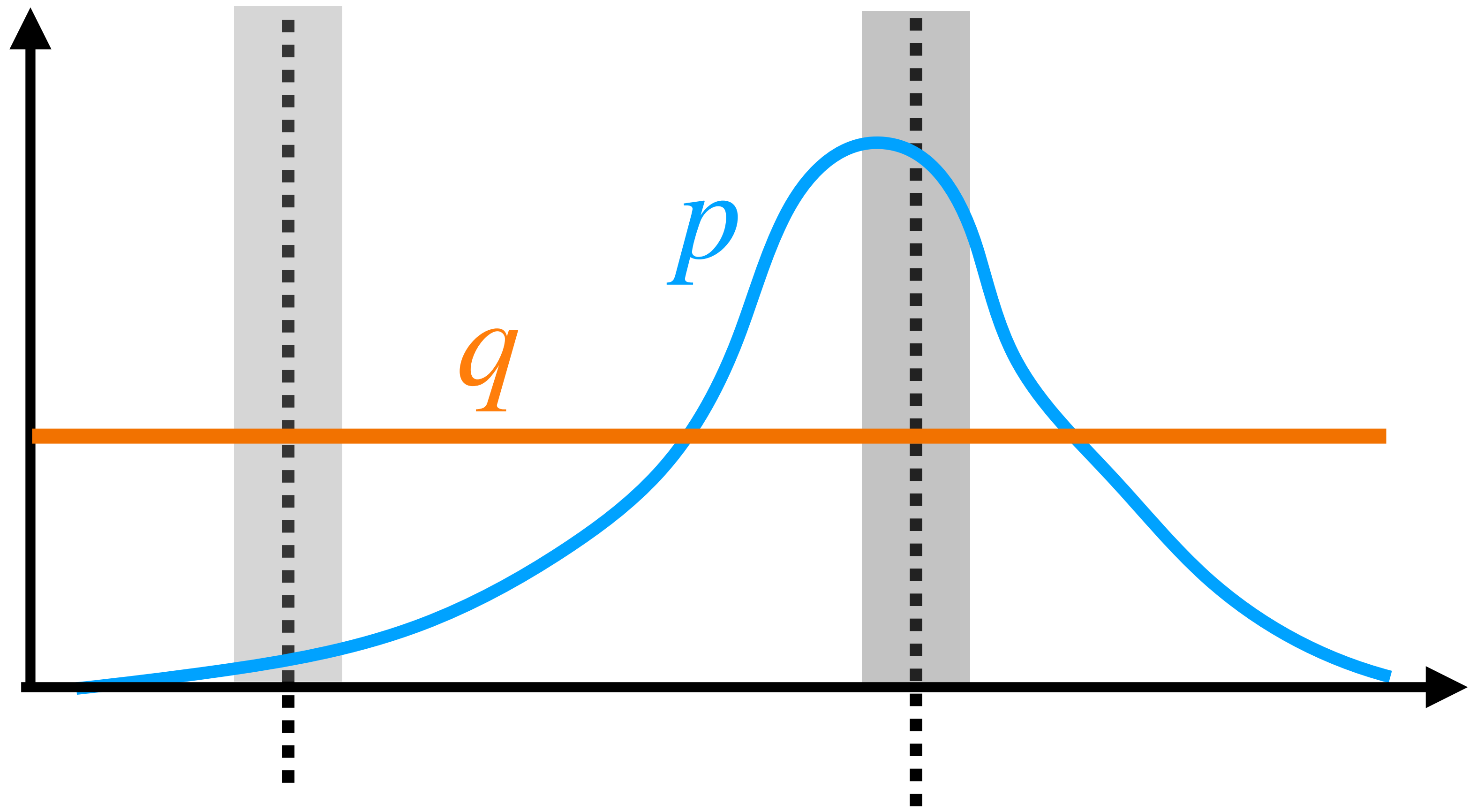


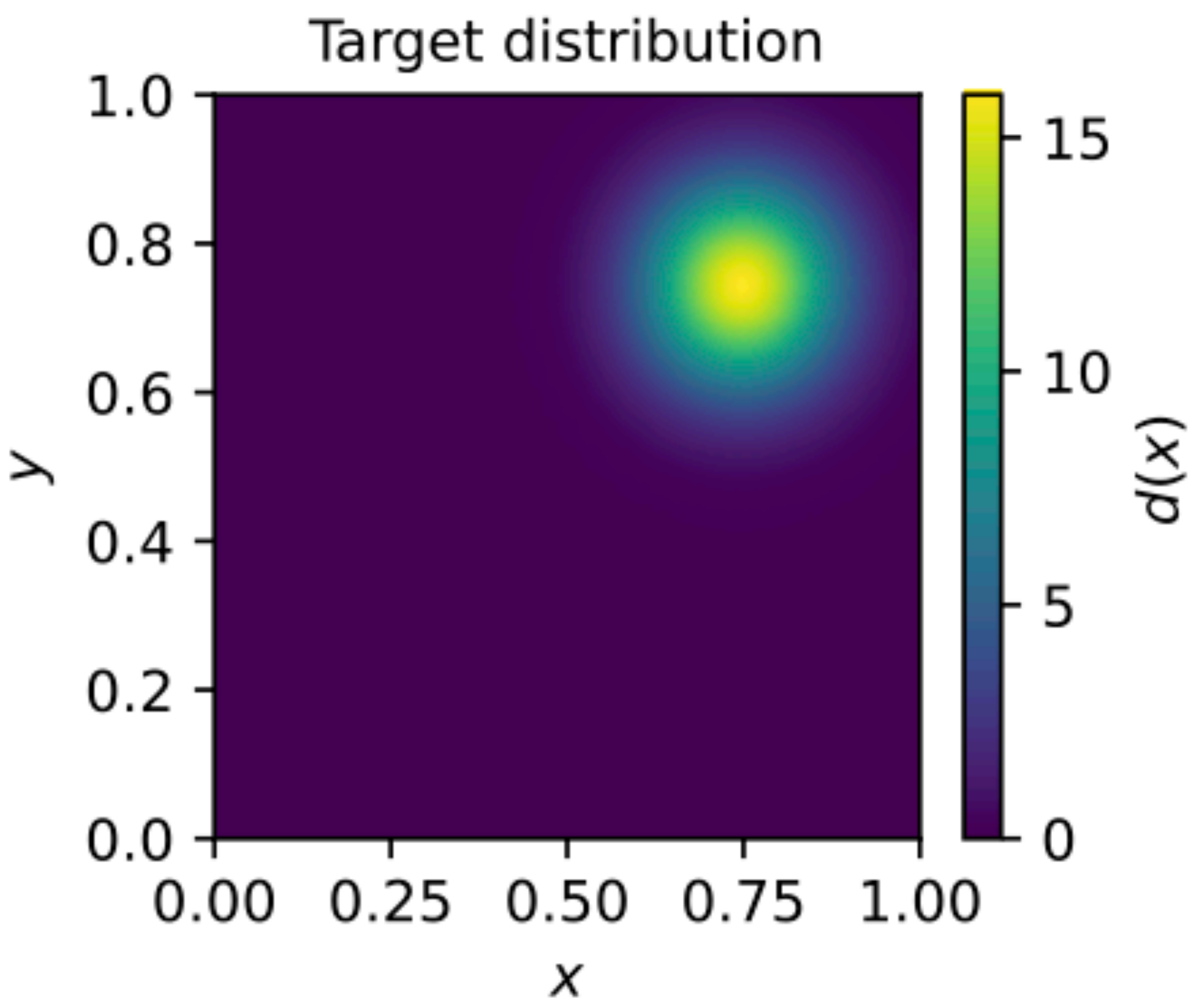
Illustration of importance weights

- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Uniform $q(x)$
- Importance weight $w(x) = \dfrac{p(x)}{q(x)}$

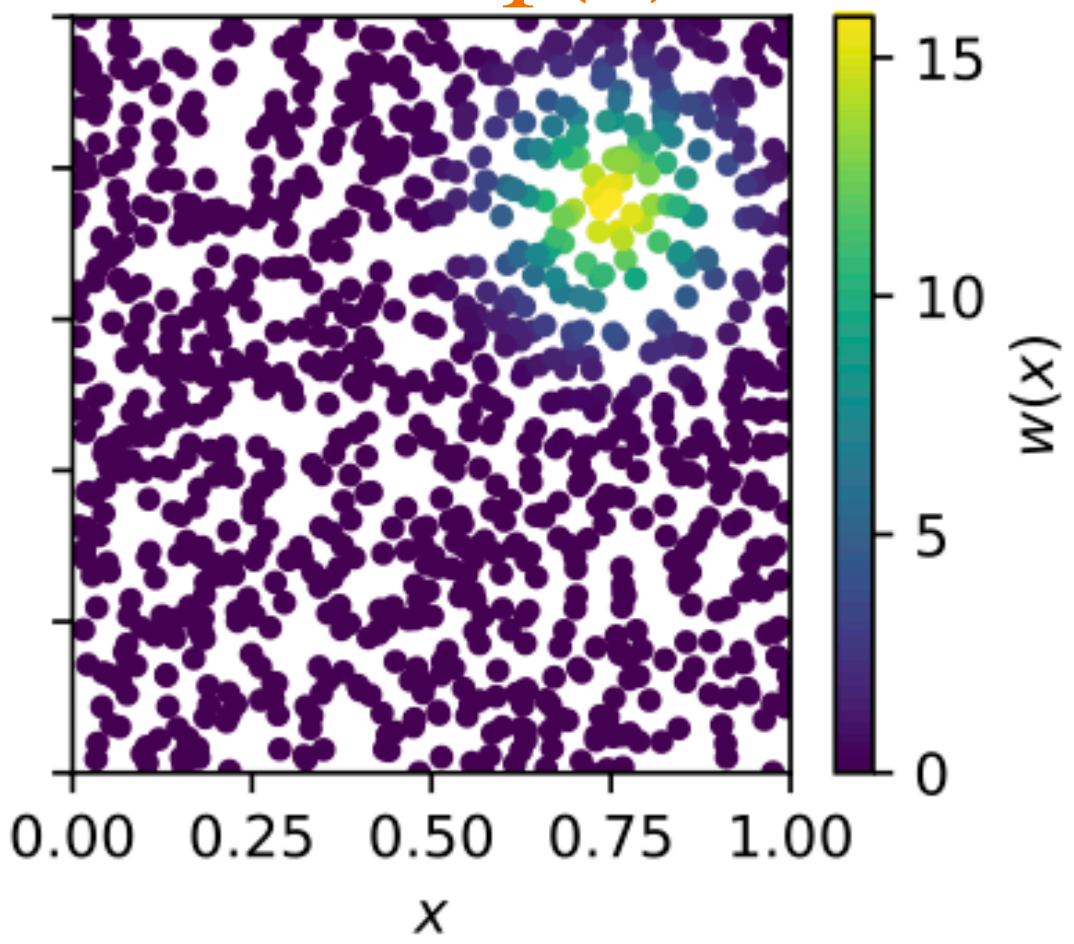


$q(x) > p(x)$ Low w

$p(x) > q(x)$ High w

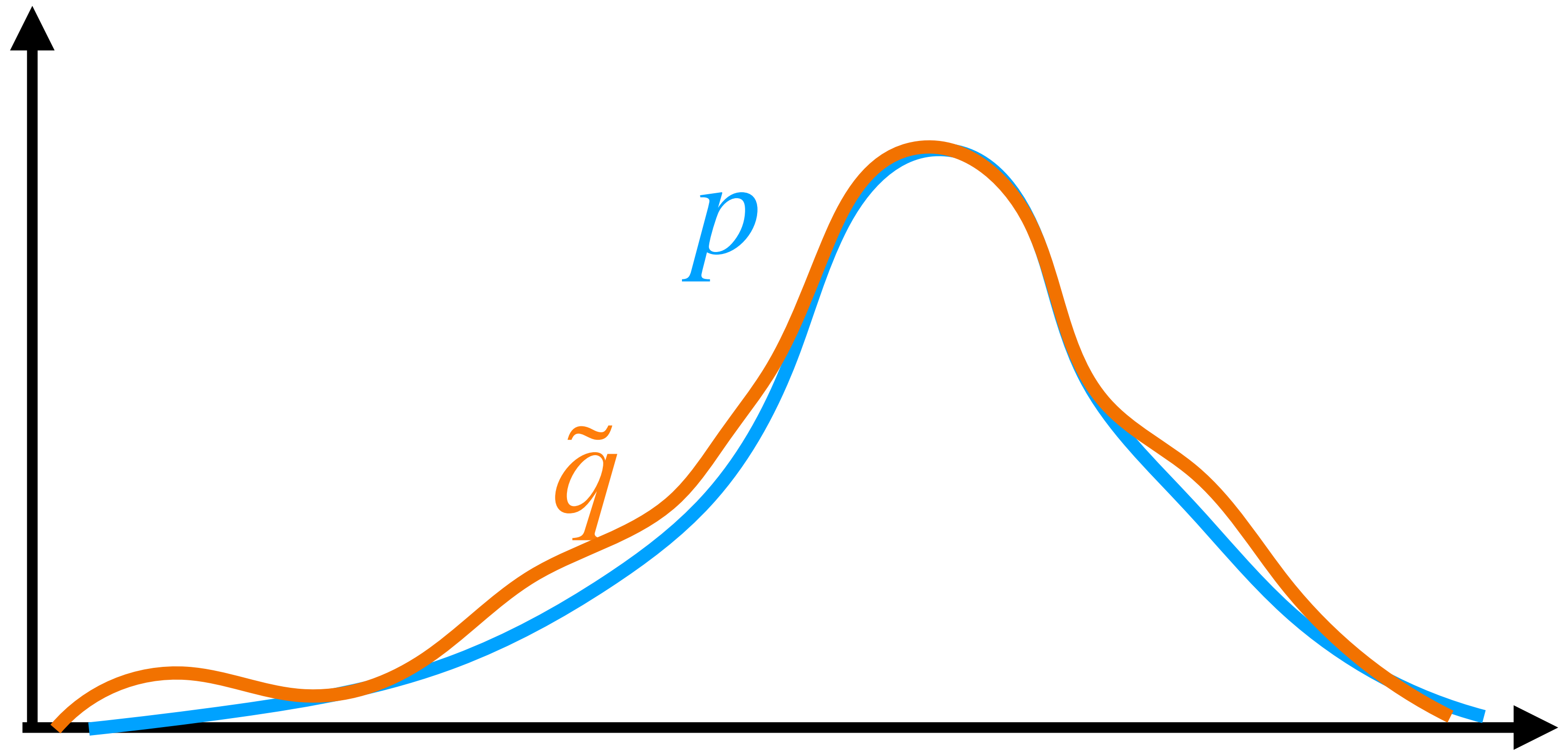


$x \sim q(x)$



Large variance in $w(x)$

Illustration of importance weights



- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Optimized distribution $\tilde{q}(x)$
- Importance weight $w(x) = \dfrac{p(x)}{q(x)}$

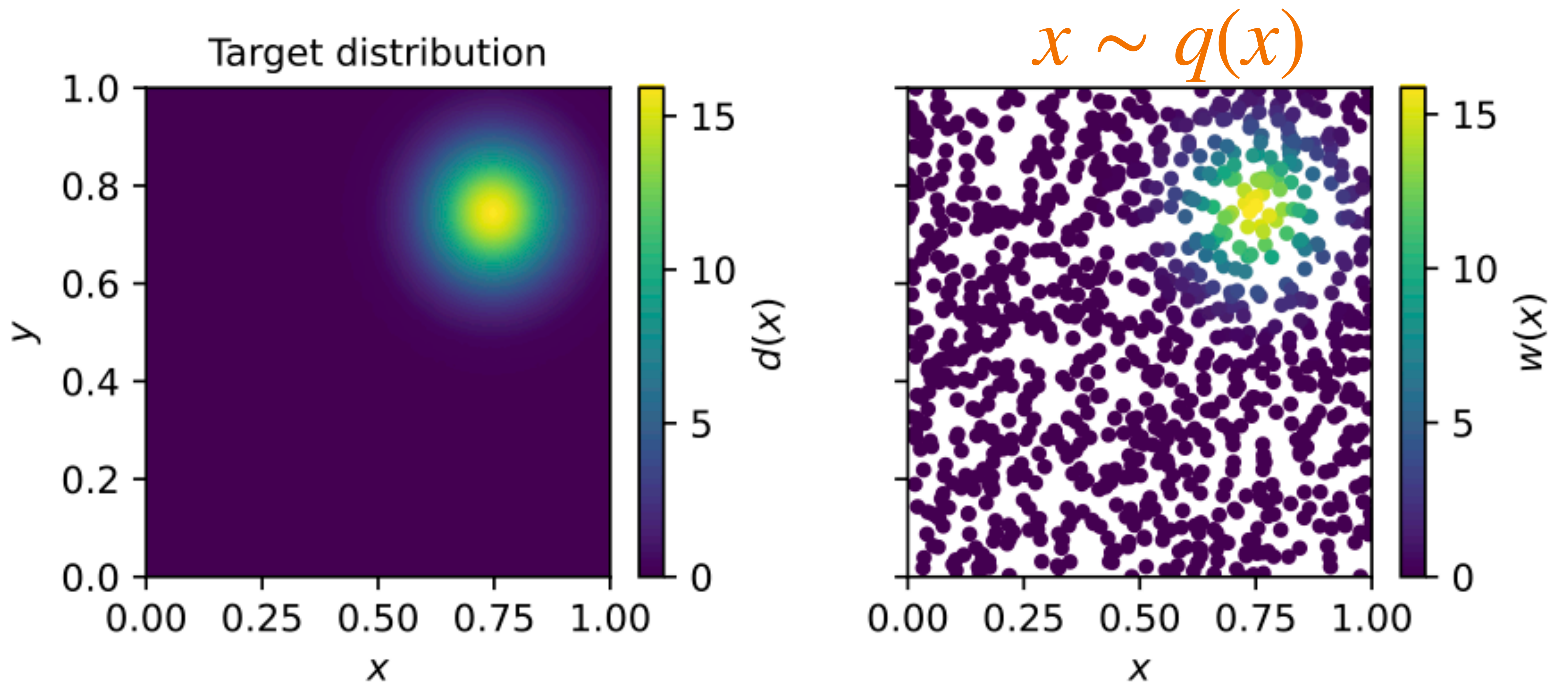
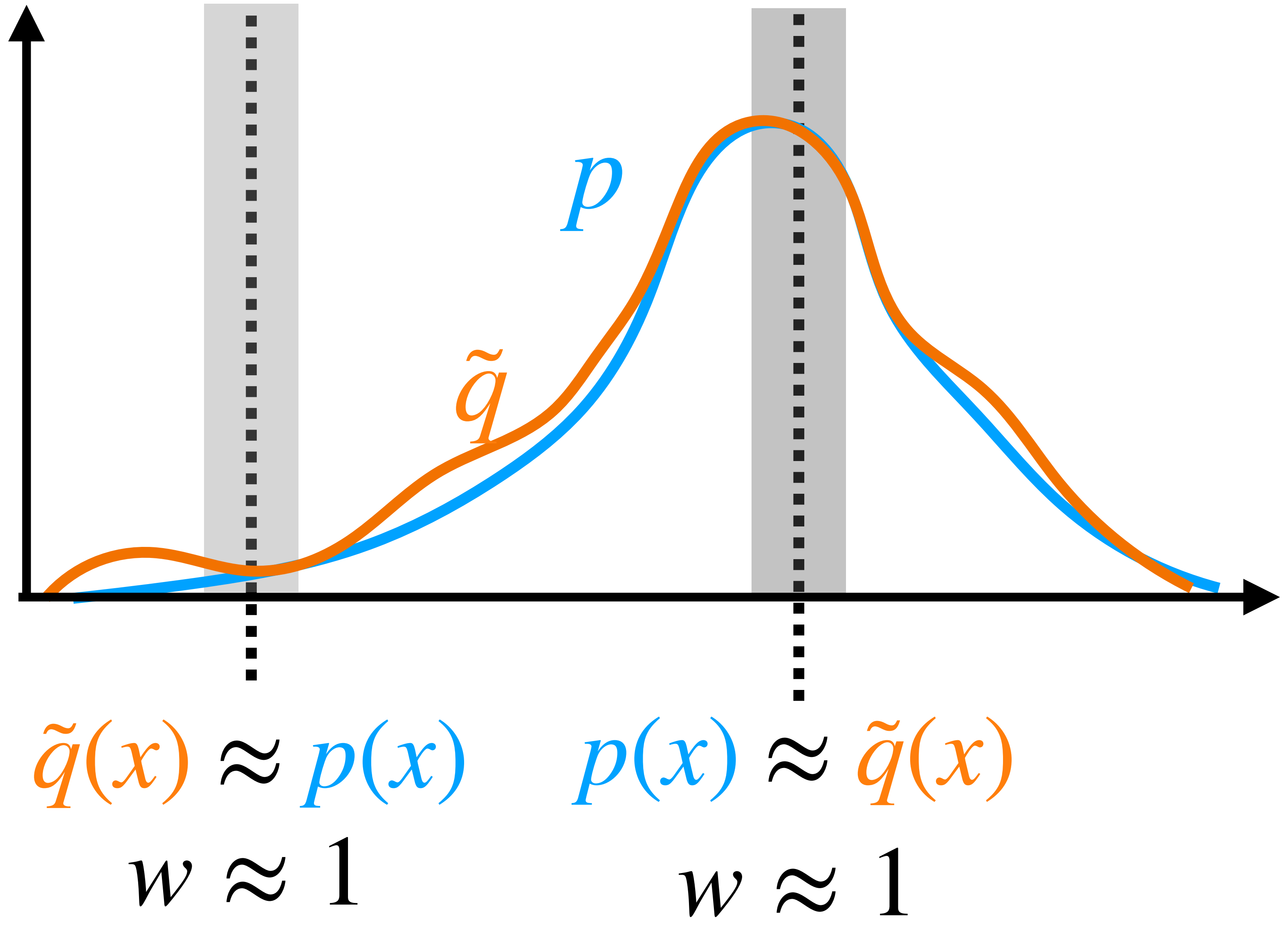


Illustration of importance weights



- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Optimized distribution $\tilde{q}(x)$
- Importance weight $w(x) = \dfrac{p(x)}{q(x)}$

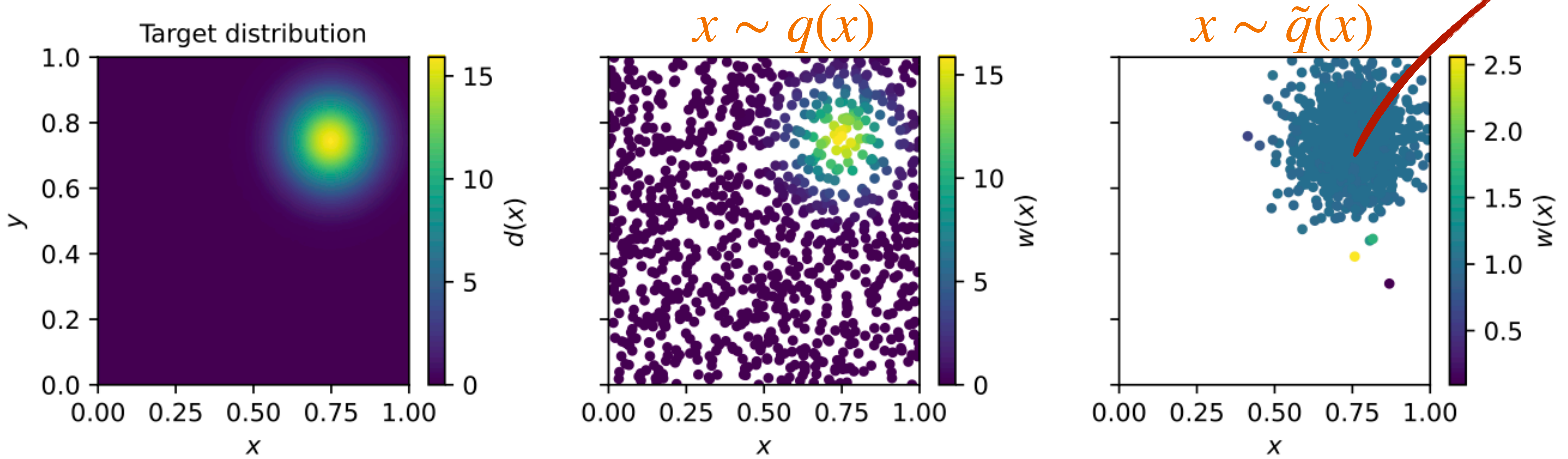
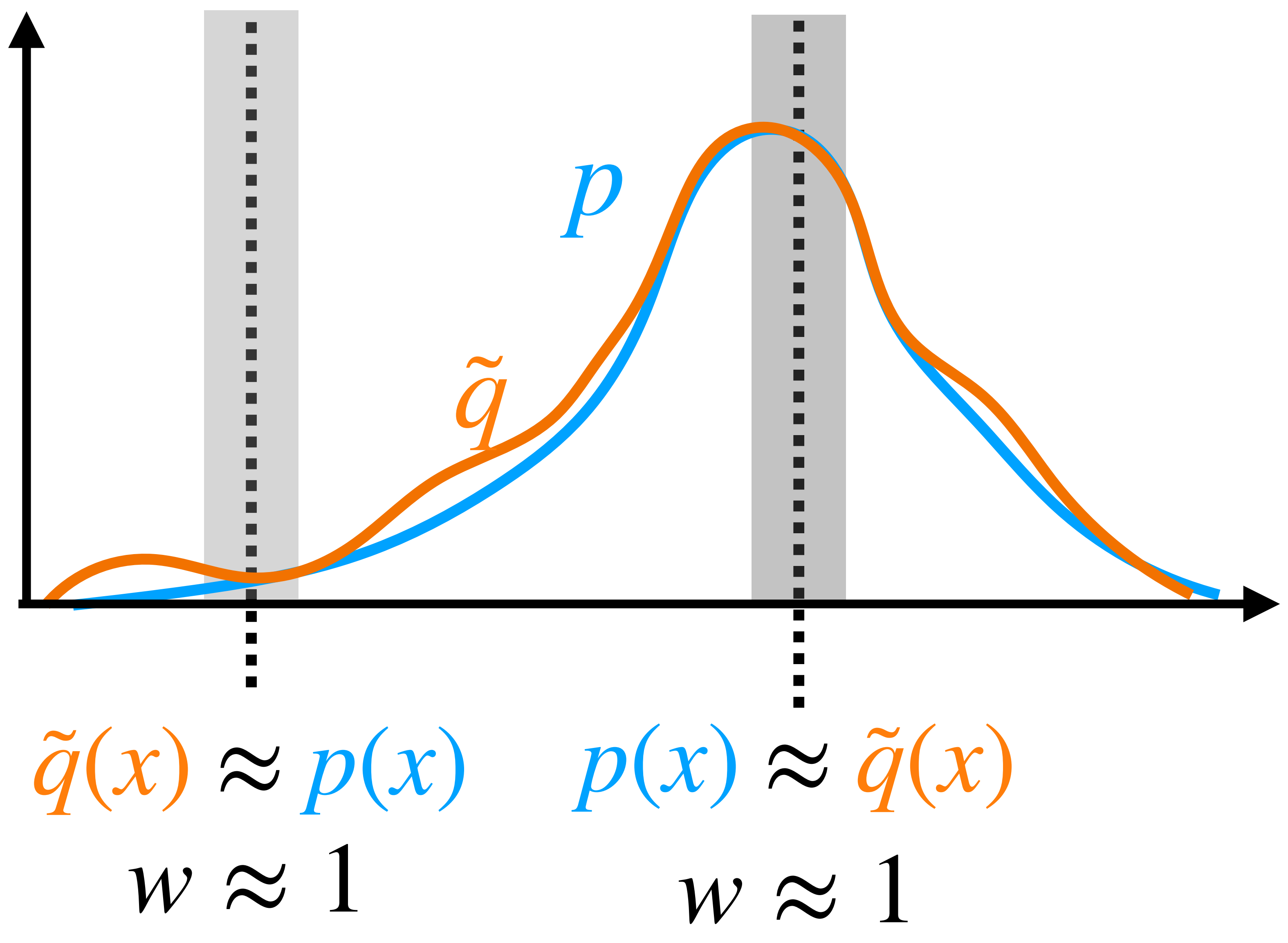
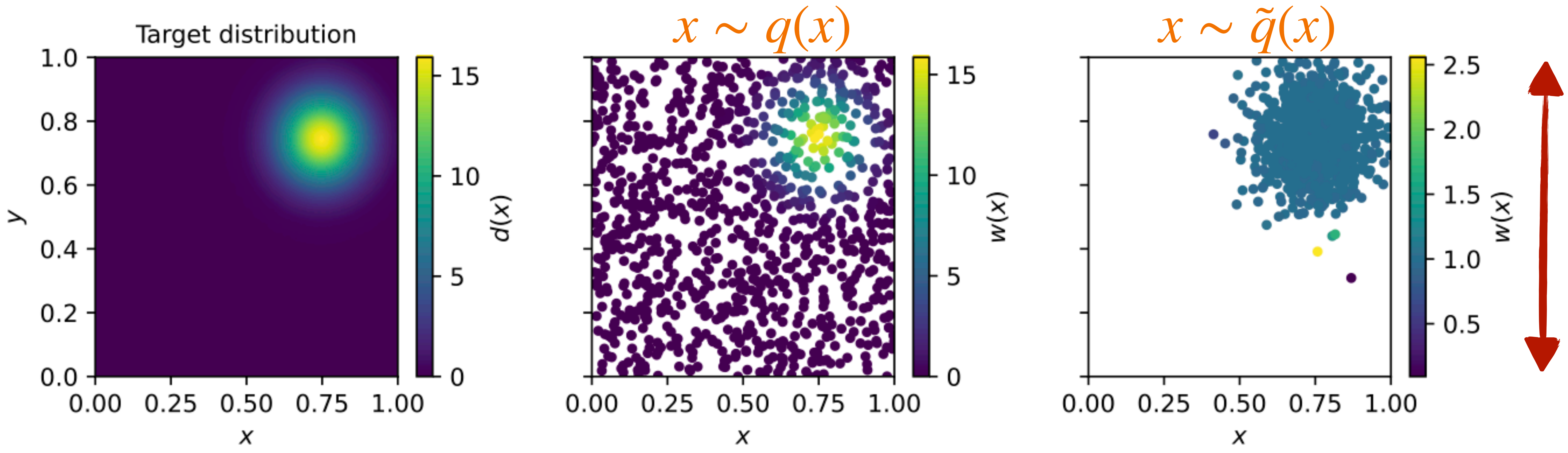


Illustration of importance weights

- Two distributions
  - Target distribution = 2D Gaussian $p(x)$
  - Proposal = Optimized distribution $\tilde{q}(x)$
- Importance weight $w(x) = \dfrac{p(x)}{q(x)}$





Small variance in $w(x)$

How do we get the true posterior?

- Assumption so far: $p(\theta \mid d)$ is known
- Not the case!
- Apply Bayes' theorem:

$$\underbrace{p(\theta \mid d)}_{\text{Posterior}} = \frac{\overbrace{p(d \mid \theta)}^{\text{Likelihood}}}{\underbrace{p(d)}_{\text{Evidence}}} \underbrace{p(\theta)}_{\text{Prior}} \propto p(d \mid \theta) \cdot p(\theta)$$

- Importance weight:

Known because of noise assumption

Known

$$w_i = \frac{p(\theta_i \mid d)}{q(\theta_i \mid d)} \propto \frac{p(d \mid \theta_i) \cdot p(\theta_i)}{q(\theta_i \mid d)} = \frac{\text{Likelihood} \cdot \text{Prior}}{\text{Proposal posterior}}$$

Known in NPE

Dax+, PRL 2023

Likelihood in GW posterior estimation

- Common assumptions about the likelihood $p(d\,|\,\theta)$
  - Stationary, Gaussian noise
  - Uncorrelated between detectors
  - "Whittle likelihood":

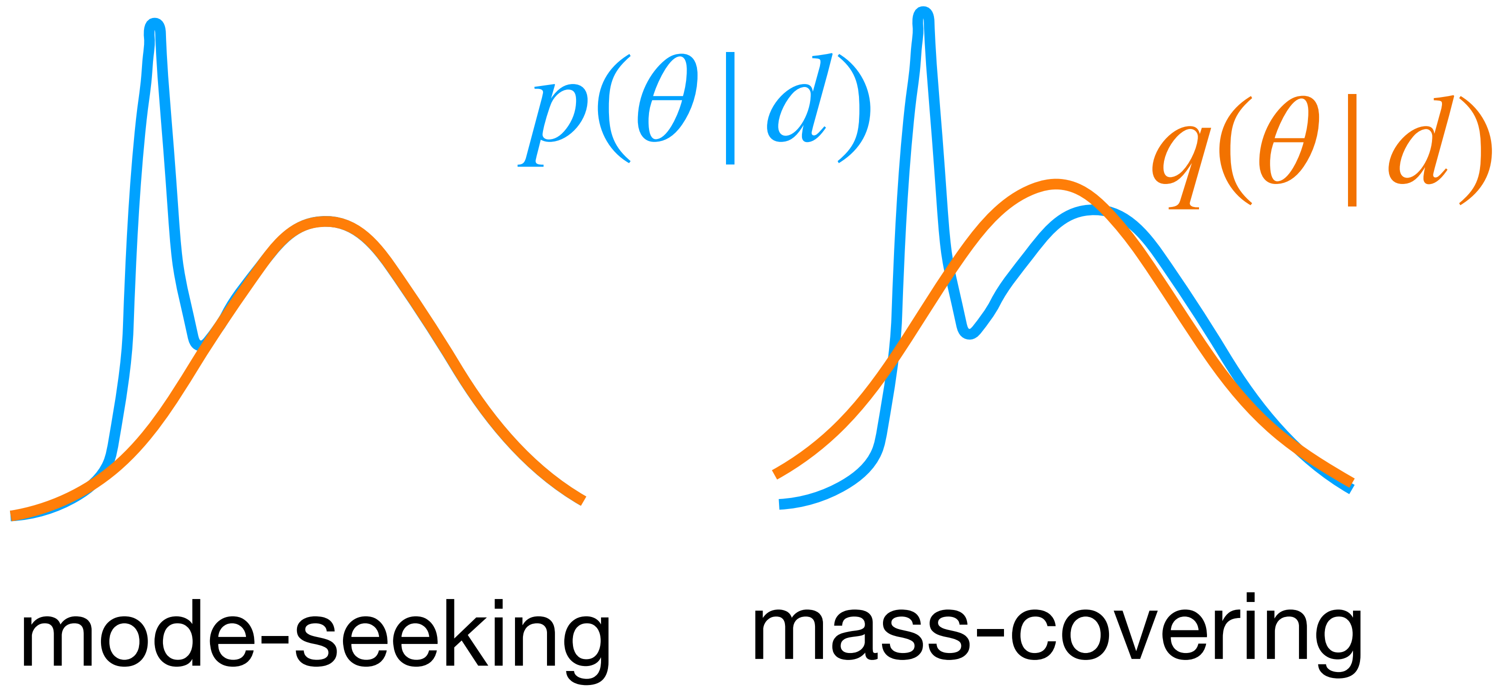
$$p(d\,|\,\theta) = \exp\left(-\frac{1}{2}\sum_I \left[\langle d_I - h_I(\theta)\,|\,d_I - h_I(\theta)\rangle + \int \ln(S_n^I(f))\,\mathrm{d}f\right]\right)$$

with noise-weighted inner product

$$\langle a\,|\,b\rangle = 2\int_0^\infty \frac{\tilde{a}(f)\tilde{b}^*(f) + \tilde{a}^*(f)\tilde{b}(f)}{S_n(f)}\mathrm{d}f$$

Abbot+, 2020, A guide to LIGO-Virgo detector noise

Requirement for proposal: Mass-covering



- What does mass-covering mean?
  - $q(\theta|d)$ covers full support of distribution $p(\theta|d)$
- Why is NPE training mass-covering?
  - Loss function based on forward KL divergence: $\mathrm{KL}\left(p(\theta|d)\,||\,q(\theta|d)\right)$ $\rightarrow$ mass-covering properties
  - NPE proposal expected to cover entire posterior support
- Why is this relevant for importance sampling?
  - Evidence estimate $p(d)$ only unbiased if proposal $q(\theta|d)$ covers support of full posterior distribution $p(\theta|d)$

Minka, 2005, Divergence measures and message passing

Dax+, PRL 2023

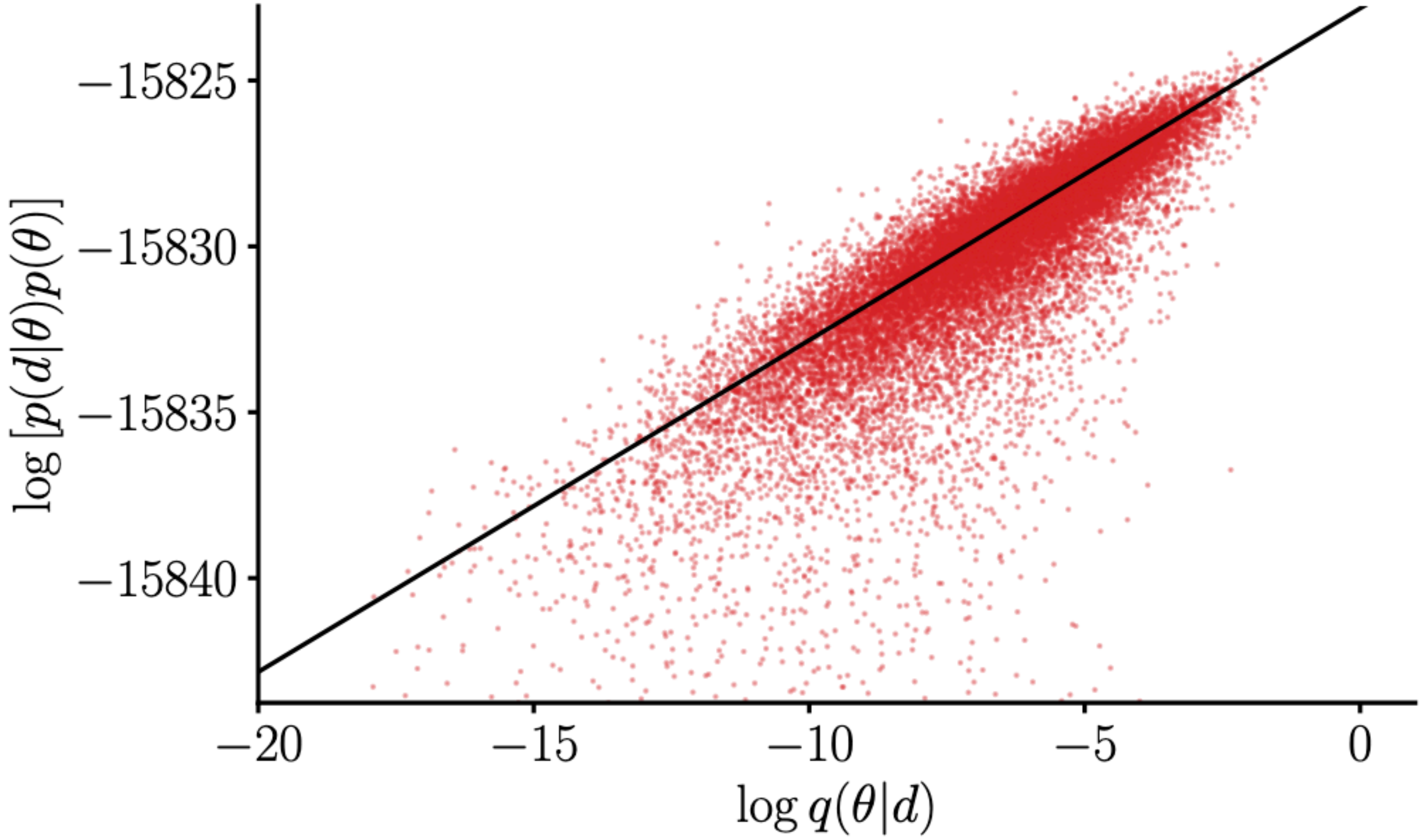
How can we visualize the importance weights?

- Plot nominator vs. denominator

$$w_i \propto \frac{p(d\,|\,\theta_i) \cdot p(\theta_i)}{q(\theta_i\,|\,d)}$$

- Perfect NPE model:
  All samples on black line

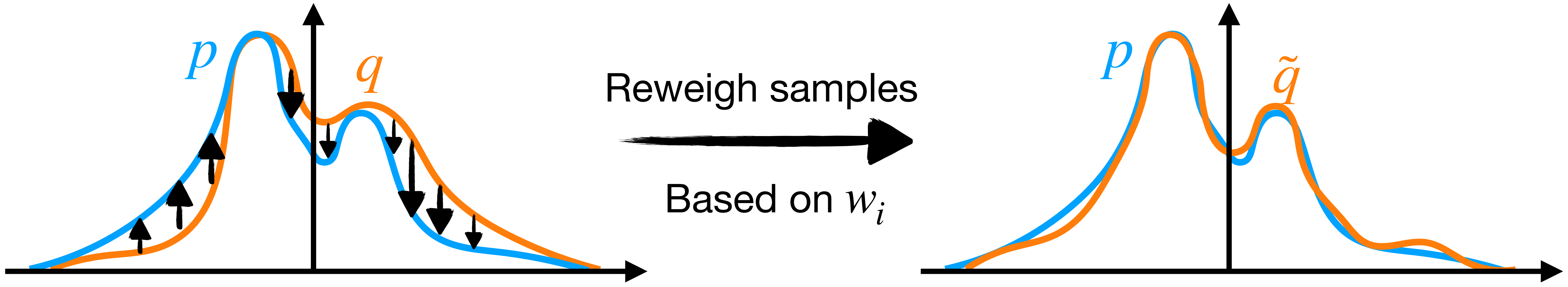
- Most samples below black line
  Reason: mass-covering behavior



Dax+, PRL 2023

What can we do with the importance weights?

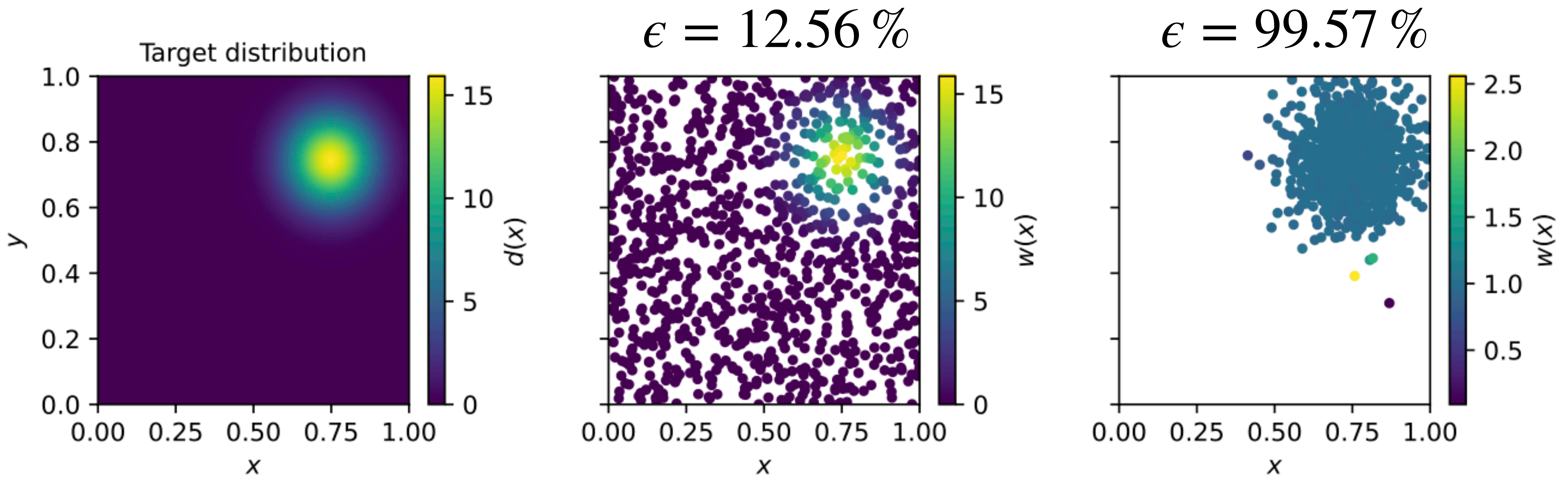
- Reweigh posterior samples $\{\theta_i\}$ towards true posterior $p(\theta \mid d)$



What can we do with the importance weights?

- Reweigh posterior samples $\{\theta_i\}$ towards true posterior $p(\theta \mid d)$

- Compute sampling efficiency: $$\epsilon = \frac{1}{N} \frac{\left(\sum_{i=1}^{N} w_i\right)^2}{\sum_{i=1}^{N} w_i^2} \quad \in [0,1]$$



Dax+, PRL 2023

Verifying Results: Sampling Efficiency

- Failure cases flagged via low sample efficiency ϵ
- Out-of-distribution data: = inconsistent with noise or signal model

→ Identification of events with known issues with data quality or modeling

Event	$\log p(d)$	ϵ	Event	$\log p(d)$	ϵ	Event	$\log p(d)$	ϵ
GW190408	-16178.332 ± 0.012	6.9%	GW190727	-15992.017 ± 0.009	10.3%	GW191230	-15913.798 ± 0.009	12.2%
_181802	-16178.172 ± 0.010	9.3%	_060333	-15992.428 ± 0.005	30.8%	_180458	-15913.918 ± 0.010	8.8%
GW190413	-15571.413 ± 0.006	22.5%	GW190731	-16376.777 ± 0.005	32.6%	GW200128	-16305.128 ± 0.013	6.1%
_052954	-15571.391 ± 0.005	26.3%	_140936	-16376.763 ± 0.005	31.0%	_022011	-16304.510 ± 0.007	18.3%
GW190413	-16399.331 ± 0.009	12.4%	GW190803	-16132.409 ± 0.006	21.4%	‡GW200129	-16226.851 ± 0.109	0.1%
_134308	-16399.139 ± 0.014	4.7%	_022701	-16132.408 ± 0.005	27.8%	_065458	-16231.203 ± 0.051	0.4%
GW190421	-15983.248 ± 0.008	15.3%	GW190805	-16073.261 ± 0.006	20.0%	GW200208	-16136.381 ± 0.007	16.6%
_213856	-15983.131 ± 0.010	9.4%	_211137	-16073.656 ± 0.007	16.6%	_130117	-16136.531 ± 0.009	11.2%
GW190503	-16582.865 ± 0.022	2.0%	GW190828	-16137.220 ± 0.009	12.2%	GW200208	-16775.200 ± 0.011	7.4%
_185404	-16583.352 ± 0.027	1.4%	_063405	-16136.799 ± 0.010	9.1%	_222617	-16774.582 ± 0.021	2.2%
GW190513	-15946.462 ± 0.043	0.6%	GW190909	-16061.634 ± 0.011	7.4%	GW200209	-16383.847 ± 0.009	12.5%
_205428	-15946.581 ± 0.017	3.4%	_114149	-16061.275 ± 0.016	3.8%	_085452	-16384.157 ± 0.025	1.6%
GW190514	-16556.466 ± 0.009	11.6%	GW190915	-16083.960 ± 0.015	20.8%	GW200216	-16215.703 ± 0.017	3.4%
_065416	-16556.314 ± 0.017	3.5%	_235702	-16083.937 ± 0.027	4.8%	_220804	-16215.540 ± 0.018	3.1%
GW190517	-16271.048 ± 0.027	1.3%	GW190926	-16015.813 ± 0.019	2.8%	GW200219	-16133.457 ± 0.011	9.6%
_055101	-16272.428 ± 0.034	0.9%	_050336	-16015.861 ± 0.009	12.1%	_094415	-16133.157 ± 0.017	4.0%
GW190519	-15991.171 ± 0.008	15.2%	GW190929	-16146.666 ± 0.018	3.2%	GW200220	-16303.782 ± 0.007	17.3%
_153544	-15991.287 ± 0.068	0.2%	_012149	-16146.591 ± 0.021	2.4%	_061928	-16303.087 ± 0.026	1.5%
GW190521	-16008.876 ± 0.008	13.4%	GW191109	-17925.064 ± 0.025	1.7%	GW200220	-16136.600 ± 0.008	13.2%
_074359	-16008.037 ± 0.015	4.2%	_010717	-17922.762 ± 0.041	0.6%	_124850	-16136.519 ± 0.037	0.7%
GW190527	-16119.012 ± 0.008	13.8%	GW191127	-16759.328 ± 0.019	2.7%	GW200224	-16138.613 ± 0.006	22.5%
_092055	-16118.781 ± 0.013	6.1%	_050227	-16758.102 ± 0.029	1.2%	_222234	-16139.101 ± 0.006	21.4%
GW190602	-16036.993 ± 0.006	25.0%	‡GW191204	-15984.455 ± 0.015	4.2%	‡GW200308	-16173.938 ± 0.013	6.0%
_175927	-16037.529 ± 0.006	23.5%	_110529	-15983.618 ± 0.063	0.3%	_173609	-16173.692 ± 0.025	1.7%
GW190701	-16521.381 ± 0.040	0.6%	GW191215	-16001.286 ± 0.013	5.8%	GW200311	-16117.505 ± 0.011	7.4%
_203306	-16521.609 ± 0.010	10.1%	_223052	-16000.846 ± 0.052	0.4%	_115853	-16117.583 ± 0.009	11.9%
GW190719	-15850.492 ± 0.008	13.4%	GW191222	-15871.521 ± 0.007	16.5%	‡GW200322	-16313.568 ± 0.307	0.0%
_215514	-15850.339 ± 0.011	8.0%	_033537	-15871.450 ± 0.005	25.8%	_091133	-16313.110 ± 0.105	0.1%

Table II. 42 BBH events from GWTC-3 analyzed with DINGO-IS. We report the log evidence $\log p(d)$ and the sample efficiency ϵ for the two waveform models IMRPhenomXPHM (upper rows) and SEOBNRv4PHM (lower rows). Highlighting colors indicate the sample efficiency (green: high; yellow: medium; orange/red: low); DINGO-IS results can be trusted for medium and high ϵ (see Supplemental Material). Events in gray suffer from data quality issues [1, 21]. ‡See remarks on these events in text.

What can we do with the importance weights?

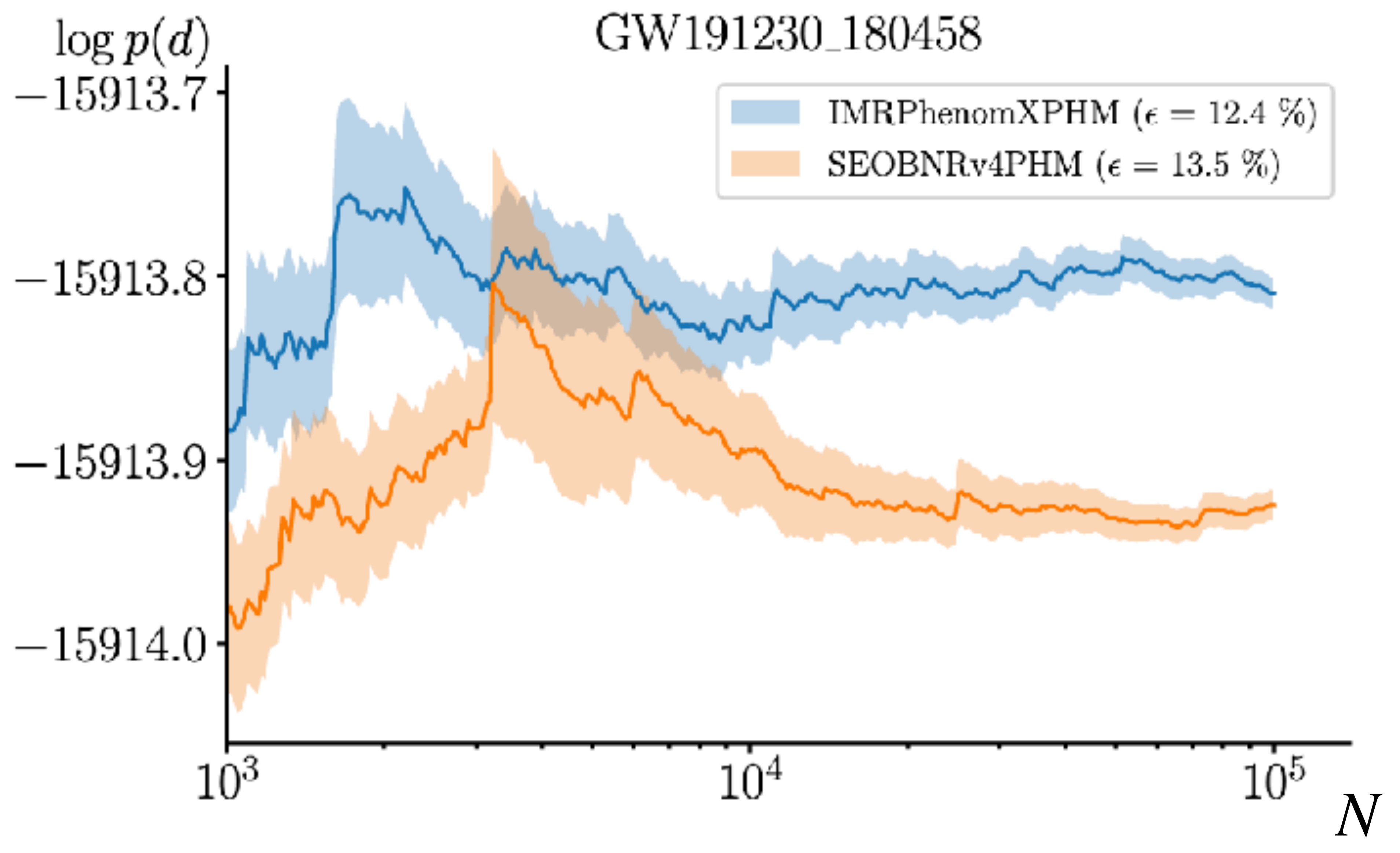
- Reweigh posterior samples $\{\theta_i\}$ towards true posterior $p(\theta\,|\,d)$

- Compute sampling efficiency: $$\epsilon = \frac{1}{N}\frac{\left(\sum_{i=1}^{N} w_i\right)^2}{\sum_{i=1}^{N} w_i^2}$$

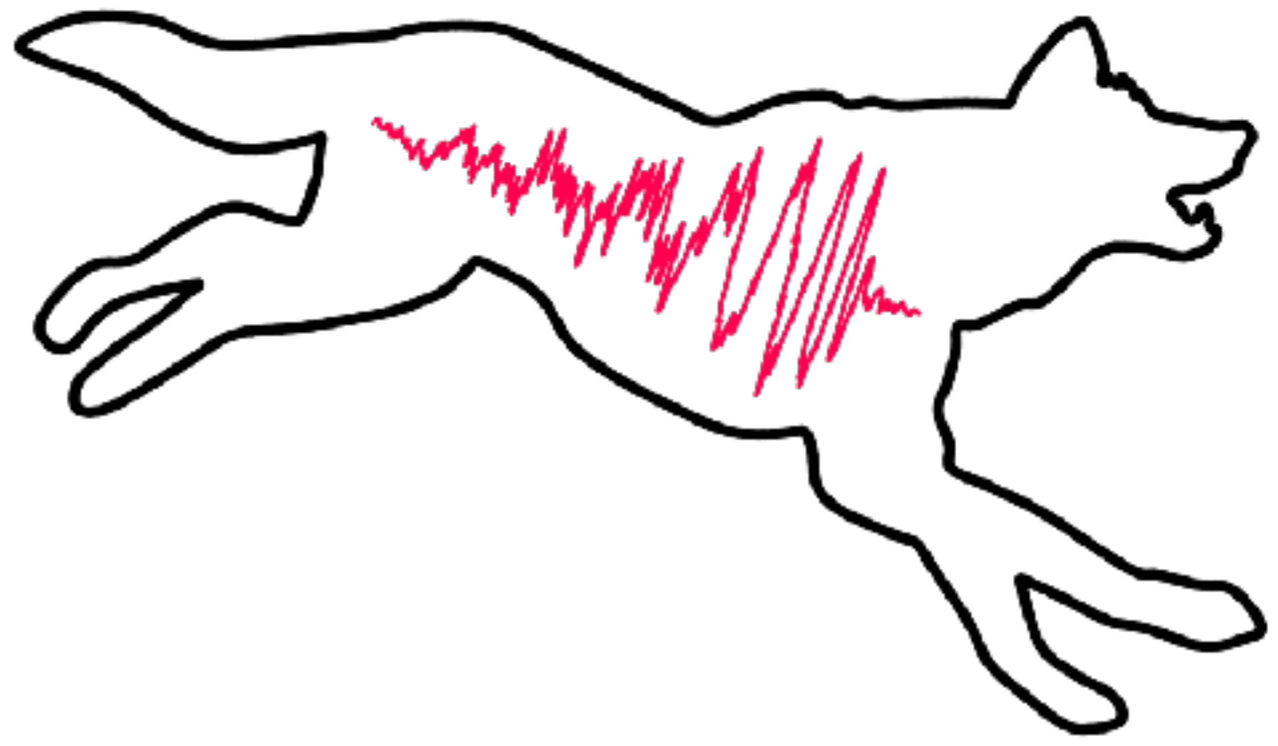
- Compute estimate for the evidence:

$$p(d) \approx \frac{1}{N}\sum_{i=1}^{N} w_i$$

→ important for model comparison
e.g. eccentricity   Gupte+, arXiv 2024



Dax+, PRL 2023

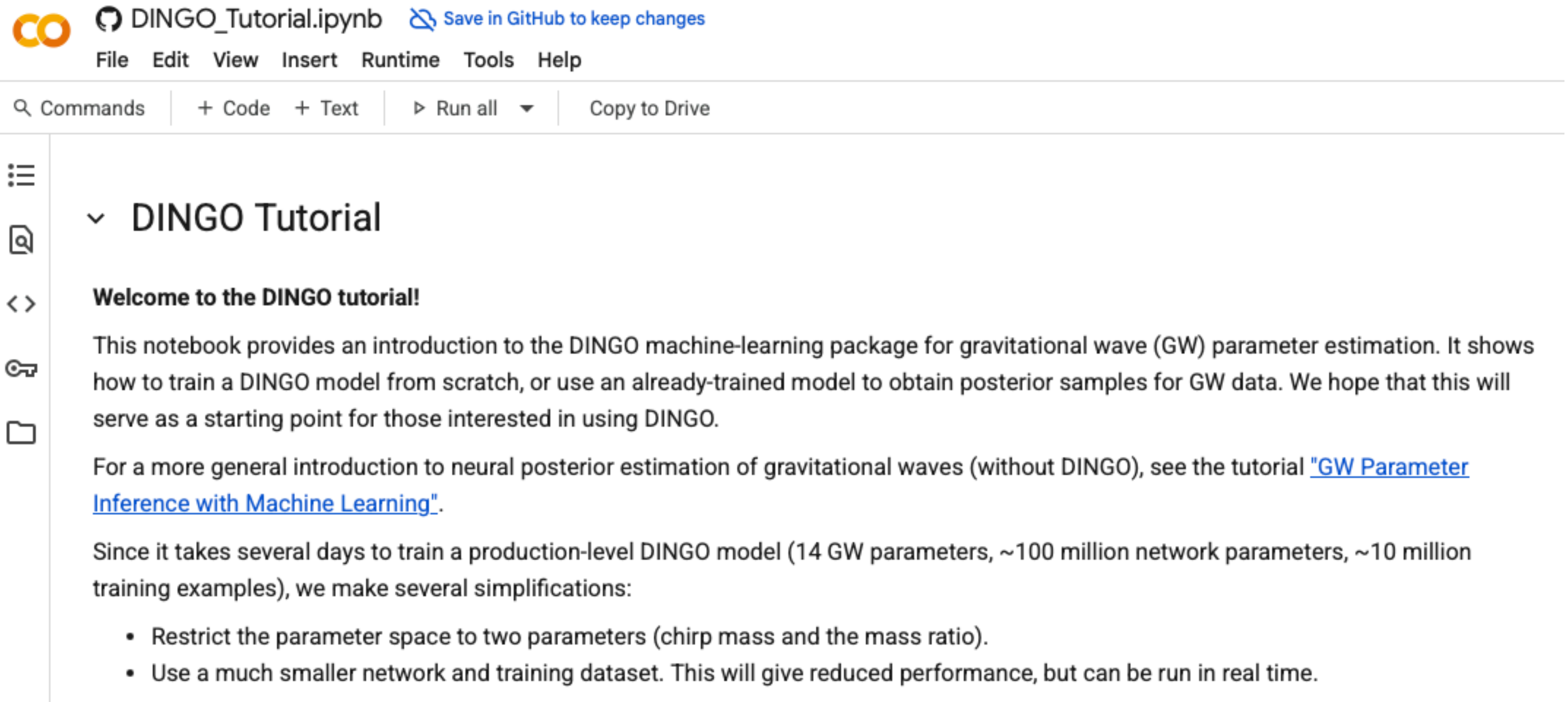


Implementation of DINGO-IS

Code overview

DINGO Tutorial

- Introductory DINGO tutorial: https://github.com/annalena-k/tutorial-dingo-introduction
- Shown by Nihar Gupte in his last talk at your seminar



Inference with DINGO

- Usually via **dingo_pipe** (equivalent to bilby_pipe)
  1. Downloads data from GWOSC
  2. Generates posterior samples from trained DINGO model
  3. Performs importance sampling
  4. Prepares diagnostic plots
- Easy to run on cluster based on .ini file
  → See DINGO tutorial for example
- In this talk: more details about importance sampling

Steps for importance sampling with DINGO

- Wrap trained DINGO model in dingo.gw.inference.GNPE sampler class

```
sampler = GWSamplerGNPE(
    model=dingo_model,
    init_sampler=init_sampler,
    num_iterations=num_gnpe_iterations,
)
```

- Sample from model $\theta \sim q(\theta \,|\, d, S_n)$ & convert to dingo.gw.result class

```
sampler.run_sampler(num_samples, batch_size=batch_size)

result = sampler.to_result()
```

- Run importance sampling

```
likelihood_kwargs = {}
result.importance_sample(
    num_processes=num_parallel_processes, **likelihood_kwargs
)
```

Importance sampling details

- Build likelihood function
- Likelihood function
    - Generate waveform based on posterior samples $\theta \sim q(\theta \mid d, S_n)$
    - Compute likelihood $\log p(d \mid \theta) = \psi - \frac{1}{2}\langle d - \mu(\theta), d - \mu(\theta) \rangle$
    - Calculate importance weights
- Compute sampling efficiency, Bayesian evidence, …

Importance sampling details

dingo.gw.likelihood

```
class StationaryGaussianGWLikelihood(GWSignal, Likelihood):
    """
    Implements GW likelihood for stationary, Gaussian noise.
    """
```

- Likelihood computation:
  - For every detector: $$\log p(d\,|\,\theta) \propto -\frac{1}{2}\langle d - h(\theta), d - h(\theta)\rangle$$

$$\log p(d\,|\,\theta) \propto -\frac{1}{2}\left(\log Z_n + \kappa^2(\theta) - \frac{1}{2}\rho_{\mathrm{opt}}^2(\theta)\right)$$

With $-2\log Z_n = \langle d, d\rangle \qquad \kappa^2(\theta) = \langle d, h(\theta)\rangle$

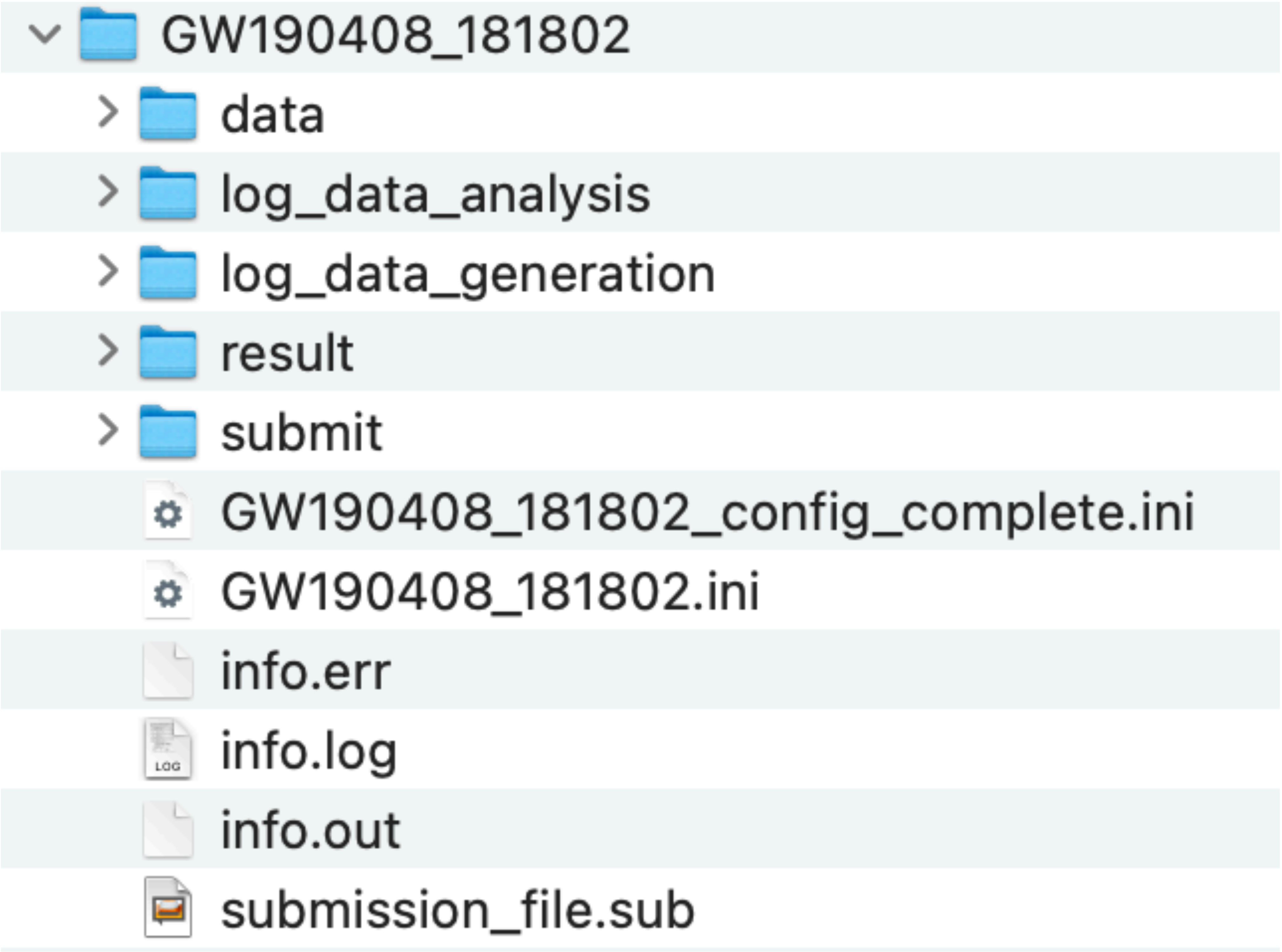
$$\rho_{\mathrm{opt}}^2 \equiv \langle h(\theta), h(\theta)\rangle$$

And the noise-weighted inner product: $$\langle a, b\rangle \equiv 4\Delta f \sum_j \mathcal{R}\left(\frac{a_j^* b_j}{\mathrm{PSD}_j}\right)$$

Based on Appendix B of Thrane & Talbot, 2018, https://arxiv.org/abs/1809.02293

dingo_pipe output

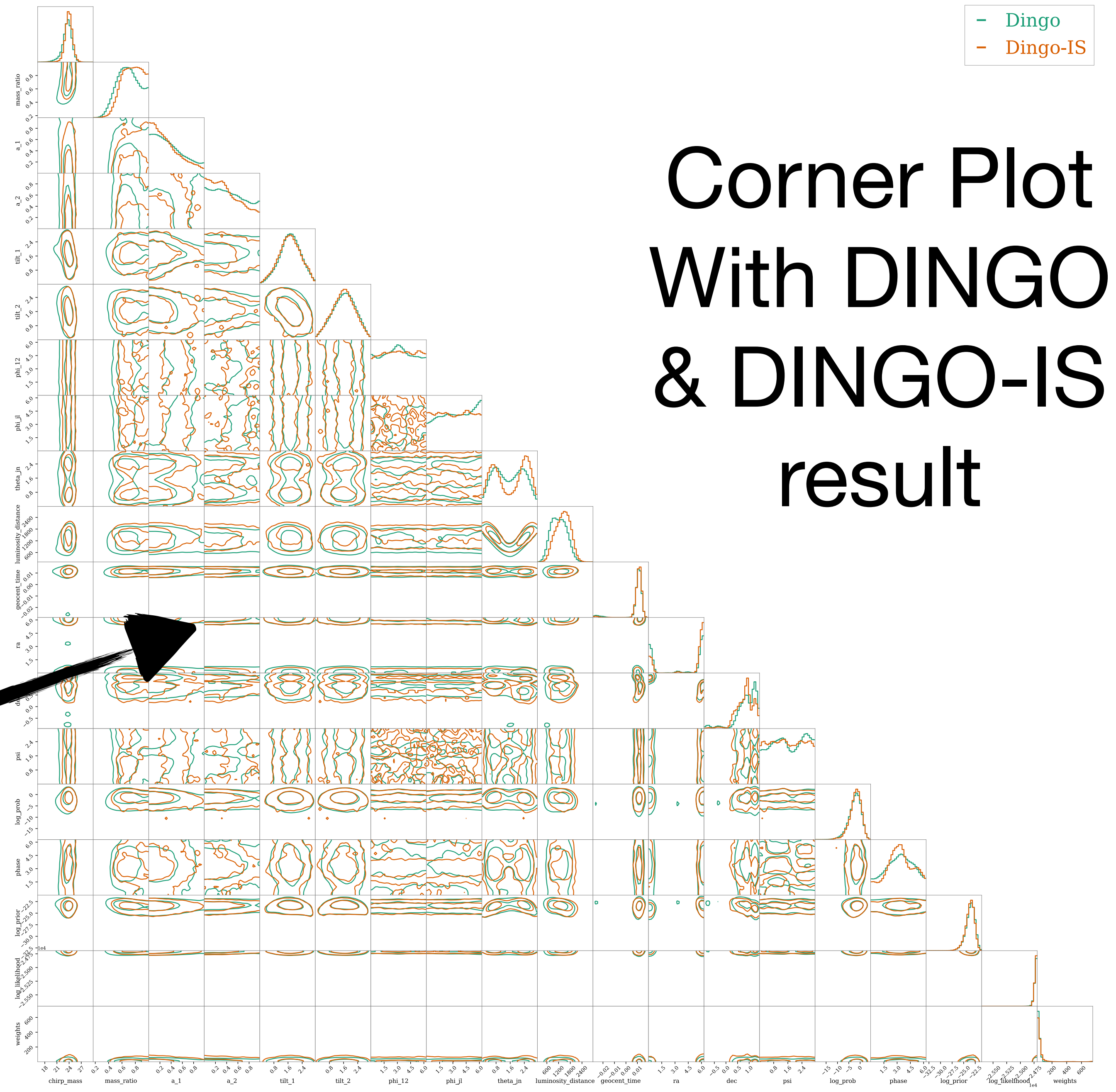
- data: downloaded data
- log_data_generation: .err and .out files for download
- log_data_analysis: .err and .out files for sampling and importance sampling
- result: posterior samples and plots
- submit: everything related to htcondor



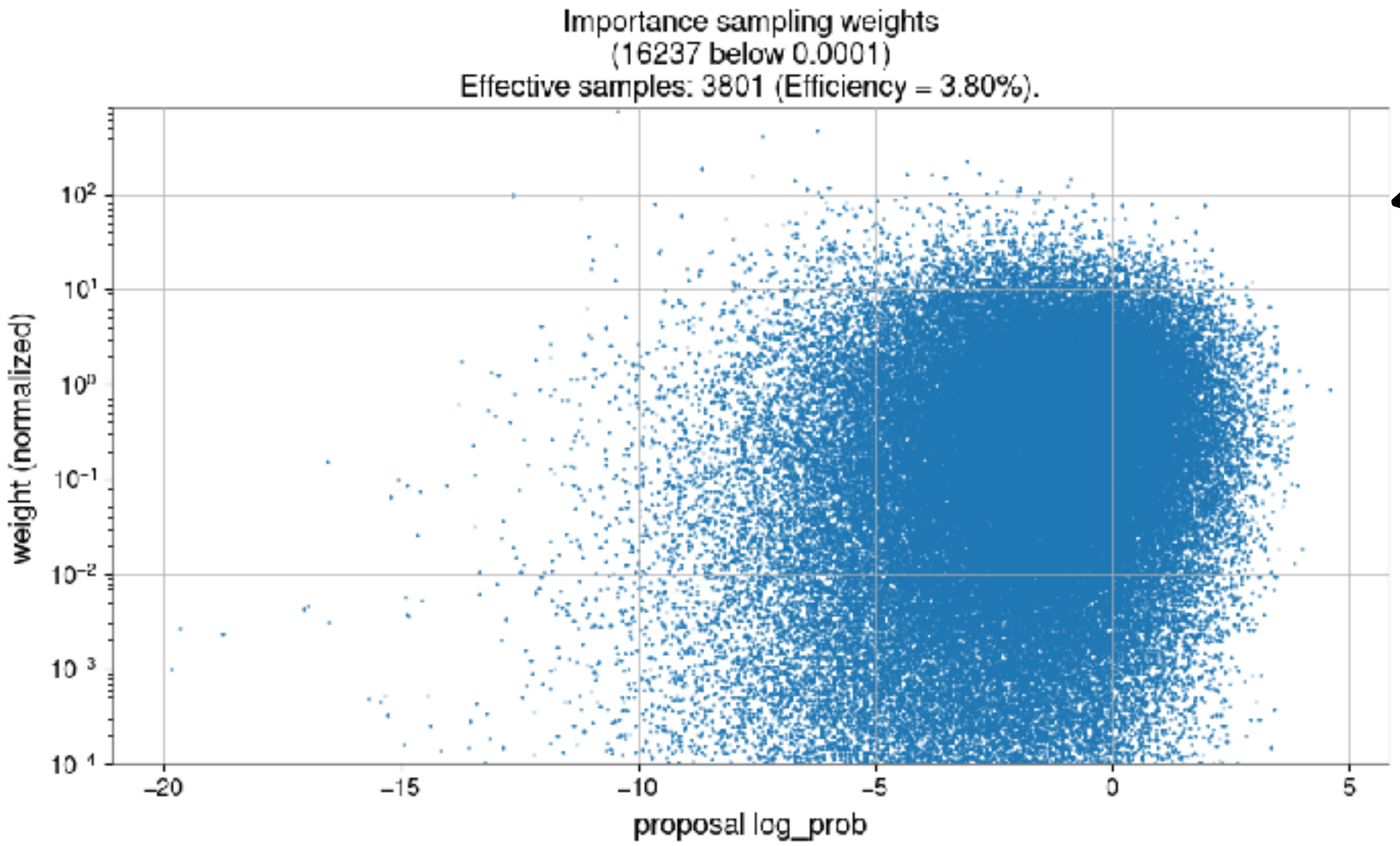
dingo_pipe output

- result: posterior samples and plots

result
- GW190408_181802_data0_1238782700-29_importance_sampling_plot_corner.pdf
- GW190408_181802_data0_1238782700-...portance_sampling_plot_log_probs.png
- GW190408_181802_data0_1238782700-...importance_sampling_plot_weights.png
- GW190408_181802_data0_1238782700-29_importance_sampling.hdf5
- GW190408_181802_data0_1238782700-29_sampling.hdf5

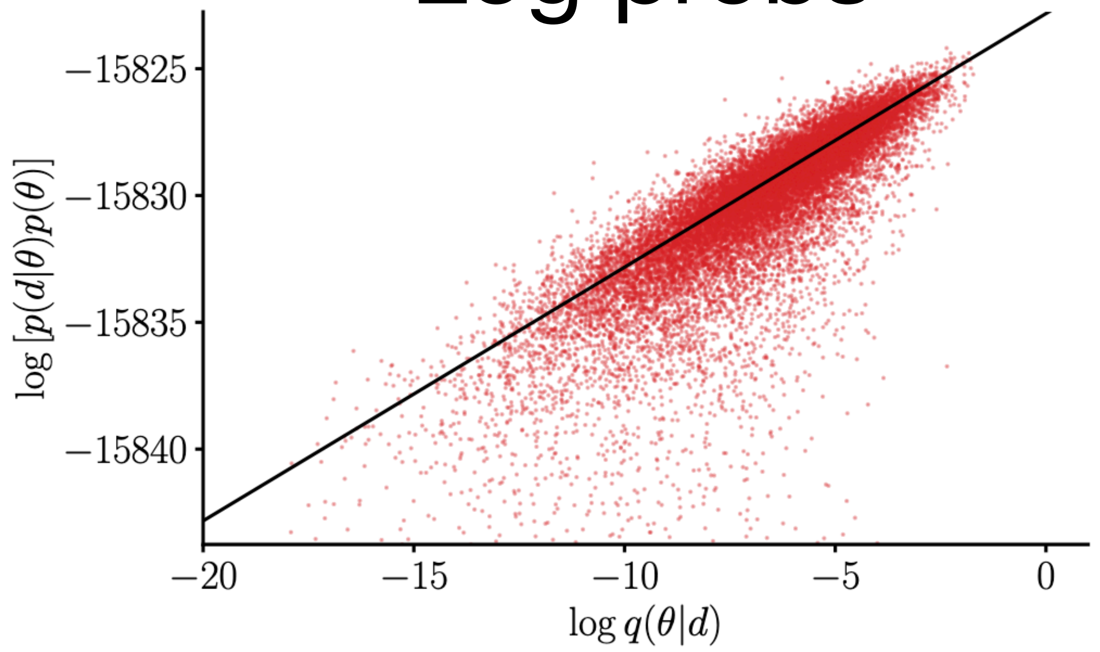


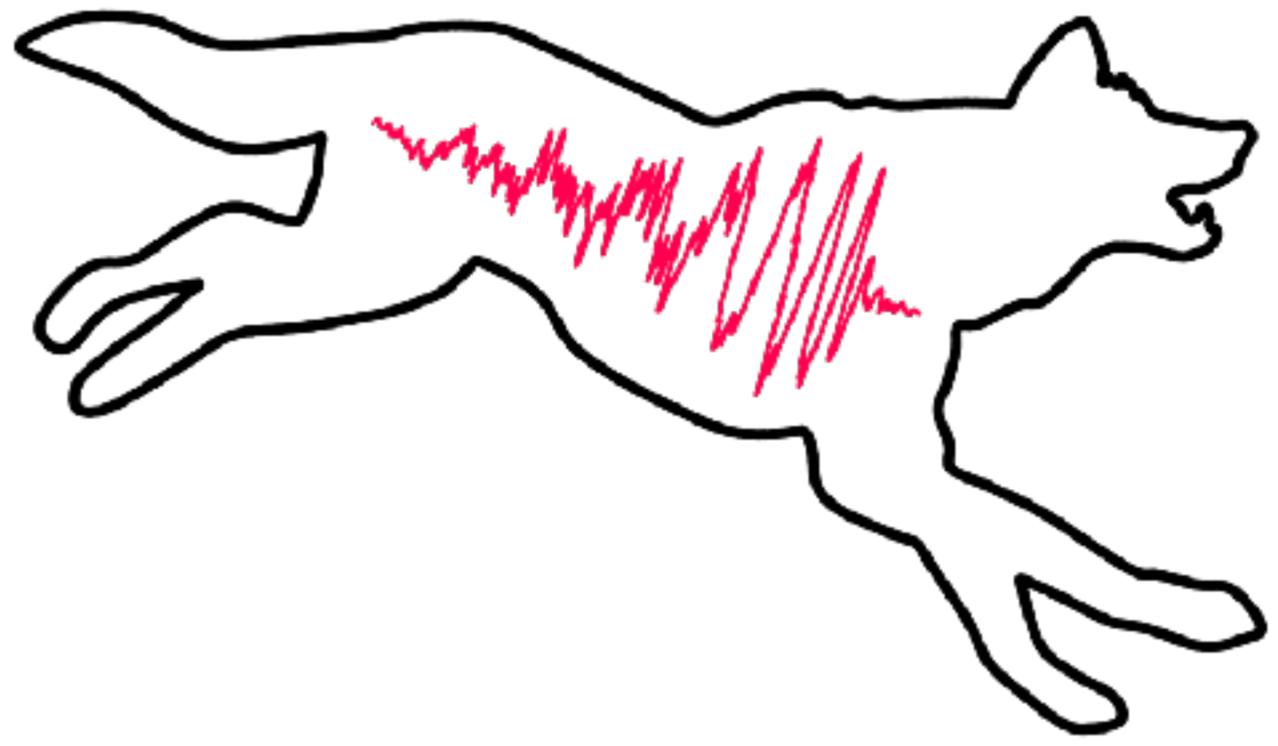
Corner Plot With DINGO & DINGO-IS result



IS weights & sample efficiency

Log probs





Take-aways & usage

Code overview

Where/How is DINGO used?

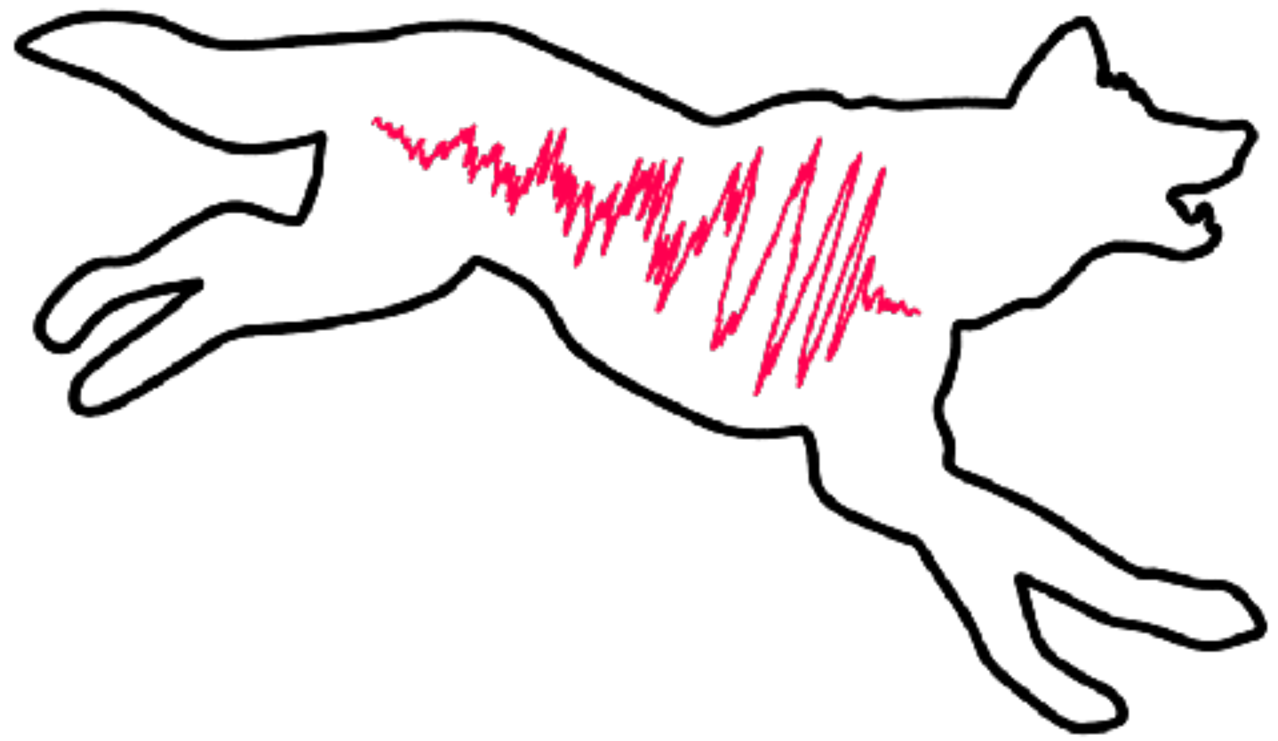
- Analyze large collection of observations
  - Learn population posterior directly — Leyde et al., 2023
  - Evidence for eccentricity — Gupte et al., 2024
- DINGO approved in internal LIGO review 🥳
  → Evaluation of O4 events!
- New paper using pre-trained DINGO models!

Hu, 2025, https://arxiv.org/abs/2507.05209

Wildberger+, PRD 2023

Take-Aways for DINGO

- Accurate inference for BBHs in seconds - minutes
  → Rapid inference of large number of events
- Validation of results with Importance Sampling
- Ready to be used:
  Code @ https://github.com/dingo-gw/dingo, Documentation, Tutorials



The Dingo Pack


Maximilian Dax


Stephen Green


Annalena Kofler


Nihar Gupte


Michael Pürrer


Alex Roussopoulos


Samuel Clyne


Ashwin Girish


Cecilia Fabbri


Jonas Wildberger


Vincent Berenz

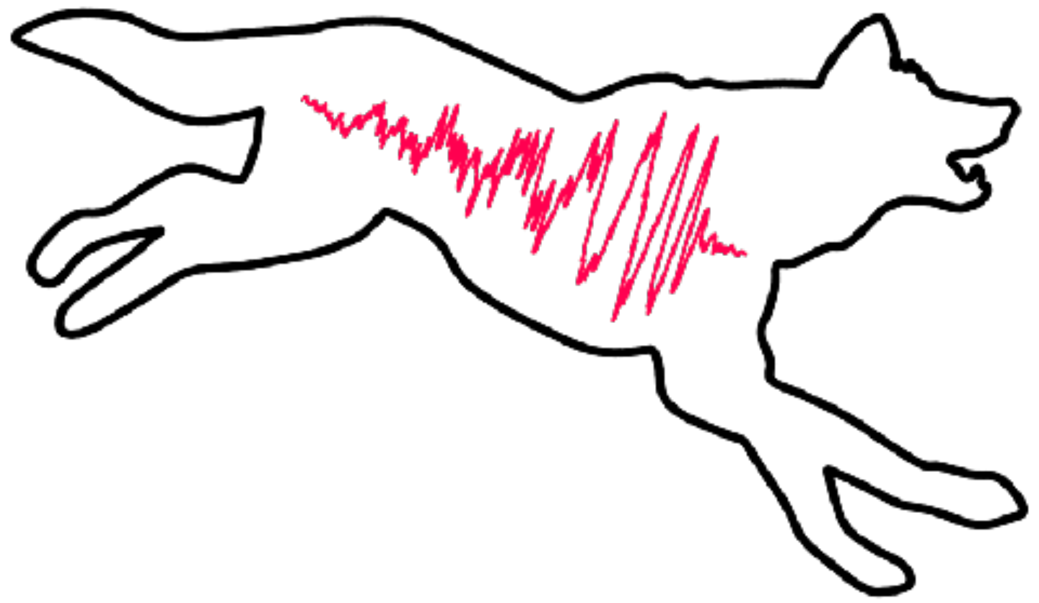

Jonathan Gair


Jakob Macke


Bernhard Schölkopf


Alessandra Buonanno

Thank you!
Do you have any questions?



References

- Dax+, Real-Time Gravitational Wave Science with Neural Posterior Estimation. PRL 127, 2021
- Dax+, Group Equivariant Neural Posterior Estimation, ICLR 2022
- Dax+, Neural Importance Sampling for Rapid and Reliable Gravitational Wave Inference, PRL 130, 2023
- Wildberger+, Adapting to noise distribution shifts in flow-based gravitational-wave inference, PRD 107, 2023
- Wildberger+, Flow Matching for Scalable Simulation-Based Inference, NeurIPS 2023
- Gupte+, Evidence for eccentricity in the population of binary black holes observed by LIGO-Virgo-KAGRA, arXiv:2404.14286v1, 2024
- Dax+, Real-time Gravitational-Wave Inference for Binary Neutron Stars using Machine Learning, Nature, 2025