Deep **learning** the **invisible** universe

Guillermo Franco Abellán

GRAPPA, University of Amsterdam

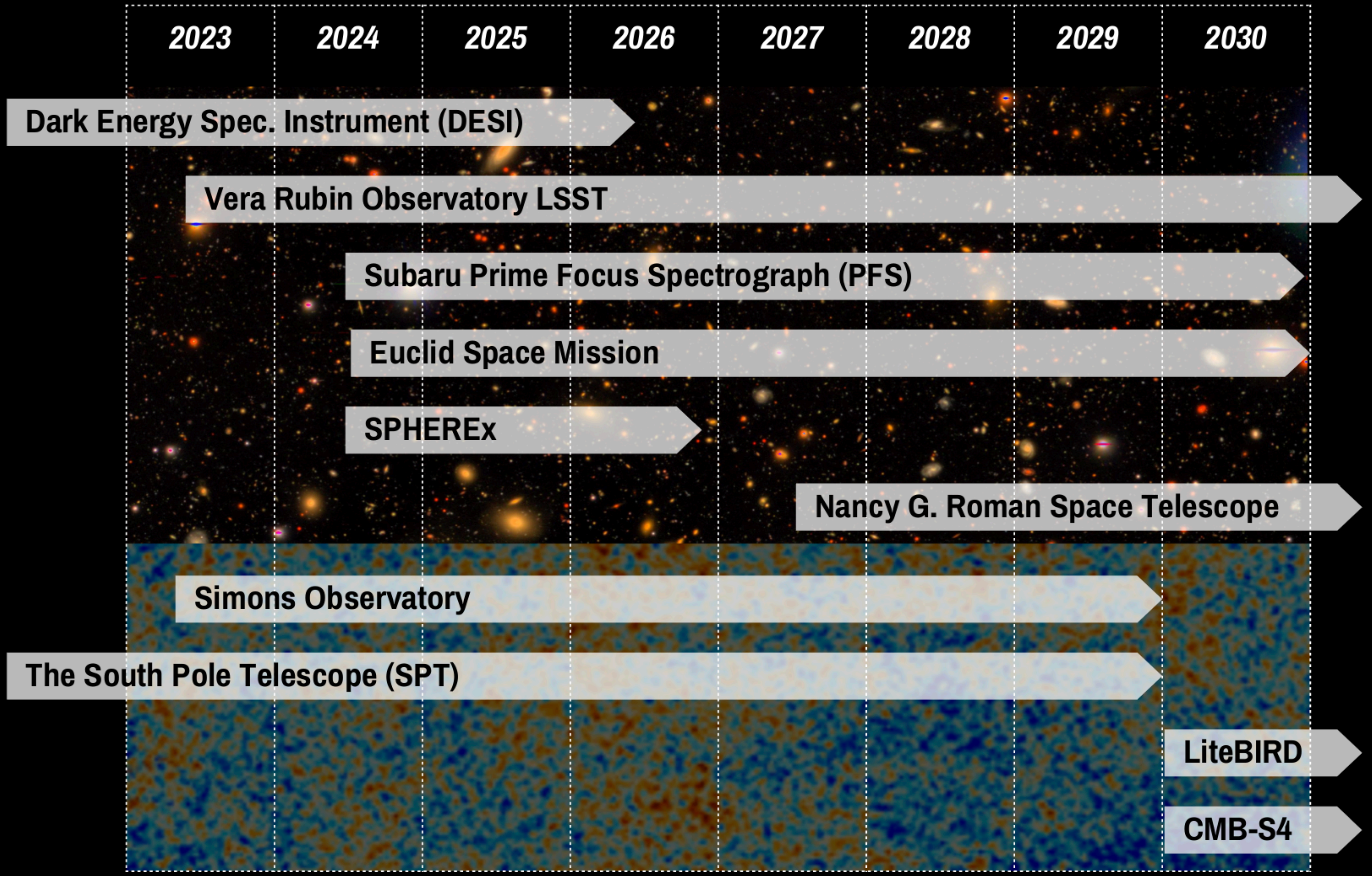
21st May 2025

CEICO Cosmology Seminar

GRAPPA

GRavitation AstroParticle Physics Amsterdam

We are entering in the era of **ultra-high precision** cosmology

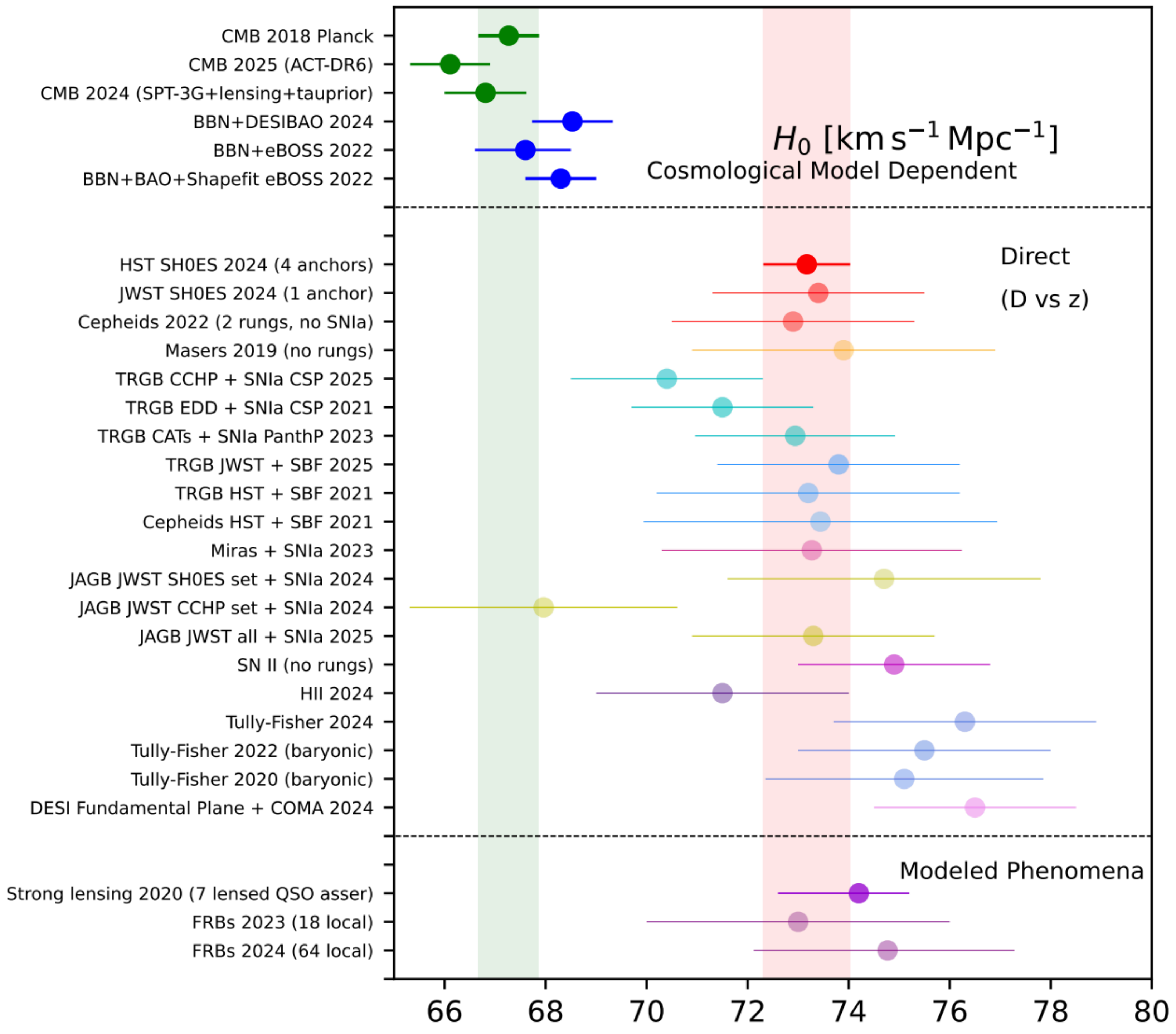


[Credit A. Bayer]

Various datasets point at **cracks in the ΛCDM model**

Various datasets point at **cracks in the ΛCDM model**

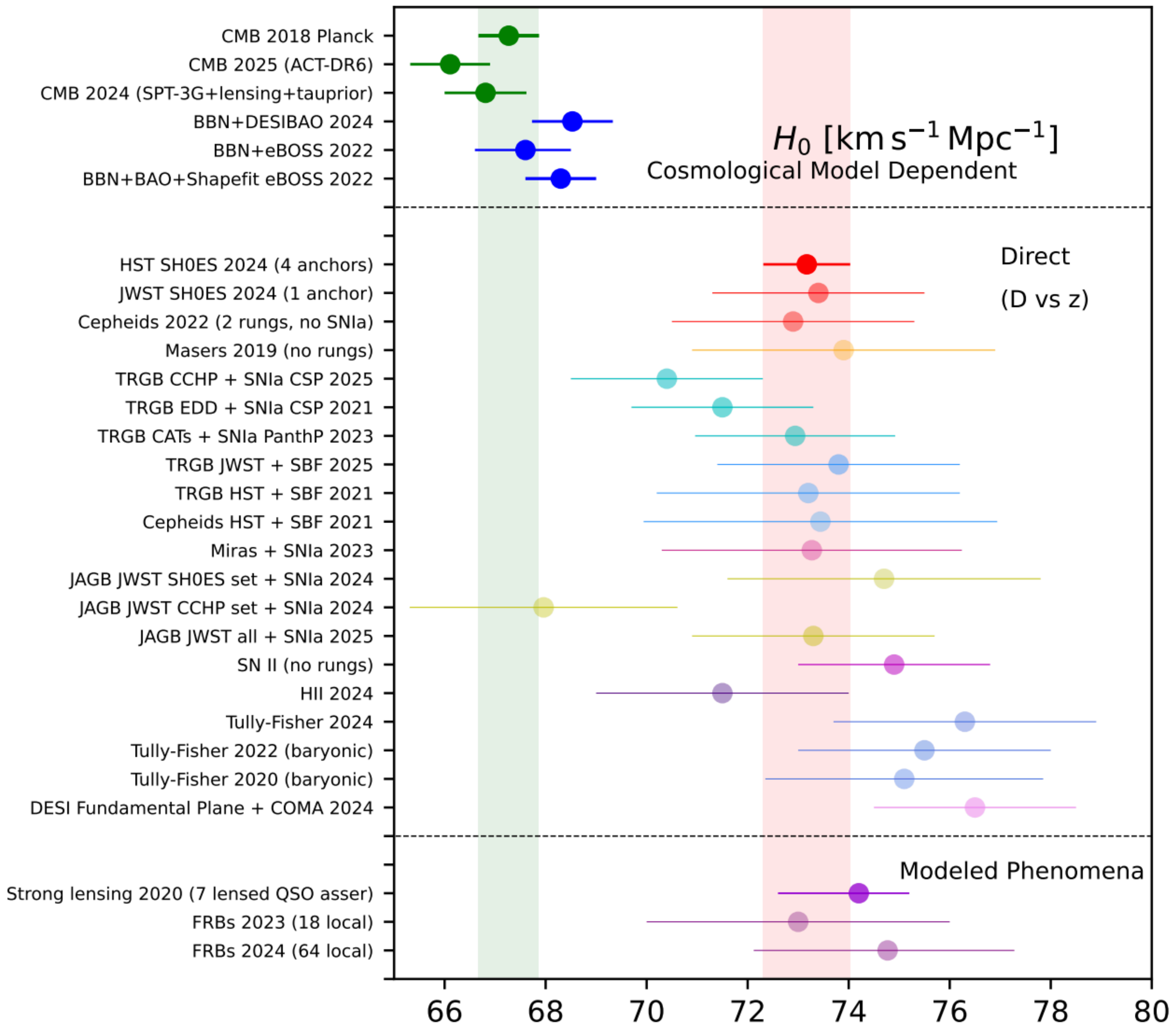
H_0 tension



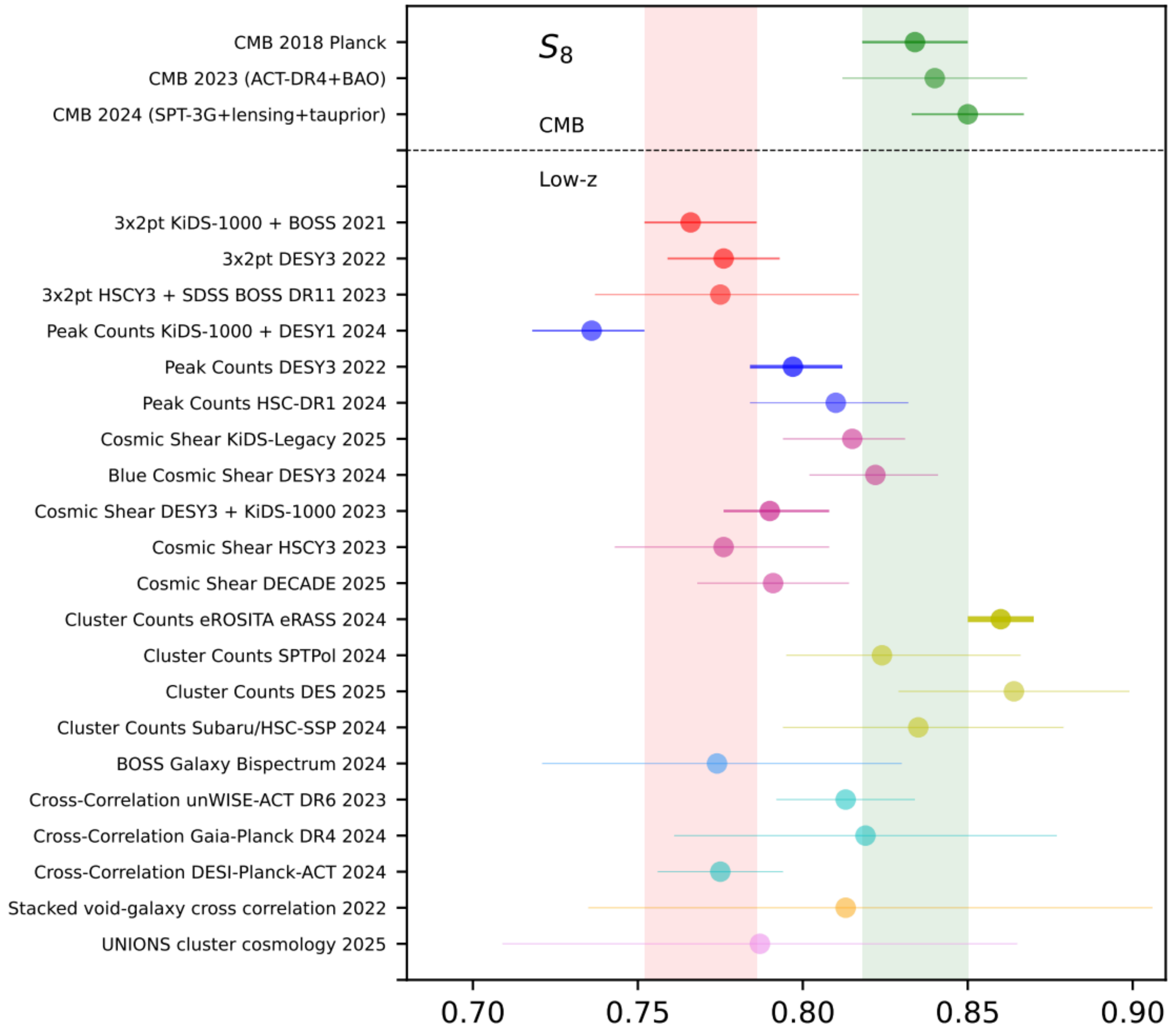
[CosmoVerse White Paper: 2504.01669]

Various datasets point at **cracks in the ΛCDM model**

H_0 tension



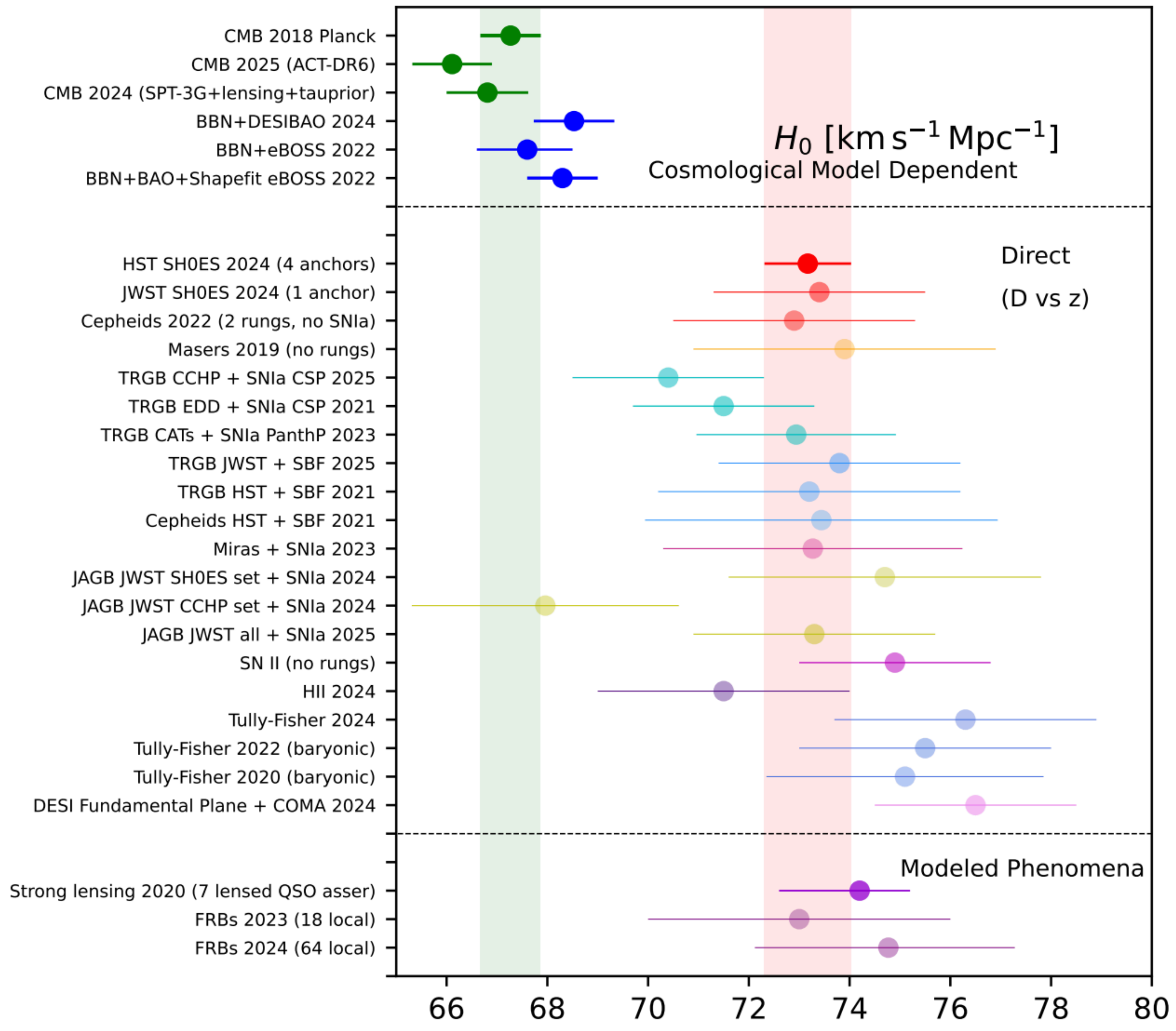
S_8 tension?



[CosmoVerse White Paper: 2504.01669]

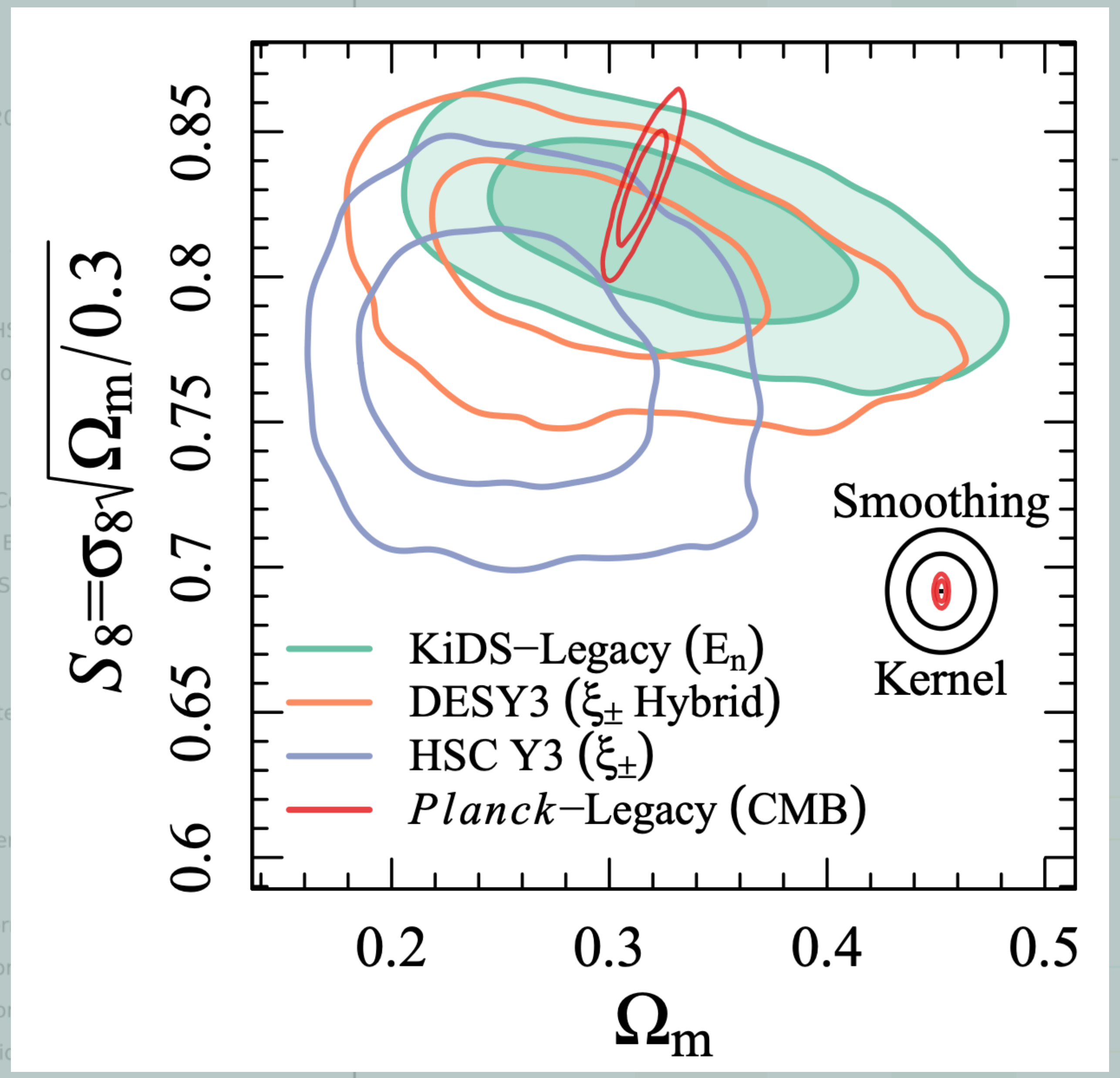
Various datasets point at **cracks in the ΛCDM model**

H_0 tension



[CosmoVerse White Paper: 2504.01669]

S_8 tension?

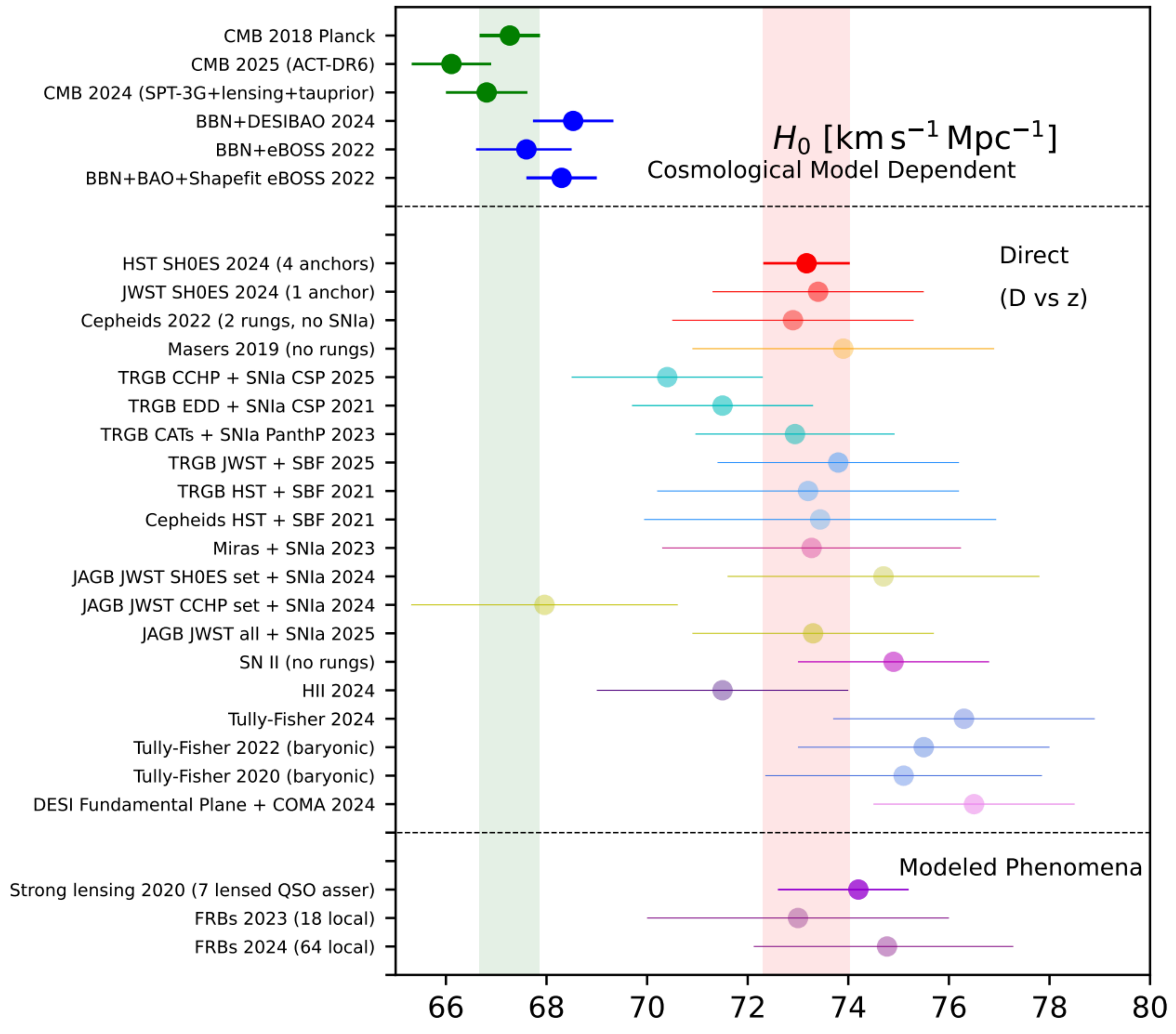


[KiDS-Legacy: 2503.19441]

Seems to be dead after KiDS-Legacy...

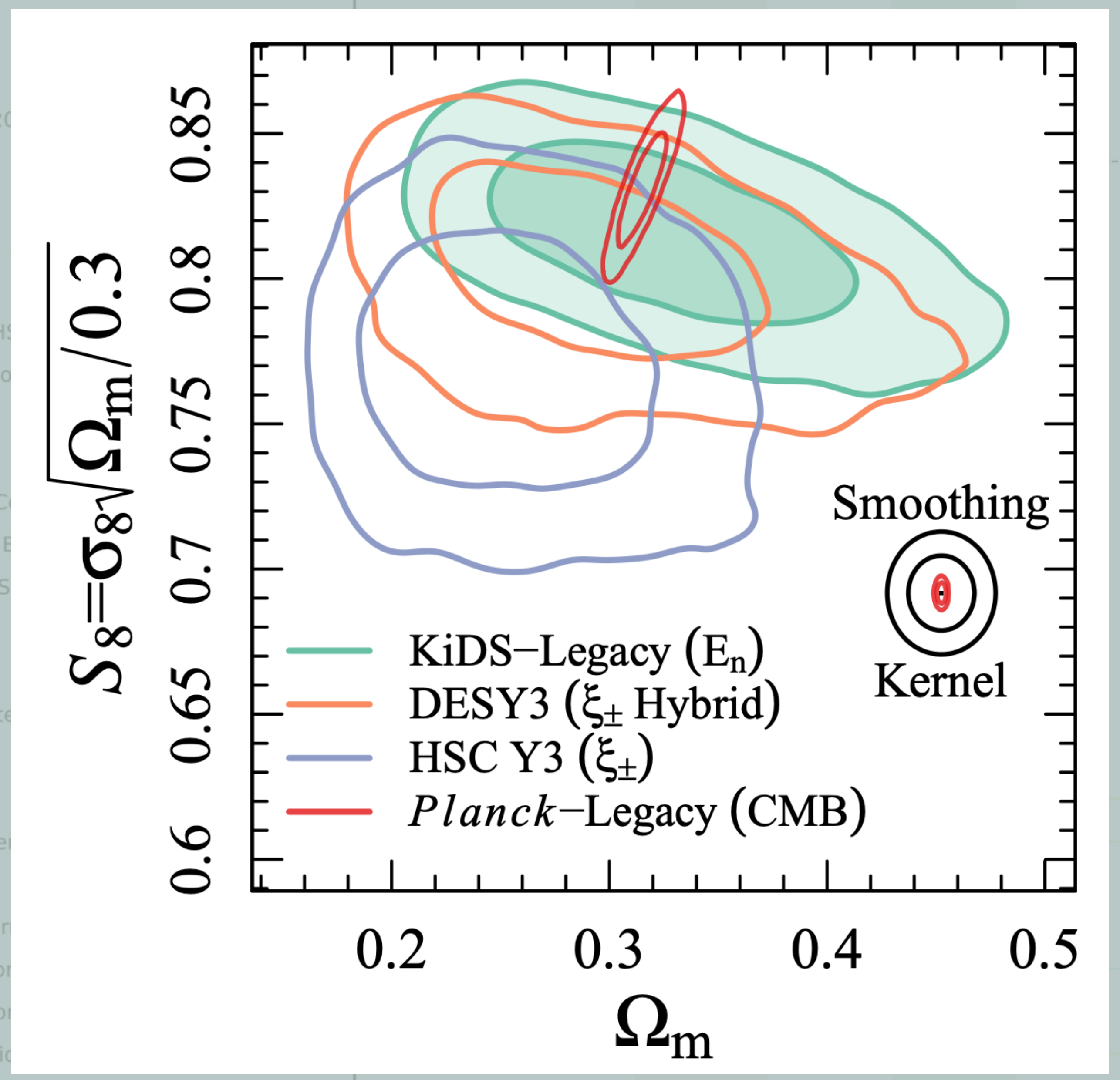
Various datasets point at cracks in the ΛCDM model

H_0 tension



[CosmoVerse White Paper: 2504.01669]

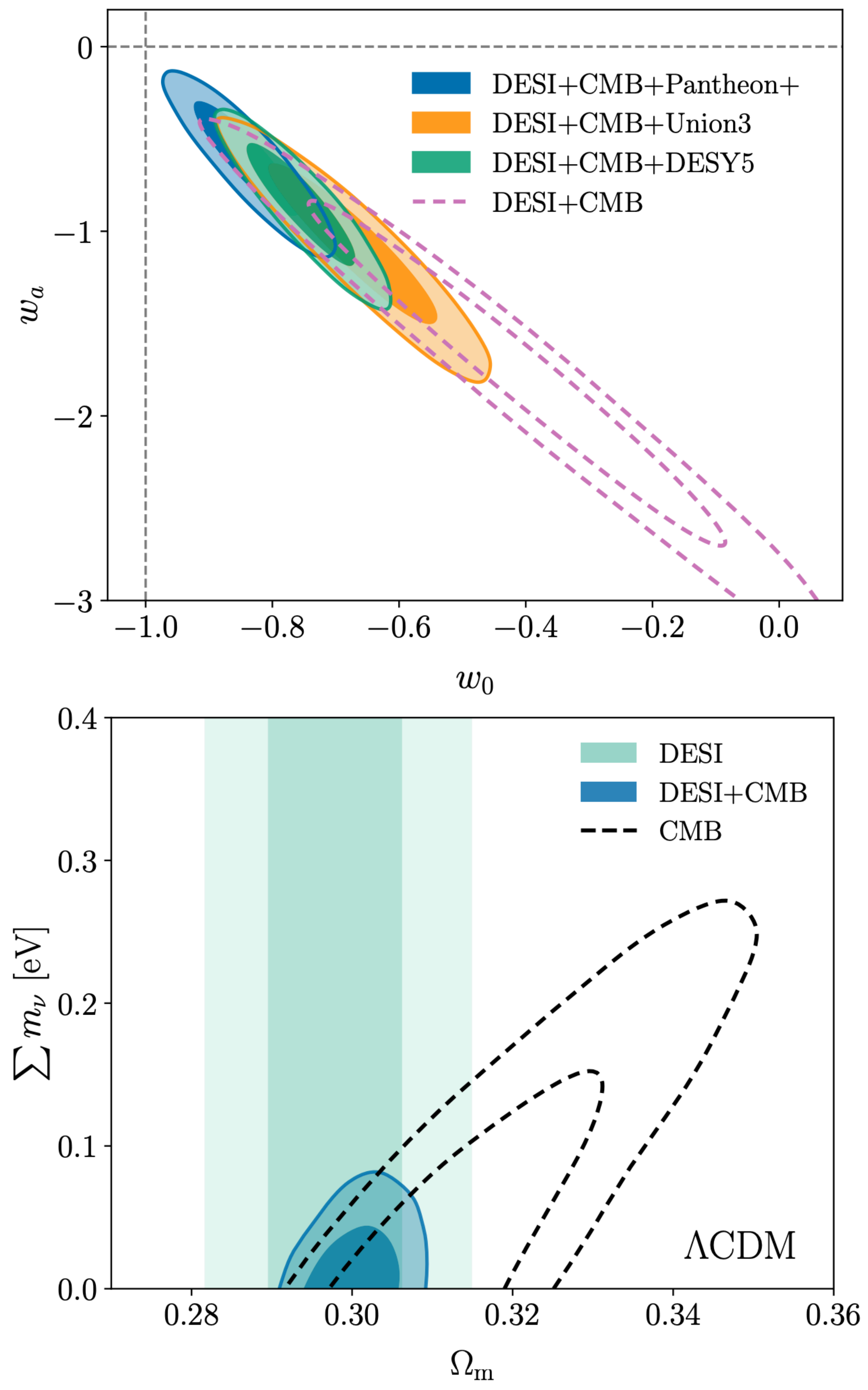
S_8 tension?



[KiDS-Legacy: 2503.19441]

Seems to be dead after KiDS-Legacy…

Latest DESI results



[DESI BAO DR2: 2503.14738]

To learn about the universe, we need to solve the **inverse** problem

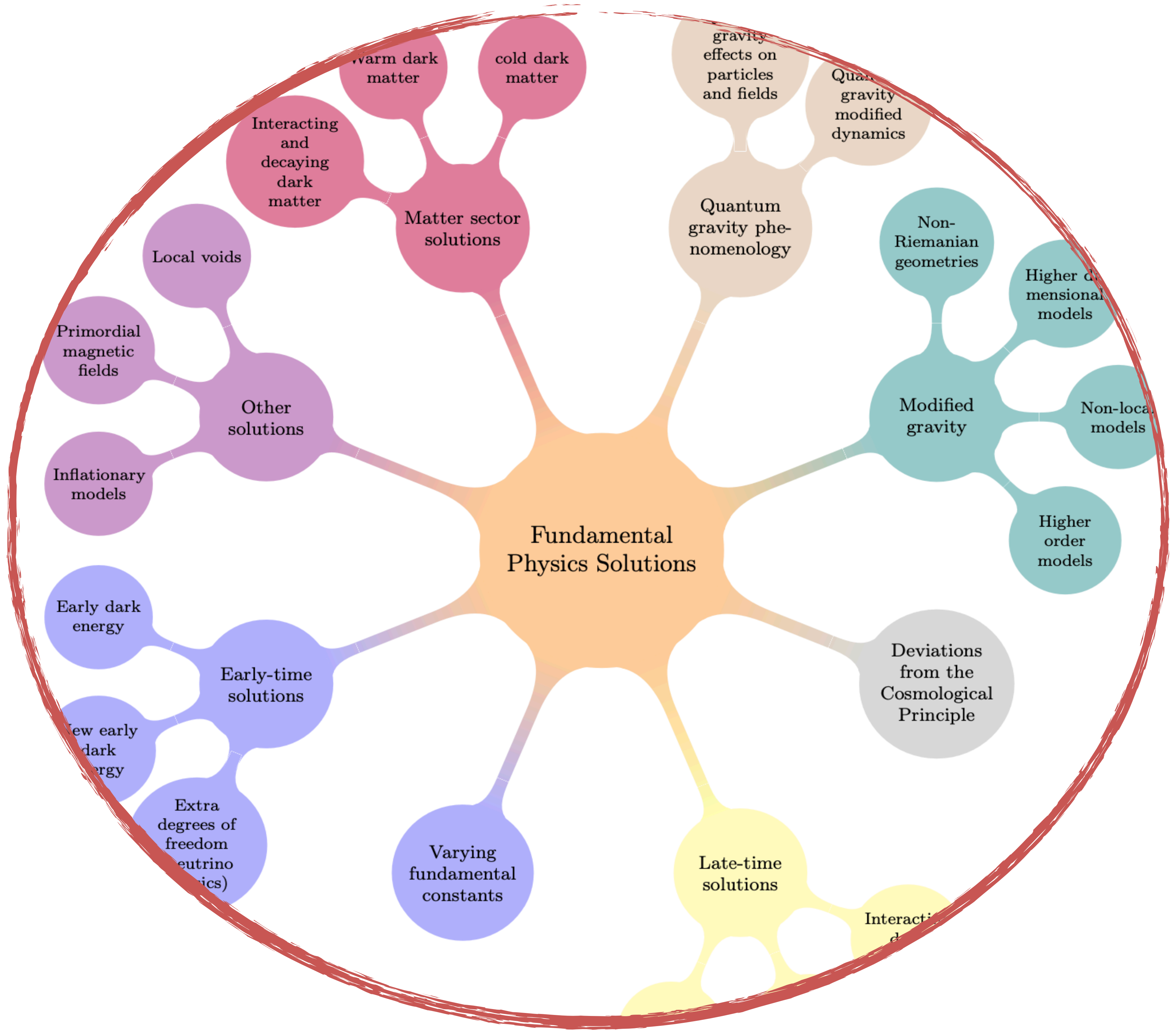
To learn about the universe, we need to solve the **inverse** problem

Data

To learn about the universe, we need to solve the **inverse** problem

Data

Theory



Some ΛCDM extensions I studied in the past

Decaying DM

[**GFA**+ 2008.09615]
[**GFA**+ 2102.12498]
[Simon, **GFA**+ 2203.07440]

Early Modified Gravity /Early Dark Energy

[**GFA**, Braglia+ 2308.12345]
[Murgia, **GFA**+ 2009.10733]

Interacting Stepped DR

[Schöneberg & **GFA** 2206.11276]
[Schöneberg, **GFA**+ 2306.12469]

Decaying neutrinos

[**GFA**+ 2112.13862]

Some ΛCDM extensions I studied in the past

Decaying DM

[**GFA**+ 2008.09615]
[**GFA**+ 2102.12498]
[Simon, **GFA**+ 2203.07440]

Early Modified Gravity /Early Dark Energy

[**GFA**, Braglia+ 2308.12345]
[Murgia, **GFA**+ 2009.10733]

Interacting Stepped DR

[Schöneberg & **GFA** 2206.11276]
[Schöneberg, **GFA**+ 2306.12469]

Decaying neutrinos

[**GFA**+ 2112.13862]

Hundreds of other models, **e.g. see reviews:**

"In the Realm of the Hubble tension - a Review of Solutions" Di Valentino+ [2103.01183]

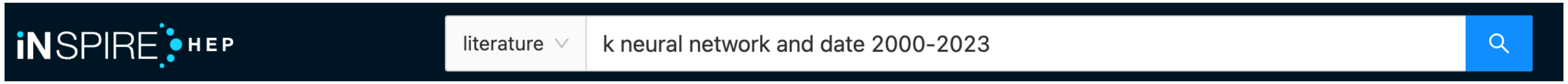
"The H_0 Olympics: a fair ranking of proposed models" Schöneberg, **GFA**+ [2107.10291]

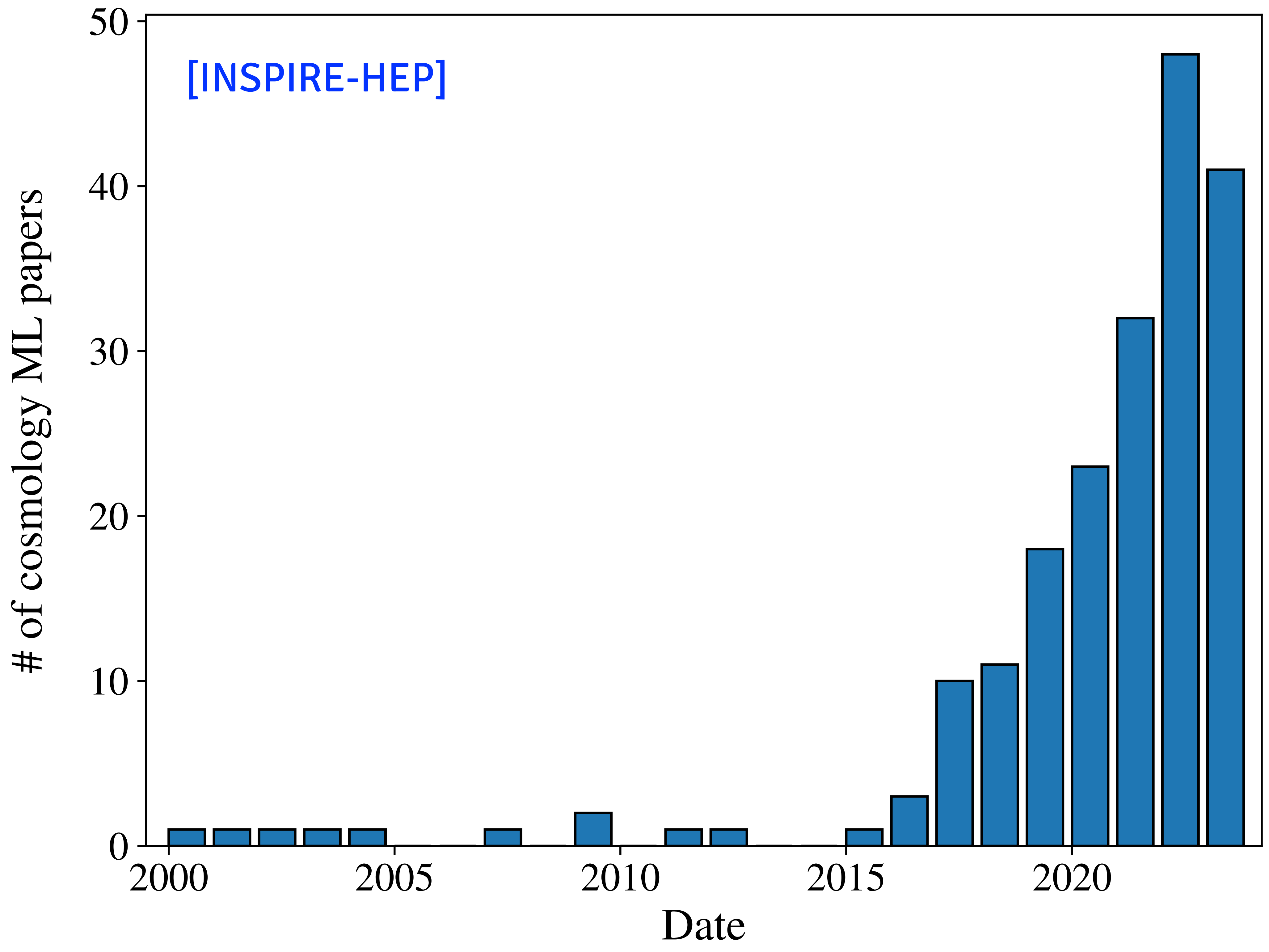
"The CosmoVerse White Paper", Di Valentino, Levi+ [2504.01669]

Comparing these vast volumes of cosmic data against the ever-increasing space of theories is a huge **challenge**

Comparing these vast volumes of cosmic data against the ever-increasing space of theories is a huge **challenge**

- Expensive simulations
- High-dimensional parameter spaces





Machine learning is having a **strong impact** in cosmology

Two main approaches in ML

Two main approaches in ML

Emulators

to **speed up** model evaluations of cosmological observables

Two main approaches in ML

Emulators

to **speed up** model evaluations of cosmological observables

New statistical methods

to improve the sampling in **high-dimensional** parameter spaces

1. Emulators

For testing **invisible neutrino decays** with latest cosmic data

2. Simulation-based inference

For efficient inference from **Euclid** data, with applications to **evolving dark energy**

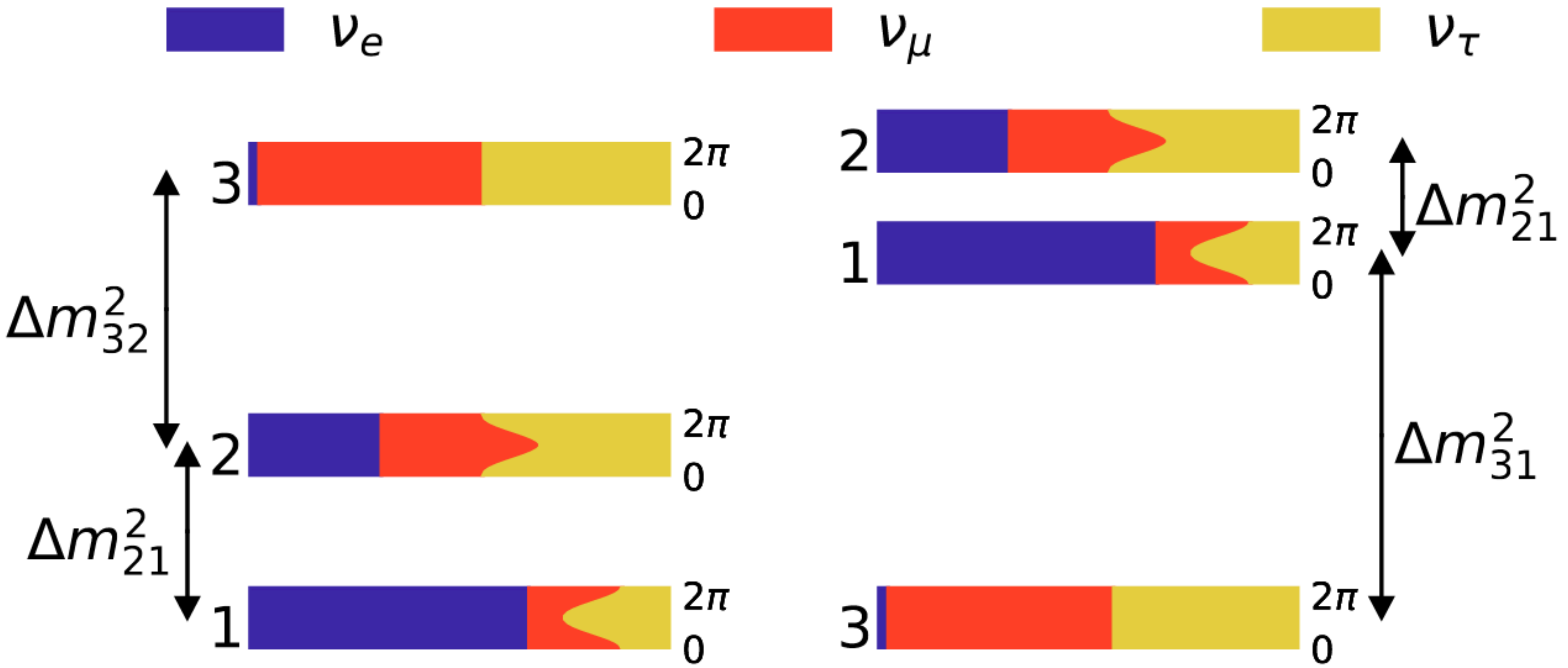
1. **Emulators**

For testing **invisible neutrino decays** with latest cosmic data

Ongoing work, to appear soon

arXiv:2506.XXXXX

Oscillation experiments have provided convincing evidence that **neutrinos have mass**



[Salas et al. 1806.11051]

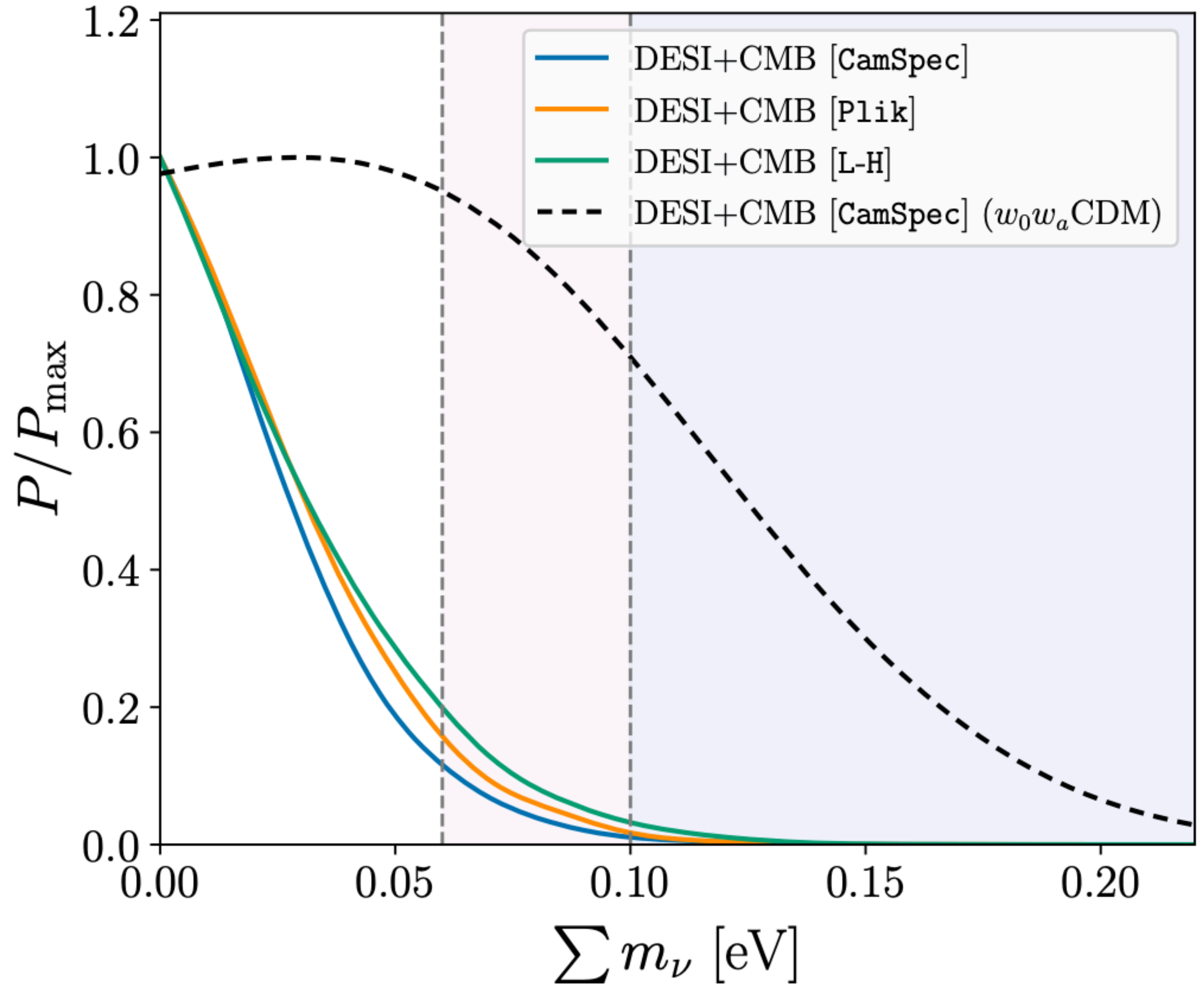
Normal

$$\sum m_\nu \gtrsim 0.06 \text{ eV}$$

Inverted

$$\sum m_\nu \gtrsim 0.10 \text{ eV}$$

Cosmological bounds



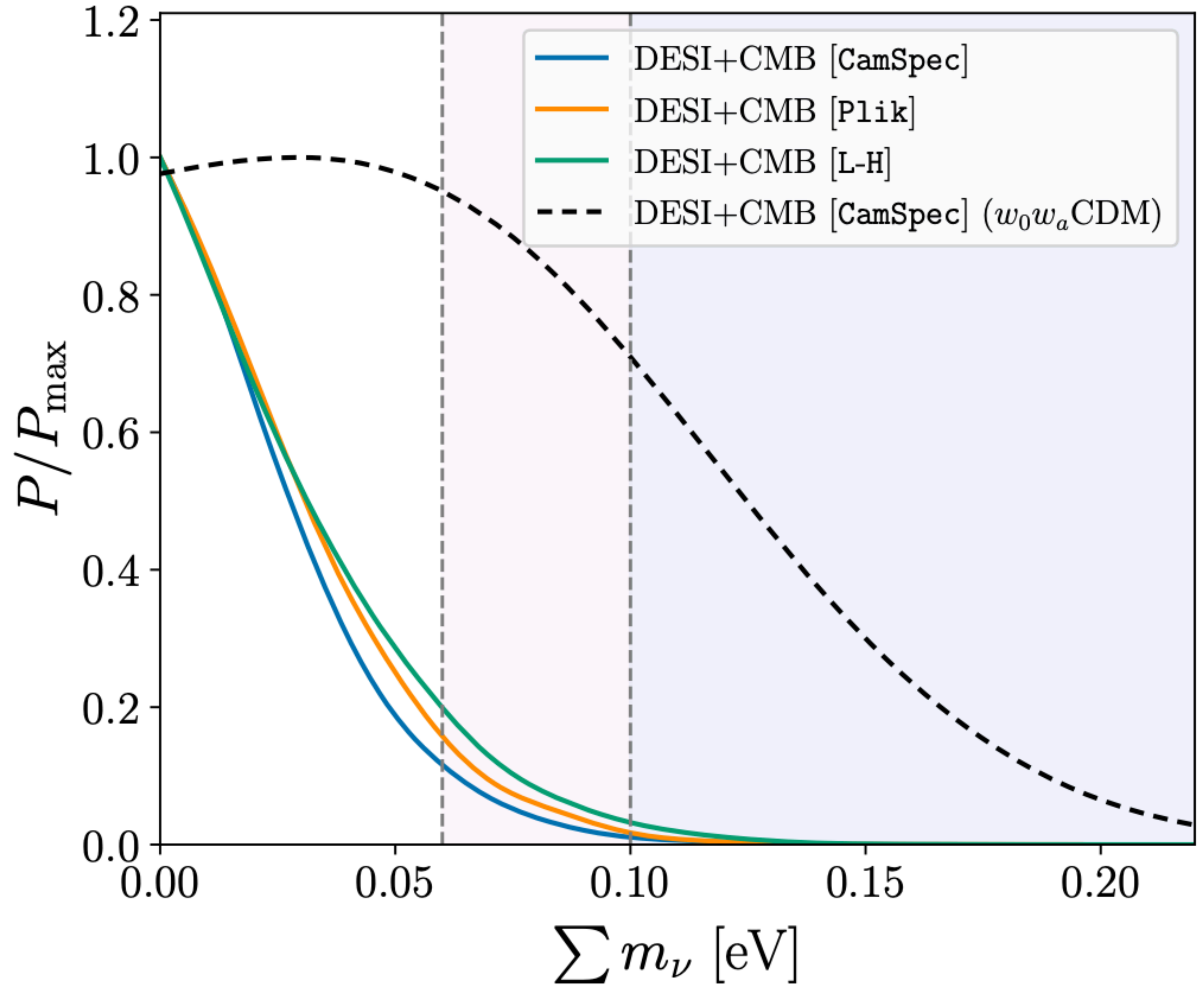
Cosmology has given tightest bound to date

$$\sum m_\nu < 0.064 \text{ eV}$$

(95%, DESI+CMB [CampSpec])

DESI DR2 Results II [2503.14738]

Cosmological bounds



Cosmology has given tightest bound to date

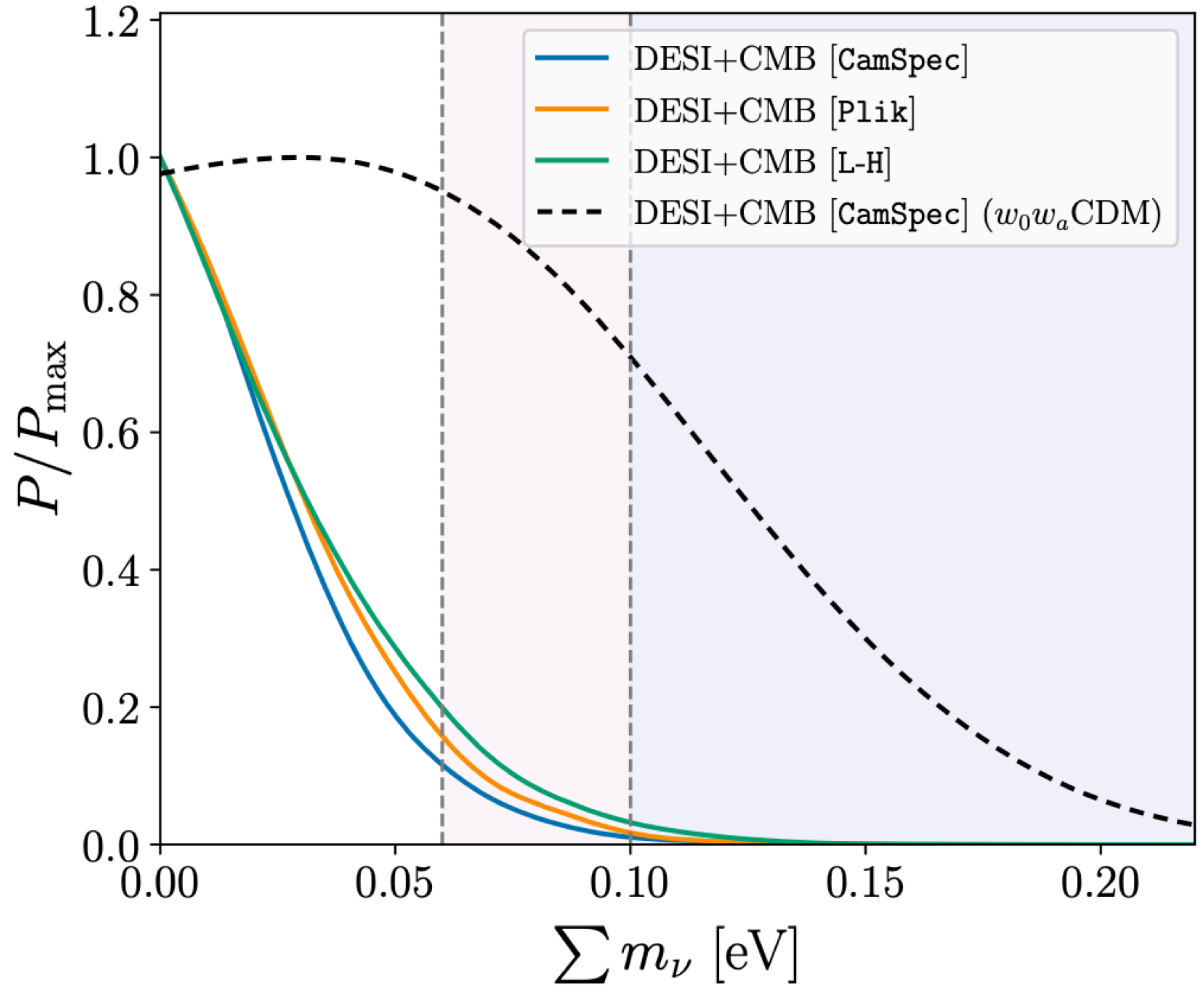
$$\sum m_\nu < 0.064\ \text{eV}$$ (95%, DESI+CMB [CampSpec])

However, these bounds are model dependent

$$\sum m_\nu < 0.163\ \text{eV}$$ (95%, w_0w_aCDM: DESI+CMB)

DESI DR2 Results II [2503.14738]

Cosmological bounds



DESI DR2 Results II [2503.14738]

Cosmology has given tightest bound to date

$$\sum m_\nu < 0.064 \text{ eV}$$

(95%, DESI+CMB [CampSpec])

However, these bounds are model dependent

$$\sum m_\nu < 0.163 \text{ eV}$$

(95%, $w_0 w_a$CDM: DESI+CMB)

What about changing neutrino properties?

Decaying neutrinos

- 2 neutrinos decay in the SM but $\tau_\nu \sim (G_F^2 m_\nu^5)^{-1} \gtrsim 10^{33}\ \mathrm{yr} \gg t_U$

Decaying neutrinos

- 2 neutrinos decay in the SM but $\tau_\nu \sim (G_F^2 m_\nu^5)^{-1} \gtrsim 10^{33}\ \mathrm{yr} \gg t_U$
- Radiative decays are strongly constrained $\tau_\nu > 10^2 - 10^{10}\ t_U$

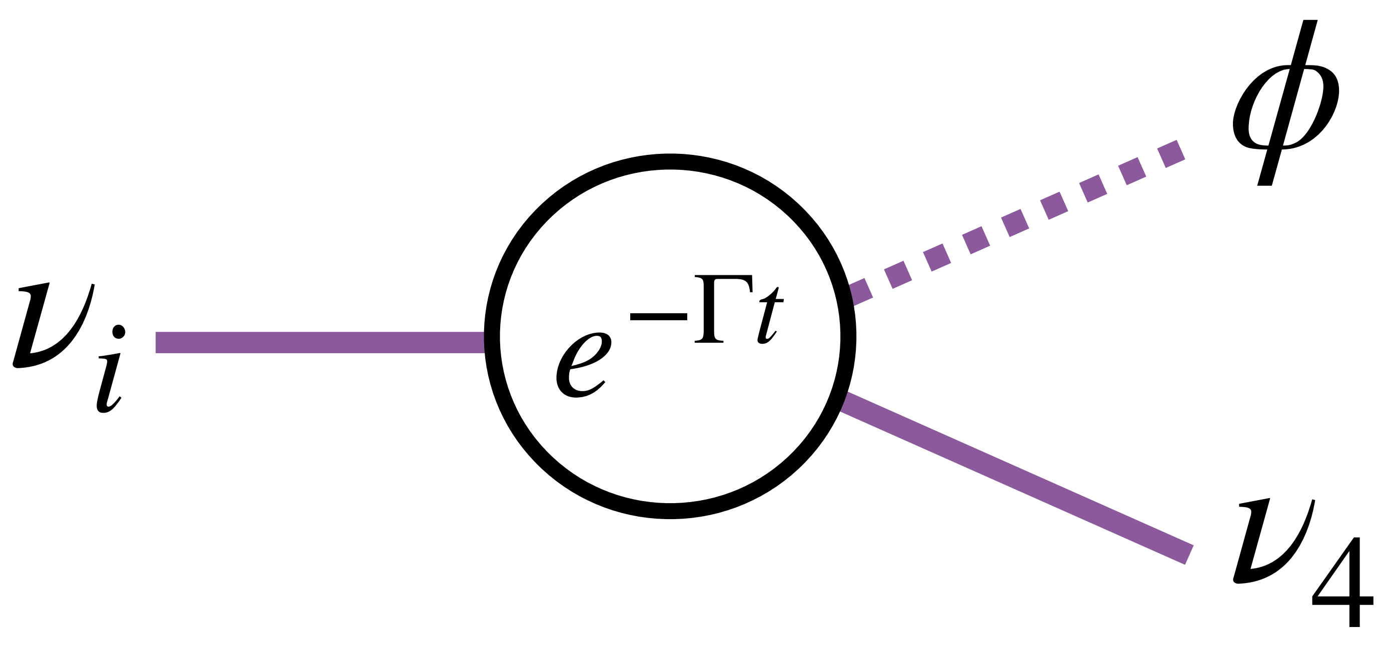
Aalberts, Ando et al. [1803.00588]

Decaying neutrinos

- 2 neutrinos decay in the SM but $\tau_\nu \sim (G_F^2 m_\nu^5)^{-1} \gtrsim 10^{33}\ \mathrm{yr} \gg t_U$
- Radiative decays are strongly constrained $\tau_\nu > 10^2 - 10^{10}\ t_U$

Aalberts, Ando et al. [1803.00588]

- Decays to dark radiation, much less constrained

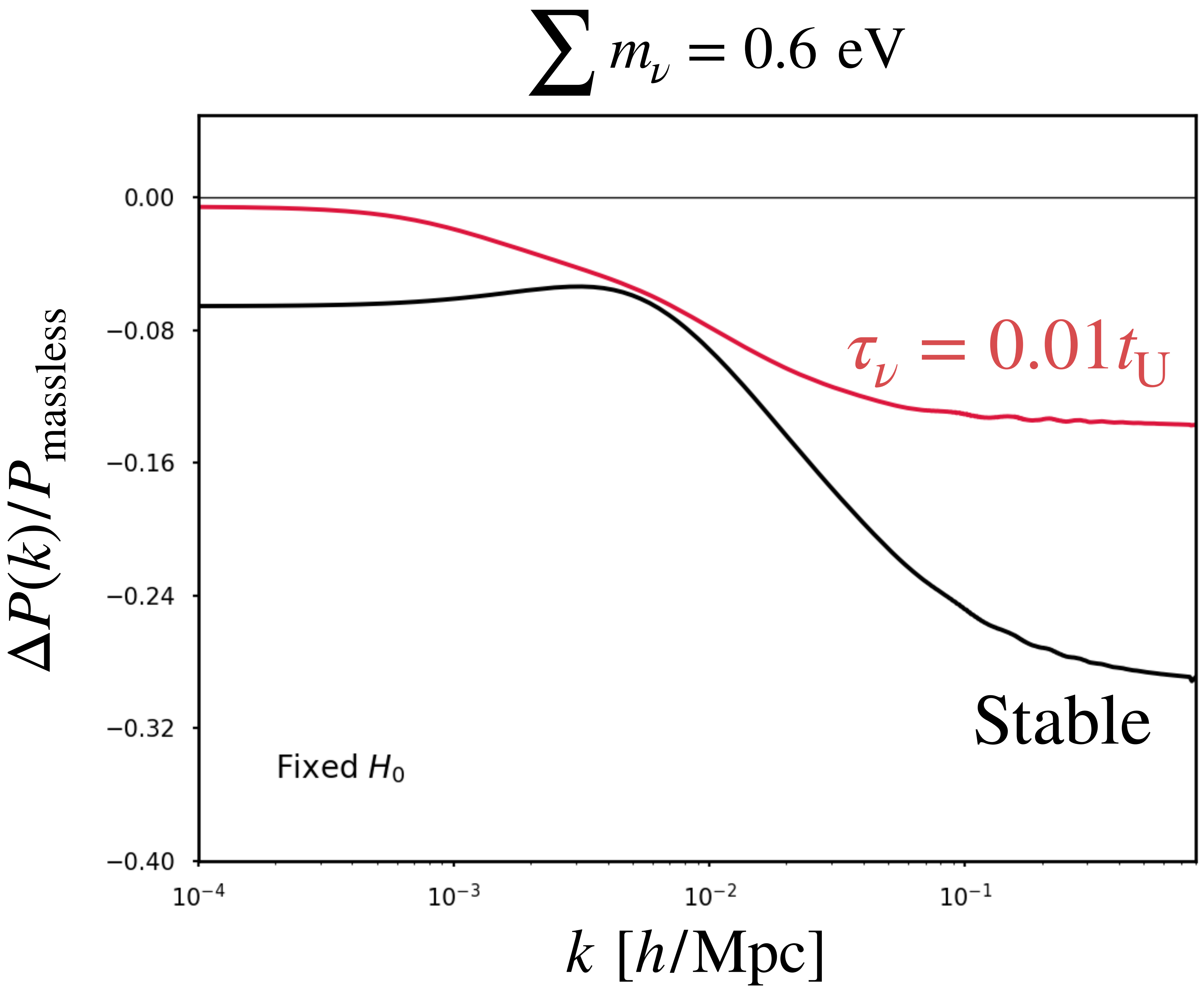


Appears naturally in many simple extensions of the seesaw mechanism

$$\mathscr{L}_{\rm int} = \lambda i \phi \bar{\nu}_i \gamma_5 \nu_4 + \mathrm{h.c.}$$

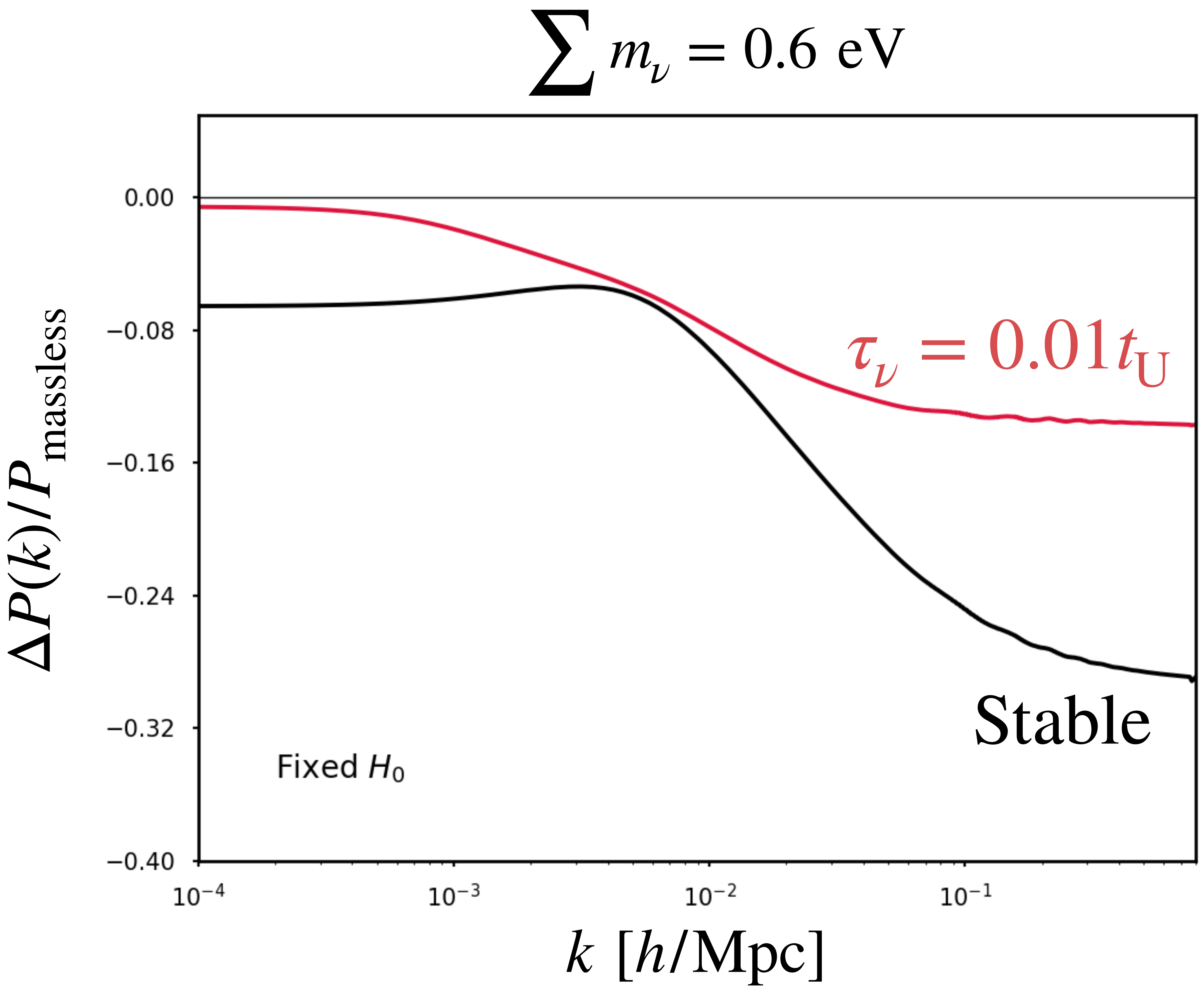
Escudero & Fairbairn [1907.05425]
Escudero et al. [2007.04994]

Decaying neutrinos



A finite neutrino lifetime allows to reduce their impact on structure growth and expansion rate

Decaying neutrinos

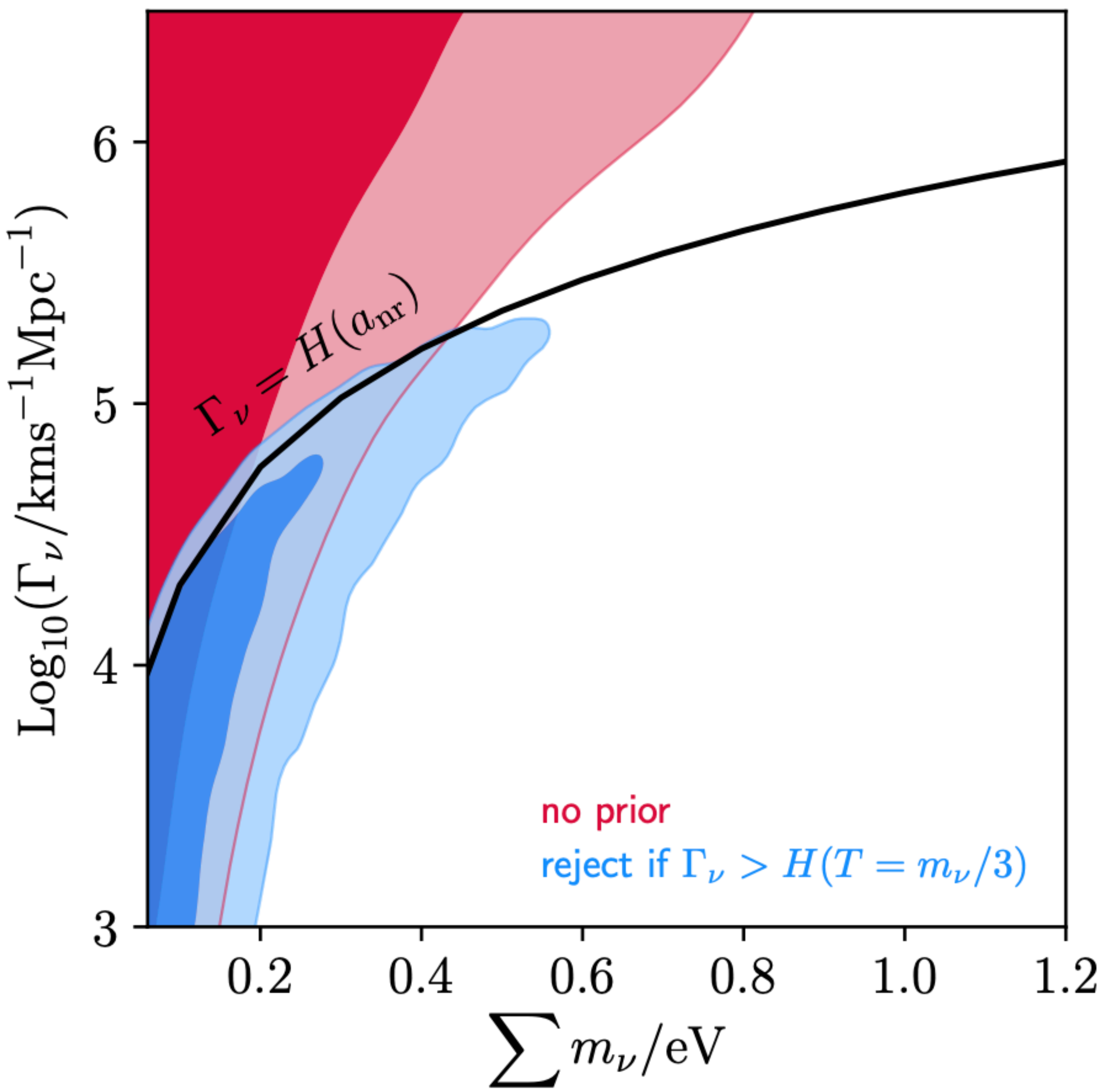


A finite neutrino lifetime allows to reduce their impact on structure growth and expansion rate

↓

This permits a relaxation of neutrino mass bounds from cosmology

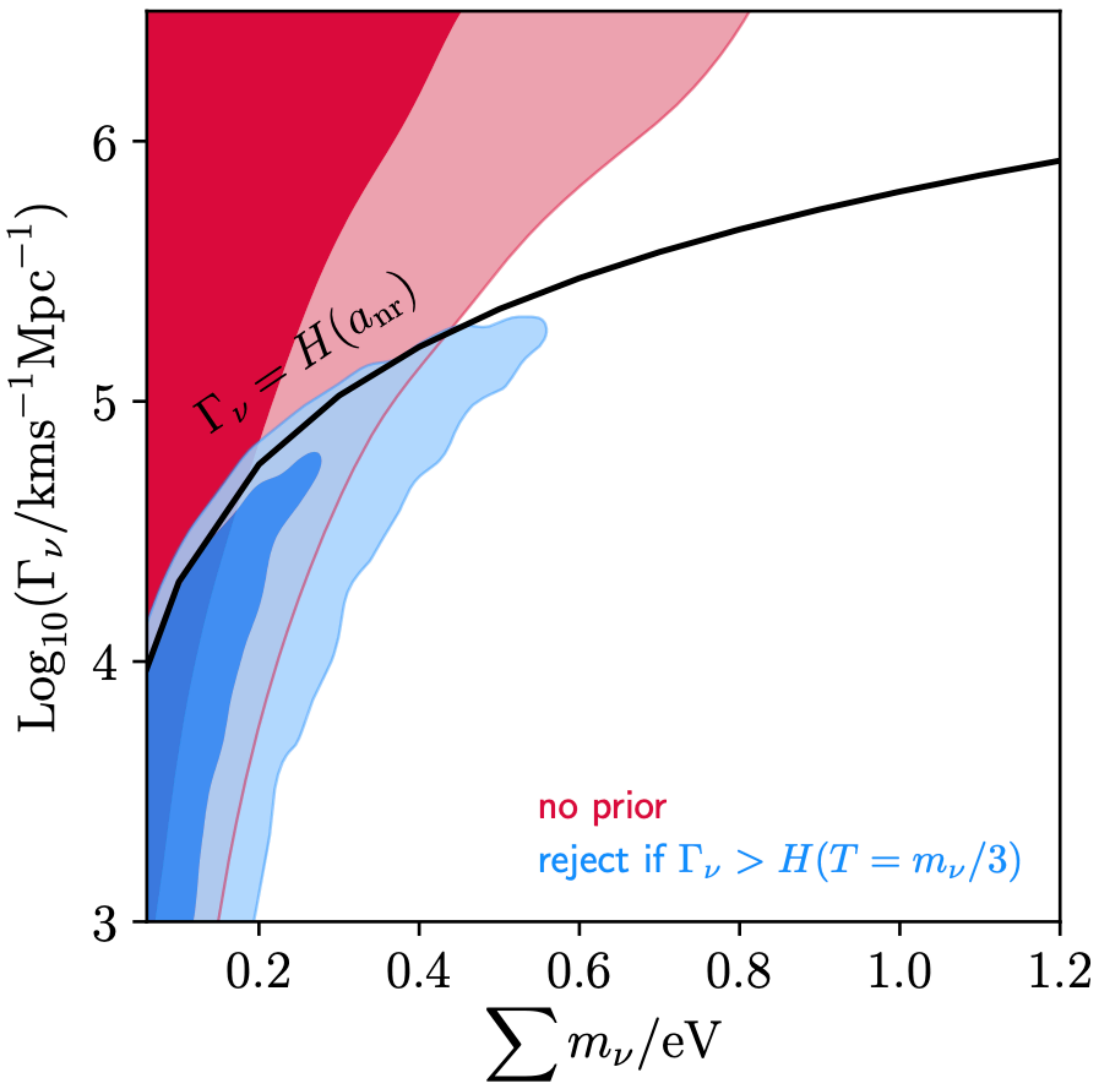
CMB+ BOSS + SNIa



Abellán et al. [2112.13862]

Back in 2021, we showed that non-relativistic neutrino decays to DR can relax mass bounds up to $\sum m_\nu < 0.4$ eV

CMB+ BOSS + SNIa



Abellán et al. [2112.13862]

Back in 2021, we showed that non-relativistic neutrino decays to DR can relax mass bounds up to $\sum m_\nu < 0.4$ eV

GOAL

Build emulator for neutrino decay model, and use it to update mass bounds in light of latest DESI data

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0,\ \Omega_b, A_s,\ n_s, \ldots$

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0,\ \Omega_b, A_s,\ n_s, \ldots$ →

Background/thermo eqs.

$$H^2 = \frac{8\pi G}{3}\sum_i \rho_i$$

$$\frac{d\rho_i}{d\tau} = \ldots, \quad \text{for i} = \text{c, b}, \gamma, \nu$$

$$\frac{dx_e}{d\tau} = \ldots, \qquad \frac{dT_b}{d\tau} = \ldots$$

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0, \Omega_b, A_s, n_s, \ldots$ →

Background/thermo eqs.

$$H^2 = \frac{8\pi G}{3} \sum_i \rho_i$$

$$\frac{d\rho_i}{d\tau} = \ldots, \quad \text{for i} = \text{c}, \text{b}, \gamma, \nu$$

$$\frac{dx_e}{d\tau} = \ldots, \qquad \frac{dT_b}{d\tau} = \ldots$$

→

Perturbation eqs.

$$\partial_\tau \Delta f_{0,i}\,(q, k, \tau) = \ldots$$

$$\partial_\tau \Delta f_{1,i}\,(q, k, \tau) = \ldots$$

$$\vdots$$

$$\partial_\tau \Delta f_{\ell_{\max},i}\,(q, k, \tau) = \ldots$$

$$\text{for i} = \text{c}, \text{b}, \gamma, \nu \quad \text{and} \quad \forall\ k, q$$

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0,\ \Omega_b, A_s,\ n_s, \ldots$ →

Background/thermo eqs.

$$H^2 = \frac{8\pi G}{3}\sum_i \rho_i$$

$$\frac{d\rho_i}{d\tau} = \ldots, \qquad \text{for i} = \text{c}, \text{b}, \gamma, \nu$$

$$\frac{dx_e}{d\tau} = \ldots, \qquad \frac{dT_b}{d\tau} = \ldots$$

→

Perturbation eqs.

$$\partial_\tau \Delta f_{0,i}\ (q, k, \tau) = \ldots$$

$$\partial_\tau \Delta f_{1,i}\ (q, k, \tau) = \ldots$$

$$\vdots$$

$$\partial_\tau \Delta f_{\ell_{\max},i}\ (q, k, \tau) = \ldots$$

$$\text{for i} = \text{c}, \text{b}, \gamma, \nu \quad \text{and} \quad \forall\ k, q$$

→

Observables

$D_A(z),\ C_\ell^{\mathrm{XY}},\ P_m(k), \ldots$

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0,\ \Omega_b, A_s,\ n_s, \ldots$ →

Background/thermo eqs.

$$H^2 = \frac{8\pi G}{3}\sum_i \rho_i$$

$$\frac{d\rho_i}{d\tau} = \ldots, \quad \text{for i} = \text{c, b}, \gamma, \nu$$

$$\frac{dx_e}{d\tau} = \ldots, \qquad \frac{dT_b}{d\tau} = \ldots$$

→

Perturbation eqs.

$$\partial_\tau \Delta f_{0,i}\ (q, k, \tau) = \ldots$$

$$\partial_\tau \Delta f_{1,i}\ (q, k, \tau) = \ldots$$

$$\vdots$$

$$\partial_\tau \Delta f_{\ell_{\max},i}\ (q, k, \tau) = \ldots$$

$$\text{for i} = \text{c, b}, \gamma, \nu \quad \text{and} \quad \forall\ k, q$$

→

Observables

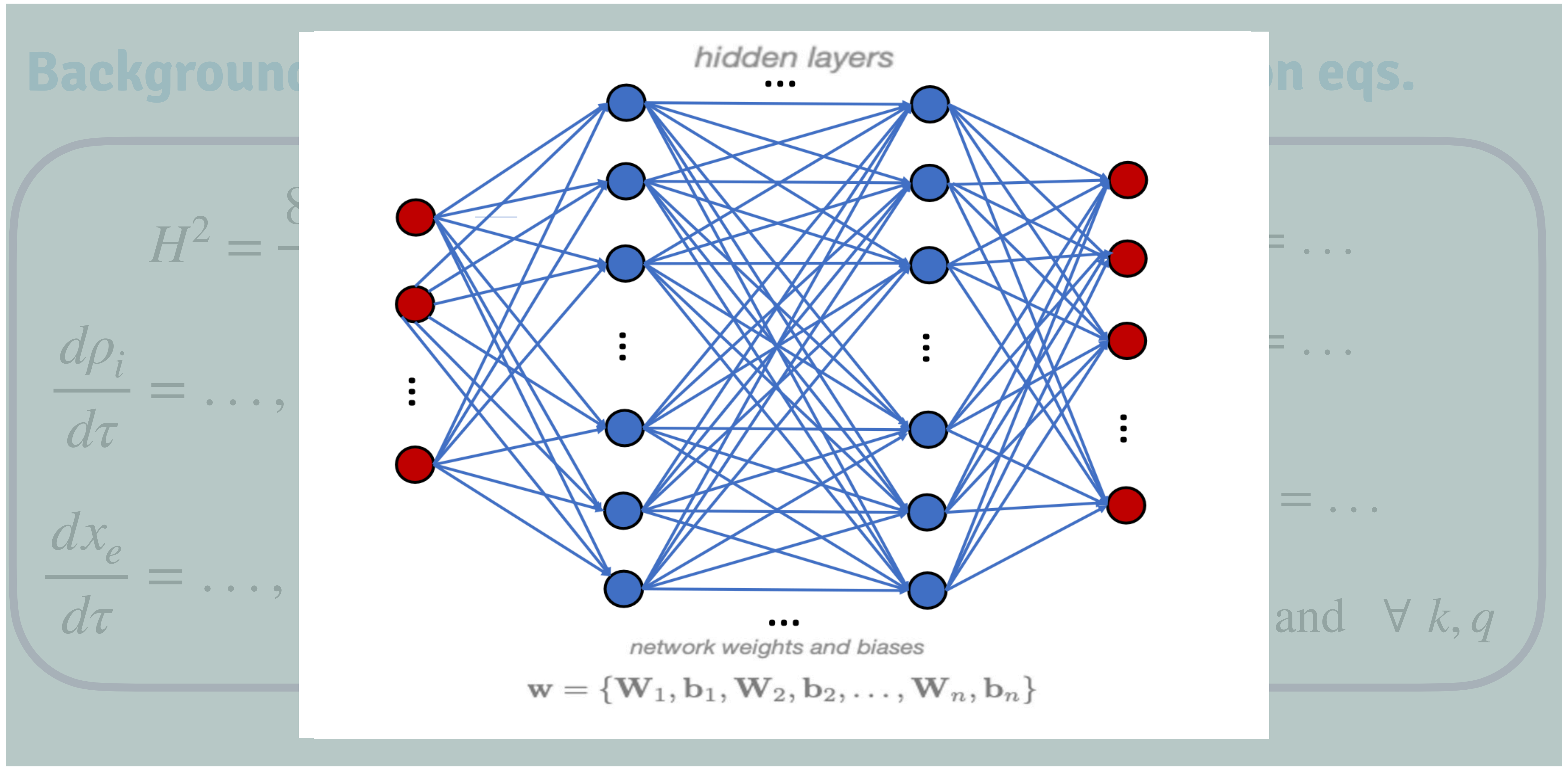
$D_A(z),\ C_\ell^{\mathrm{XY}},\ P_m(k), \ldots$

A single call to an EBS (e.g. CLASS) can be time-consuming in ΛCDM extensions.
Moreover, one often needs ~10^6 model evaluations for accurate inference.

Emulating an Einstein-Boltzmann Solver (EBS)

Cosmo. pars

$H_0,\ \Omega_b, A_s,\ n_s, \ldots$ →



→ **Observables**

$D_A(z),\ C_\ell^{\rm XY},\ P_m(k), \ldots$

IDEA: Replace the calculation of the full system of linear Einstein-Boltzmann eqs. by a trained emulator, e.g. a neural network (NN)

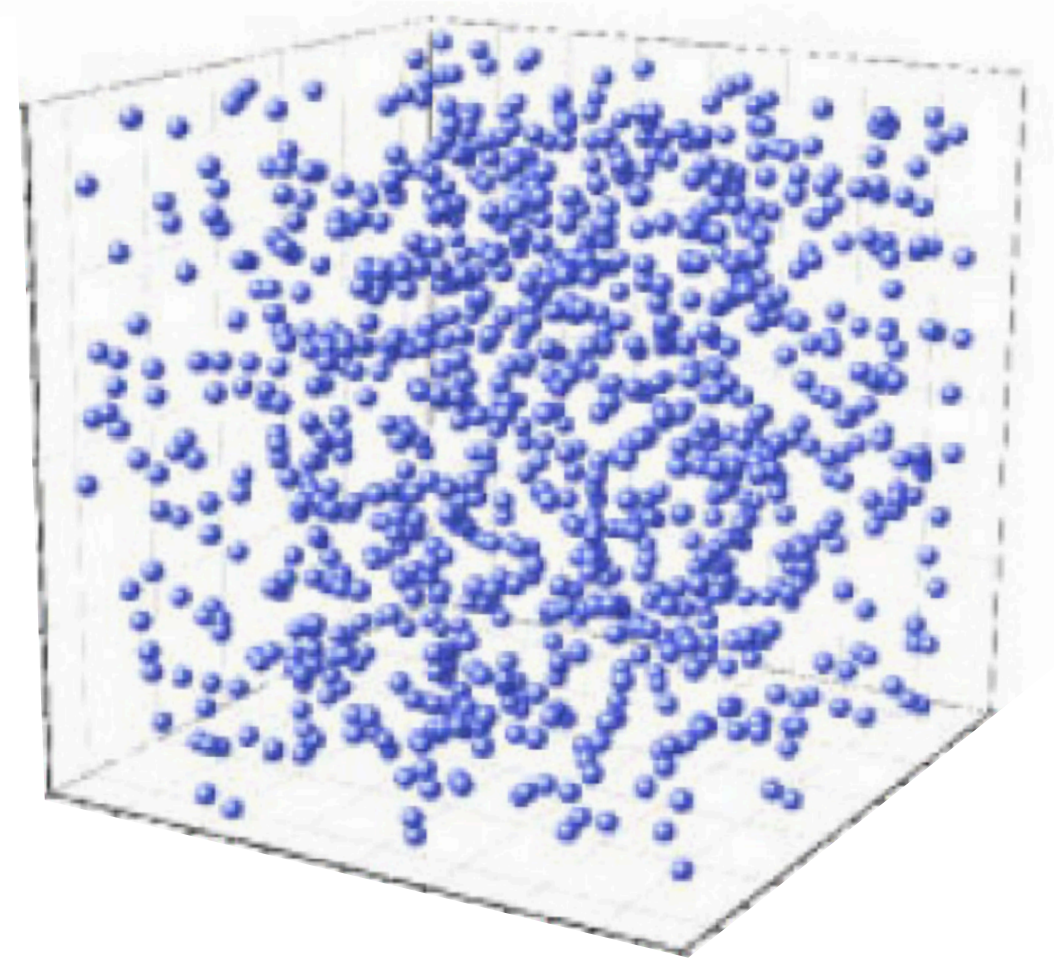
Emulating an Einstein-Boltzmann Solver (EBS)

Many EBS emulators available on the market (e.g. CosmoPower, CosmicNet, Capse.jl)
We rely on CONNECT [Nygaard et al. 2205.15726]

Emulating an Einstein-Boltzmann Solver (EBS)

Many EBS emulators available on the market (e.g. CosmoPower, CosmicNet, Capse.jl)
We rely on CONNECT [Nygaard et al. 2205.15726]

Latin Hypercube (LHC) sampling

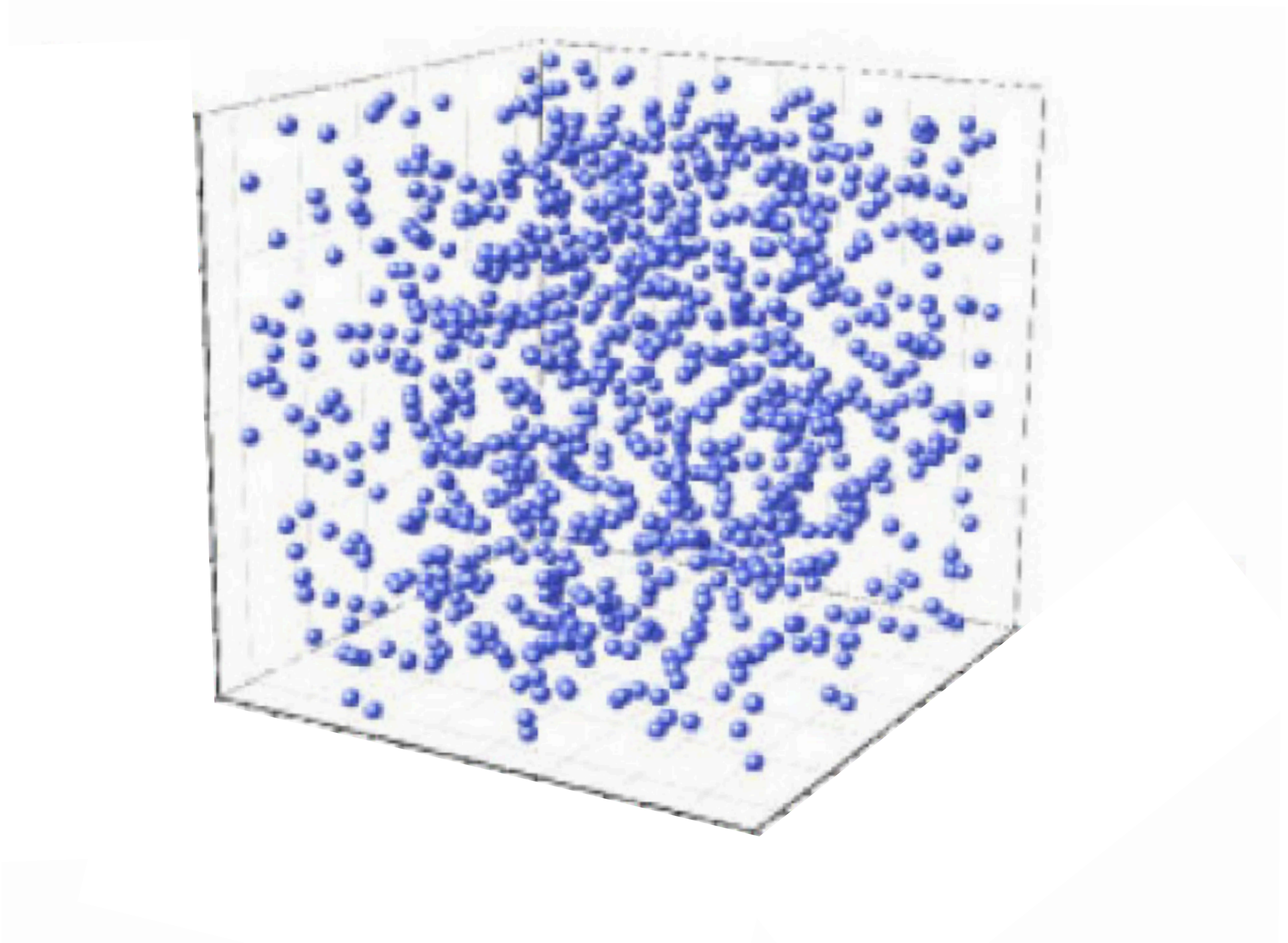


Achieves close-to-uniform sampling, but models near the corners have very low likelihood

Emulating an Einstein-Boltzmann Solver (EBS)

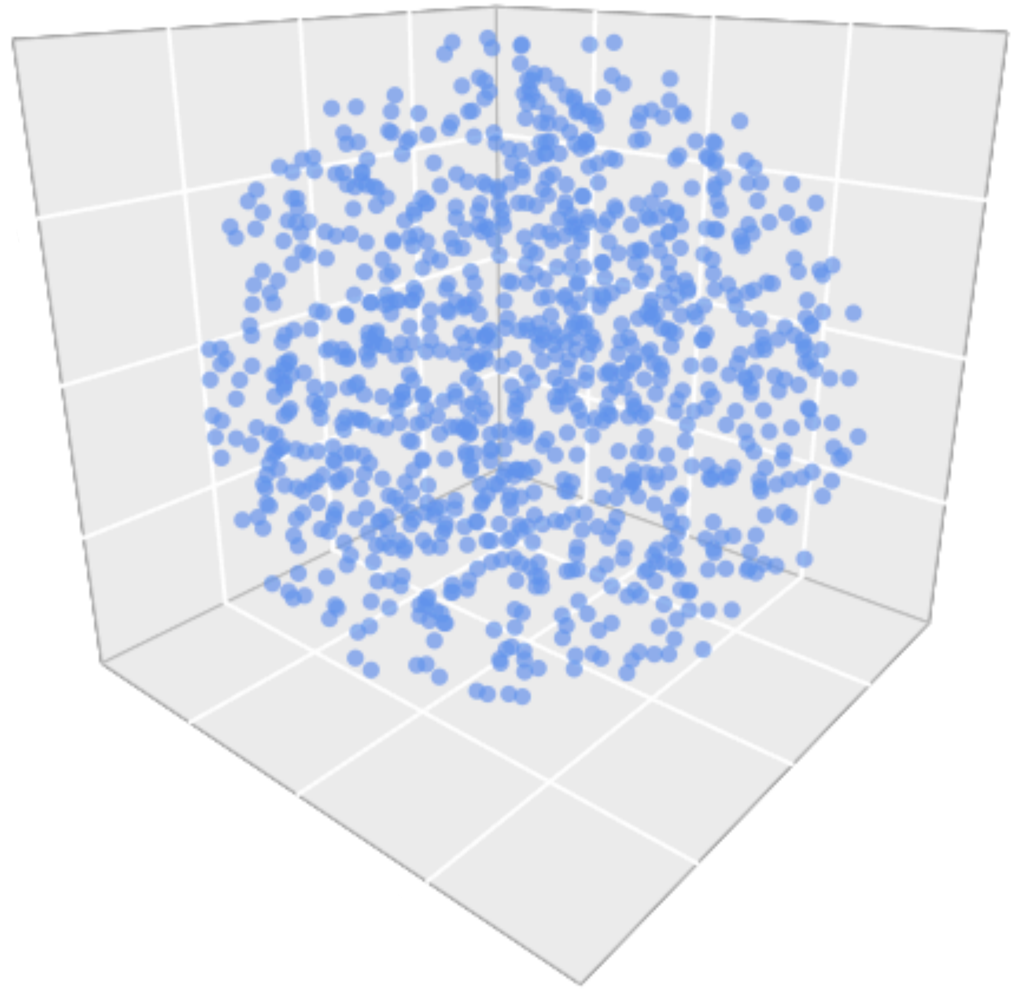
Many EBS emulators available on the market (e.g. CosmoPower, CosmicNet, Capse.jl)
We rely on CONNECT [Nygaard et al. 2205.15726]

Latin Hypercube (LHC) sampling



Achieves close-to-uniform sampling, but models near the corners have very low likelihood

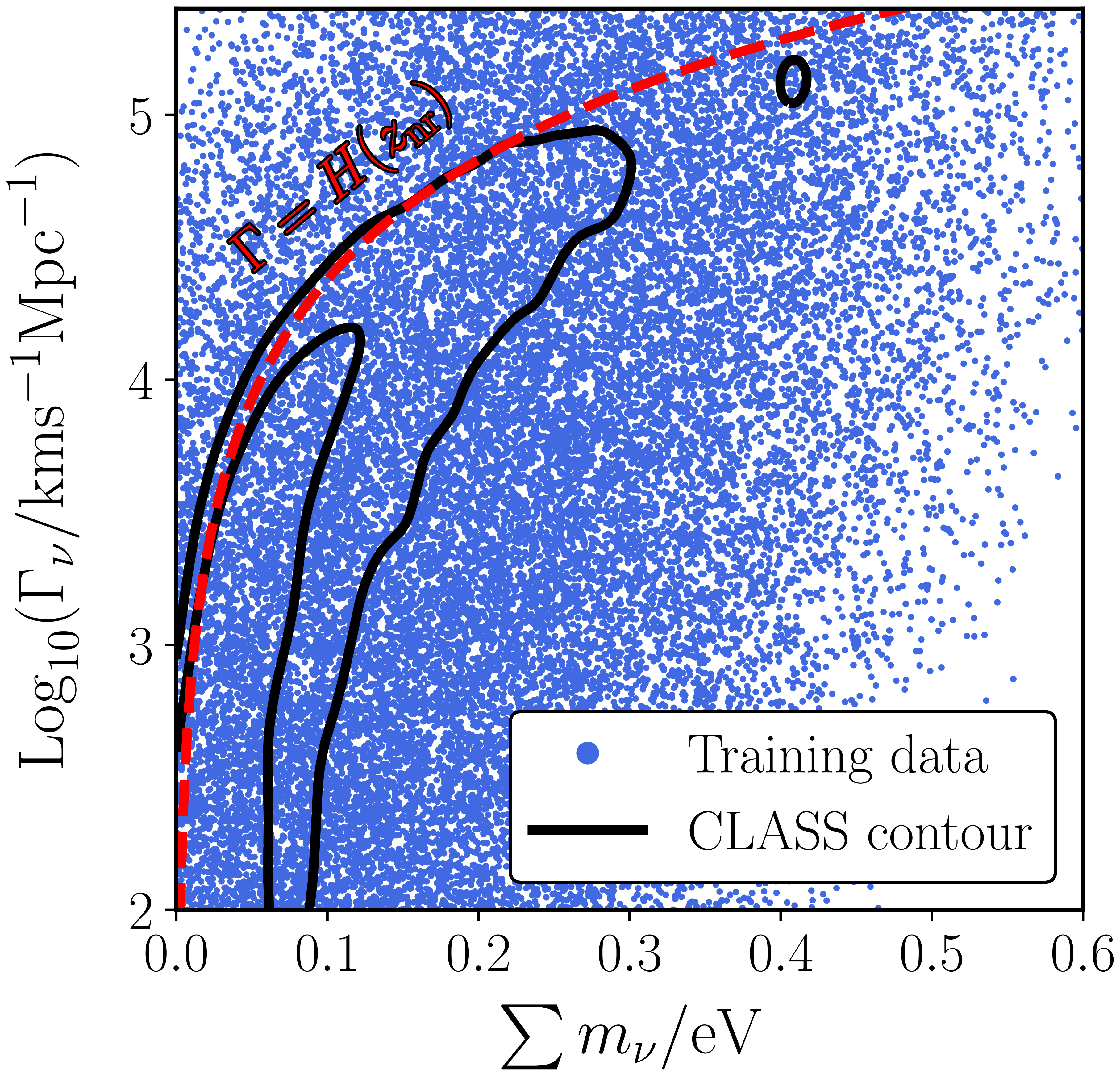
Hypersphere sampling



Allows to concentrate points in high-likelihood regions

→ more efficient and accurate

Building emulator for decaying neutrinos

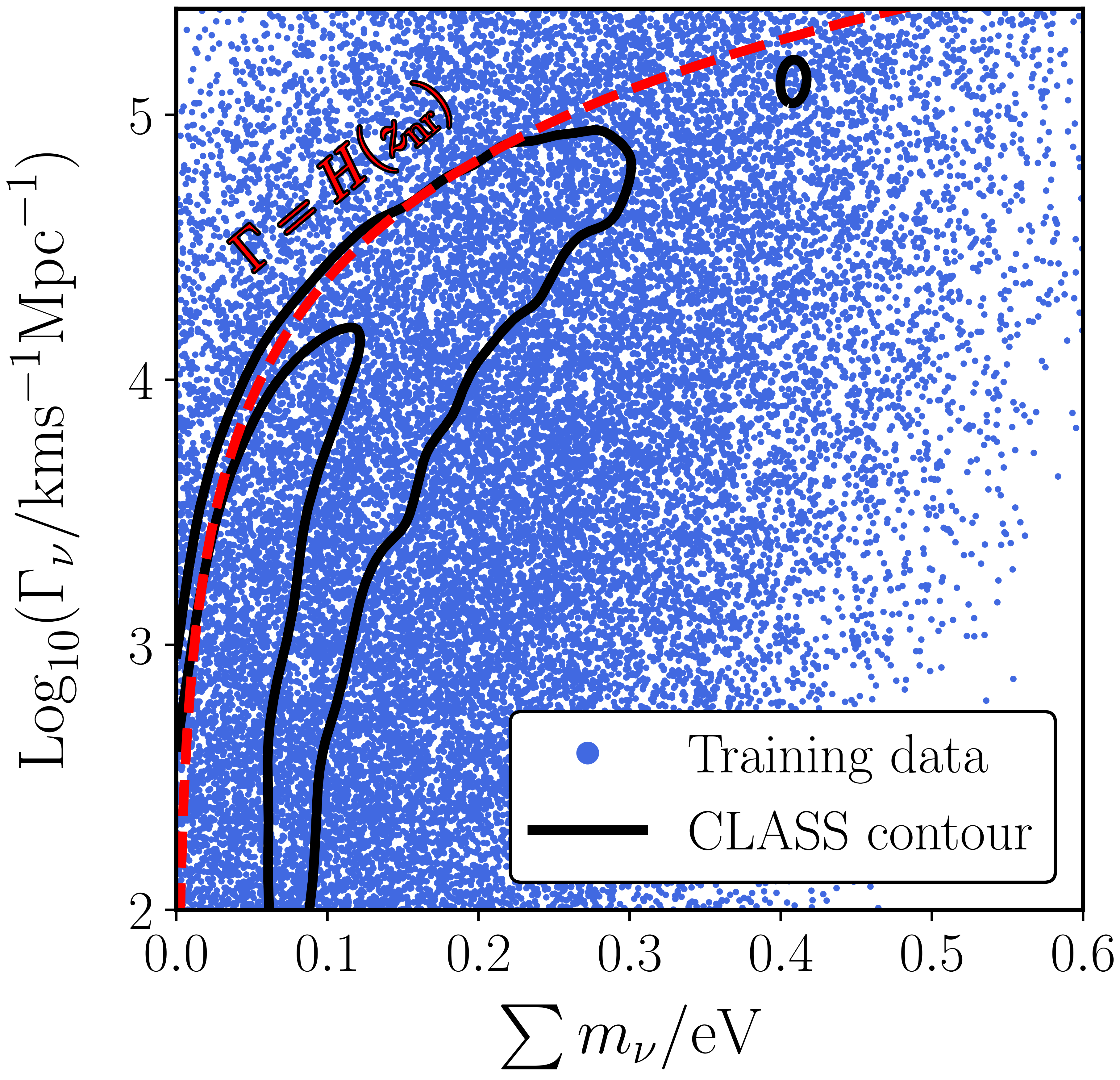


Training data: 2.5×10^4 samples of $\{\boldsymbol{\theta}, \boldsymbol{x}\}$ in the 8-dimensional hypersphere

$$\boldsymbol{\theta} = \{\omega_b,\ \omega_c,\ H_0,\ \ln(10^{10}A_s),\ n_s,\ \tau_{\rm reio},\ m_\nu,\ \log_{10}\Gamma_\nu\}$$

$$\boldsymbol{x} = \{C_\ell^{\rm TT},\ C_\ell^{\rm EE},\ C_\ell^{\rm TE},\ C_\ell^{\phi\phi},\ H(z),\ D_A(z)\}$$

Building emulator for decaying neutrinos



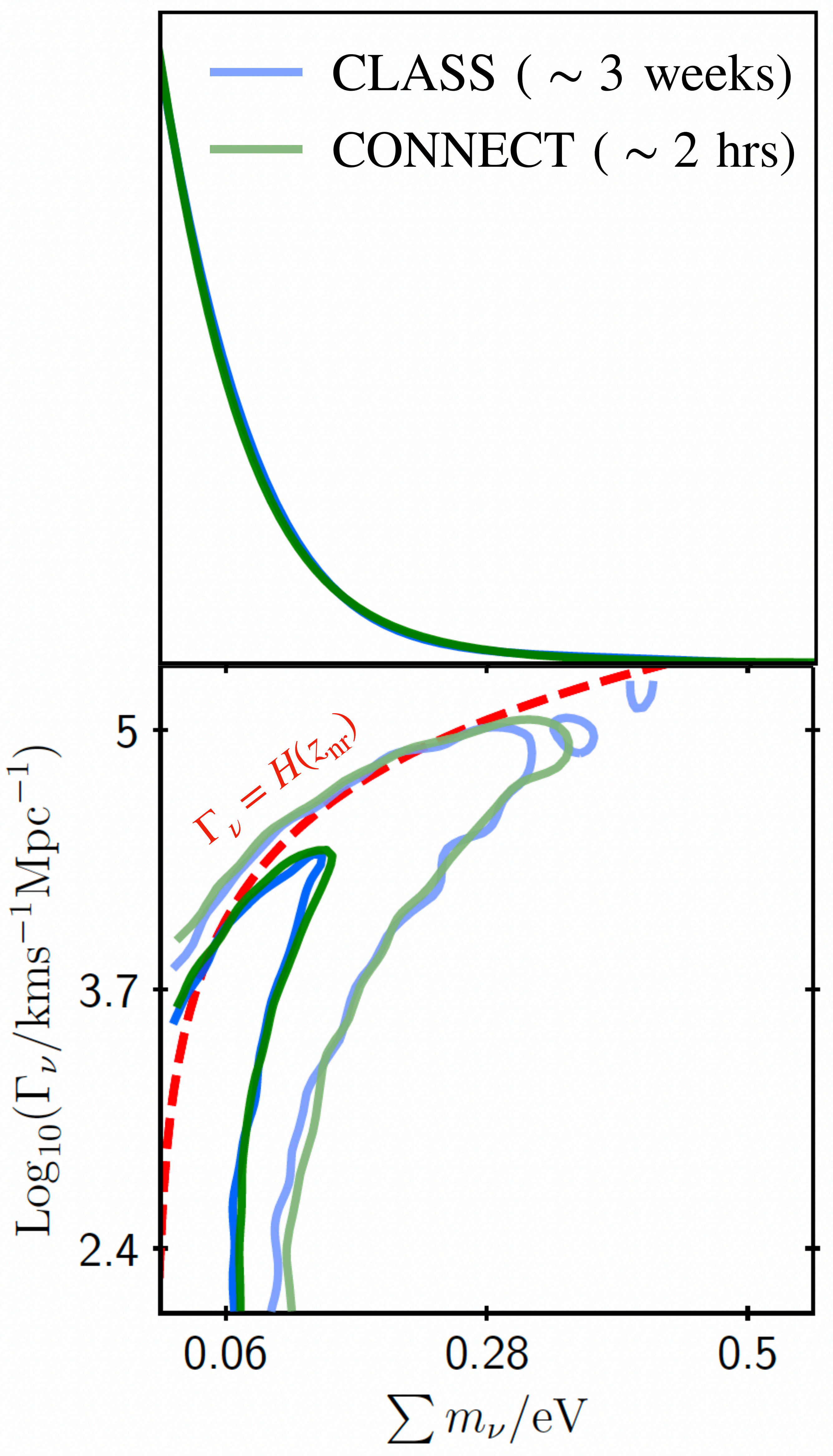
Training data: 2.5×10^4 samples of $\{\boldsymbol{\theta}, \boldsymbol{x}\}$ in the 8-dimensional hypersphere

$$\boldsymbol{\theta} = \{\omega_b,\ \omega_c,\ H_0,\ \ln(10^{10}A_s),\ n_s,\ \tau_{\rm reio},\ m_\nu,\ \log_{10}\Gamma_\nu\}$$

$$\boldsymbol{x} = \{C_\ell^{\rm TT},\ C_\ell^{\rm EE},\ C_\ell^{\rm TE},\ C_\ell^{\phi\phi},\ H(z),\ D_A(z)\}$$

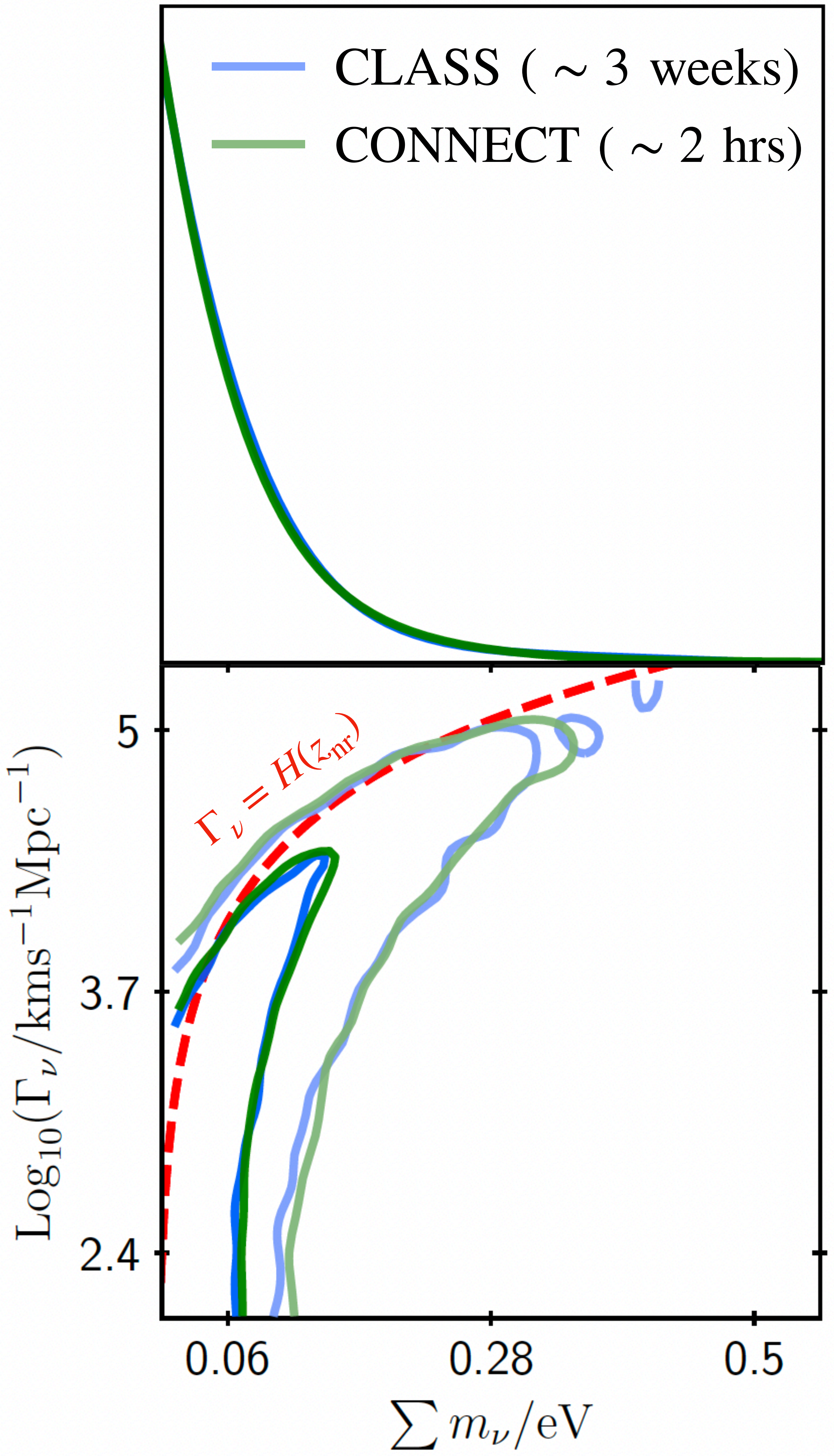
Data generation took ~ 1day on 128 CPUs (training the NNs was much faster)

Cosmological constraints



CONNECT posteriors are in good agreement with CLASS, while being $\times 250$ faster

Cosmological constraints



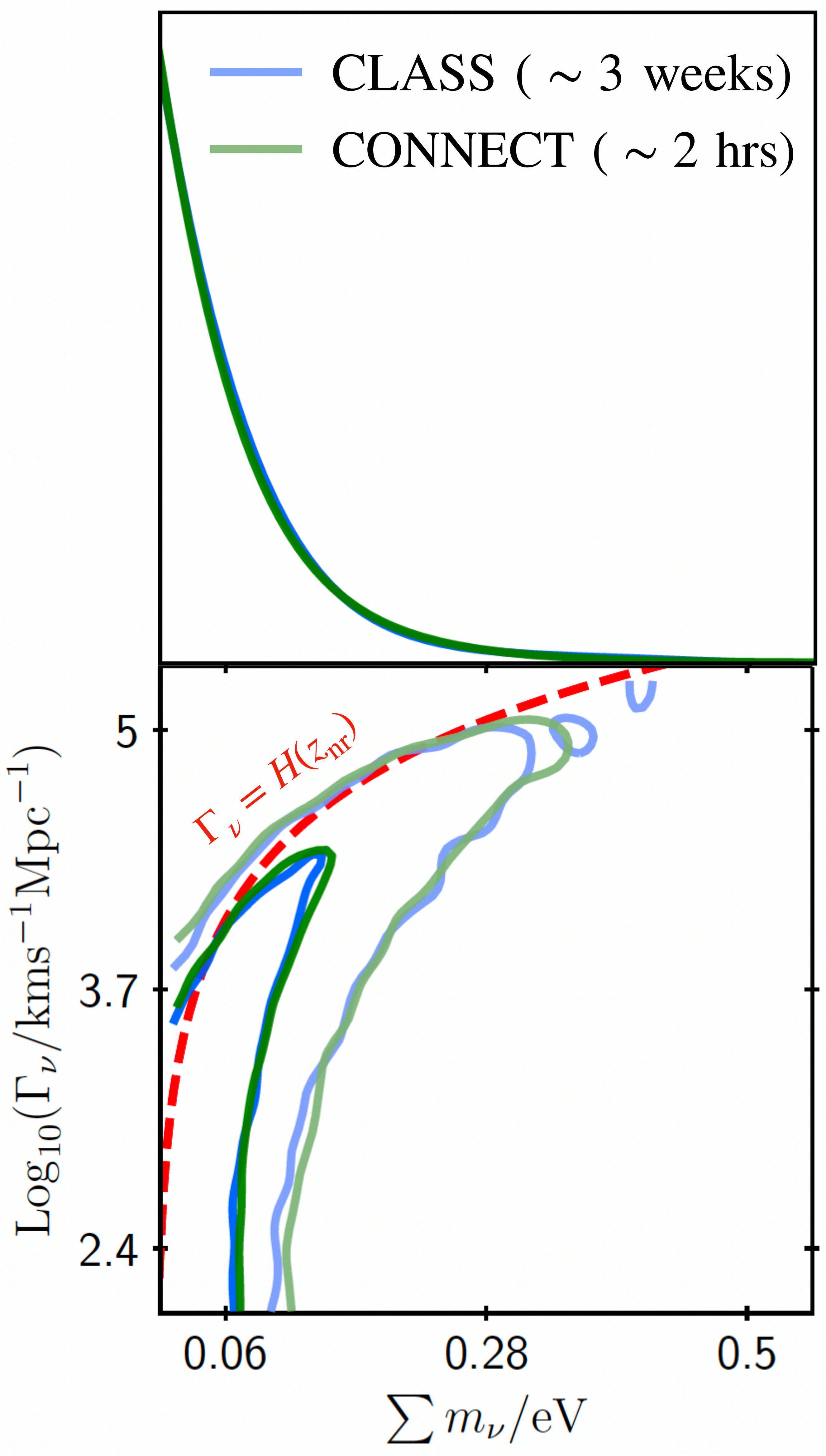
CONNECT posteriors are in good agreement with CLASS, while being × 250 faster

New mass bound:

$$\sum m_\nu < 0.24 \text{ eV}$$

(95%, DESI DR2+CMB [Plik])

Cosmological constraints



CONNECT posteriors are in good agreement with CLASS, while being $\times$ 250 faster

New mass bound:

$$\sum m_\nu < 0.24\ \mathrm{eV}$$

(95%, DESI DR2+CMB [Plik])

Next step: derive bounds for neutrino decays with realistic mass splittings $\nu_i \rightarrow \nu_j + \phi$

Based on arXiv:2403.14750
arXiv:2506.XXXX

with Guadalupe Cañas-Herrera, Matteo Martinelli, Oleg Savchenko, Noemi Anau Montel & Christoph Weniger

2. Simulation-based inference

For efficient inference from **Euclid** data, with applications to **evolving dark energy**

Bayesian inference

$\mathbf{x}_0$: Data

$\boldsymbol{\theta}$: Parameters

Posterior — Likelihood — Prior

$$p(\boldsymbol{\theta} \mid \mathbf{x}_0) \propto p(\mathbf{x}_0 \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$$

Bayesian inference

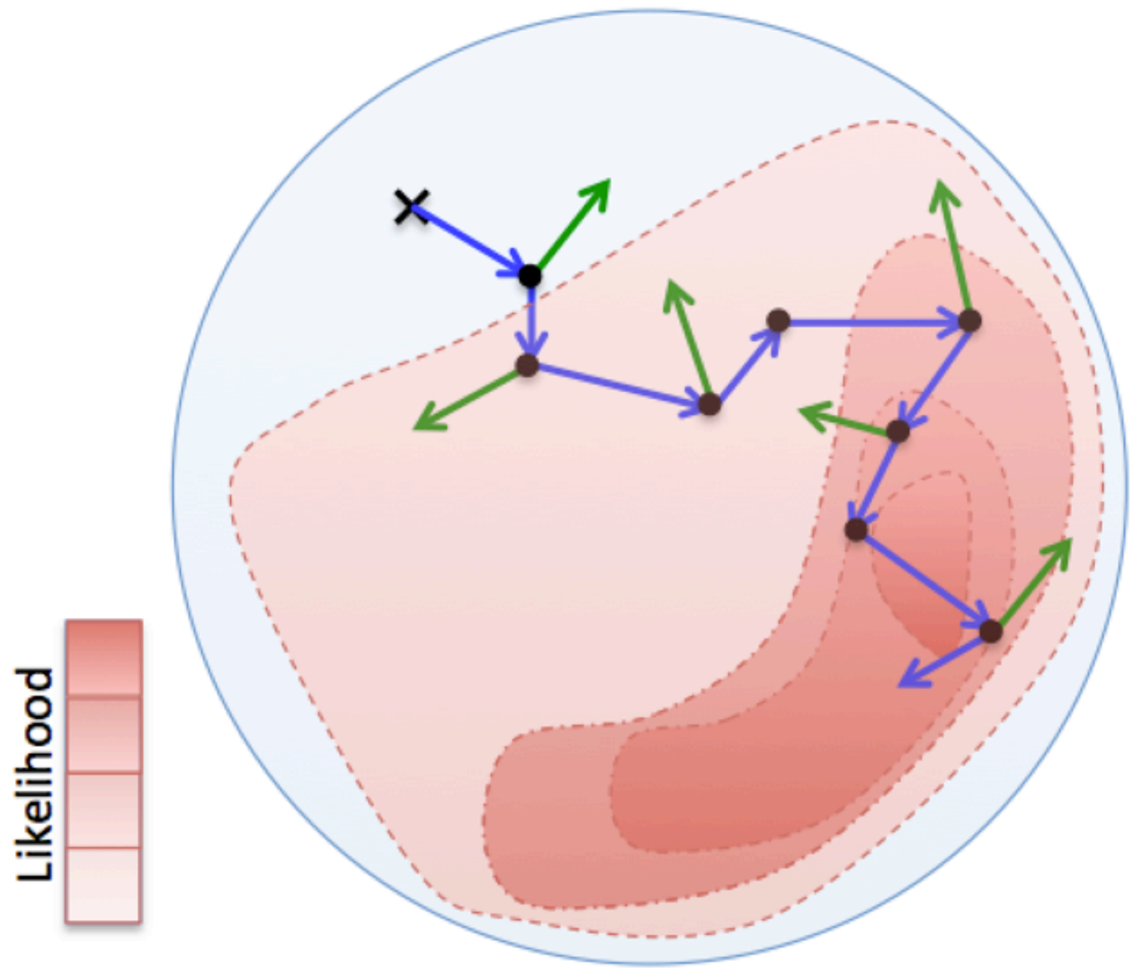
$\mathbf{x}_0$: Data

$\boldsymbol{\theta}$: Parameters

Posterior — Likelihood — Prior

$$p(\boldsymbol{\theta} \mid \mathbf{x}_0) \propto p(\mathbf{x}_0 \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$$

Metropolis-Hastings algorithm



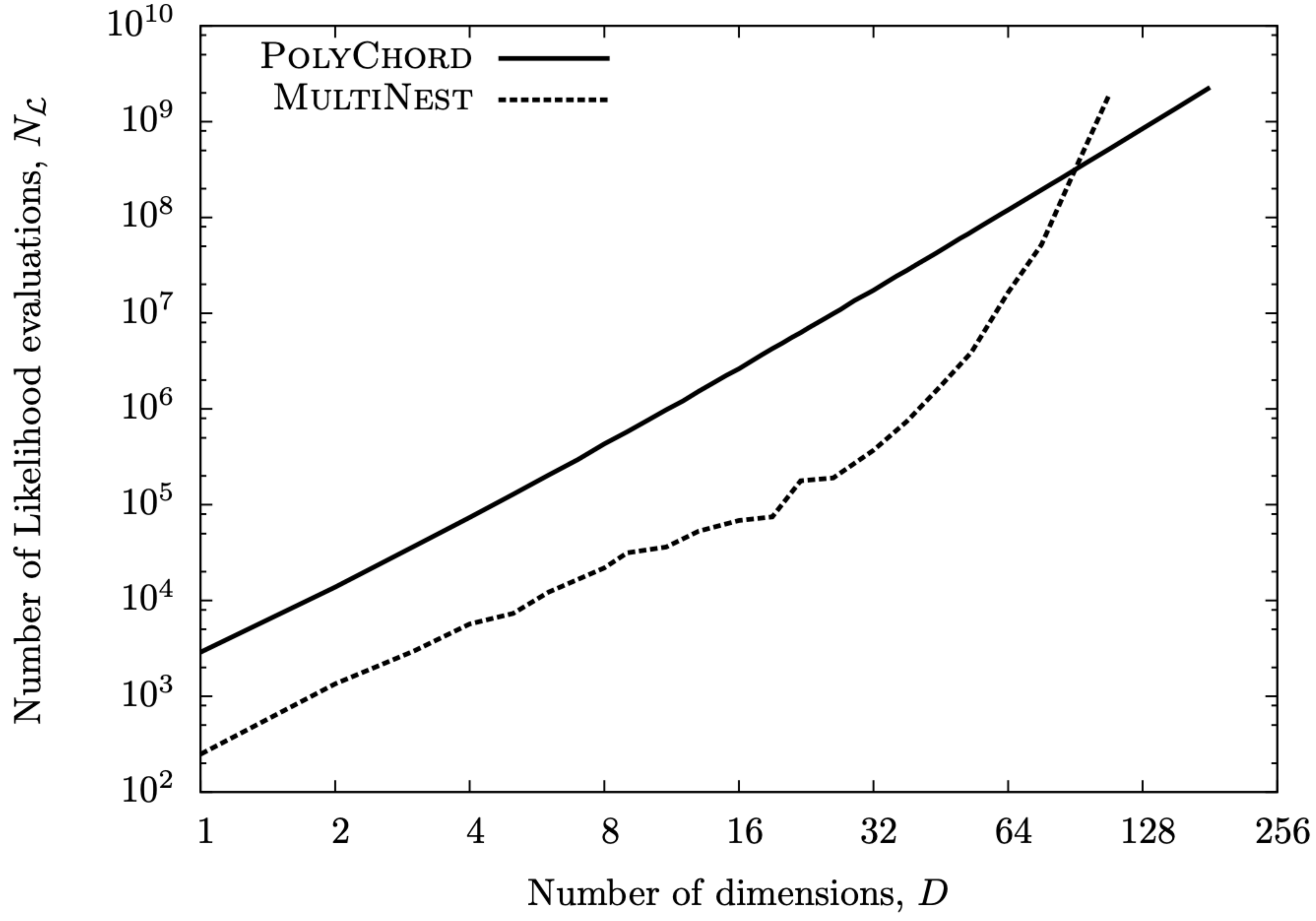
Traditional likelihood-based methods allow to get samples from the full joint posterior for some fixed data $\mathbf{x}_0$

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta} \mid \mathbf{x}_0), \qquad \text{for} \quad \boldsymbol{\theta} \in \mathbb{R}^D$$

Current challenges

- MCMC methods scale poorly with the dimensionality

[Handley+ 1506.00171]



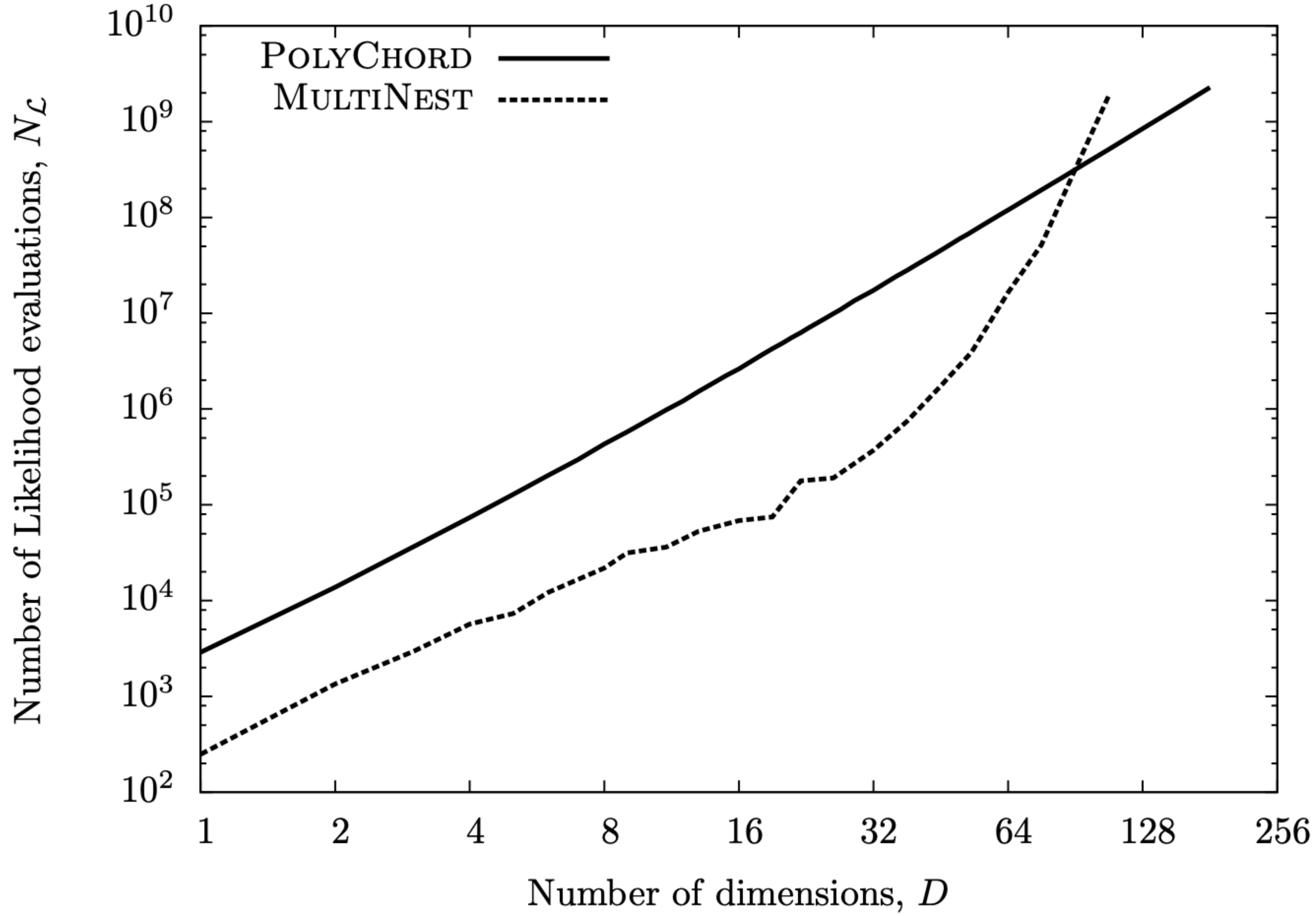
Current challenges

- MCMC methods scale poorly with the dimensionality

↓

For surveys like Euclid we expect $\sim 50 - 100$ nuisance params!

[Handley+ 1506.00171]



Current challenges

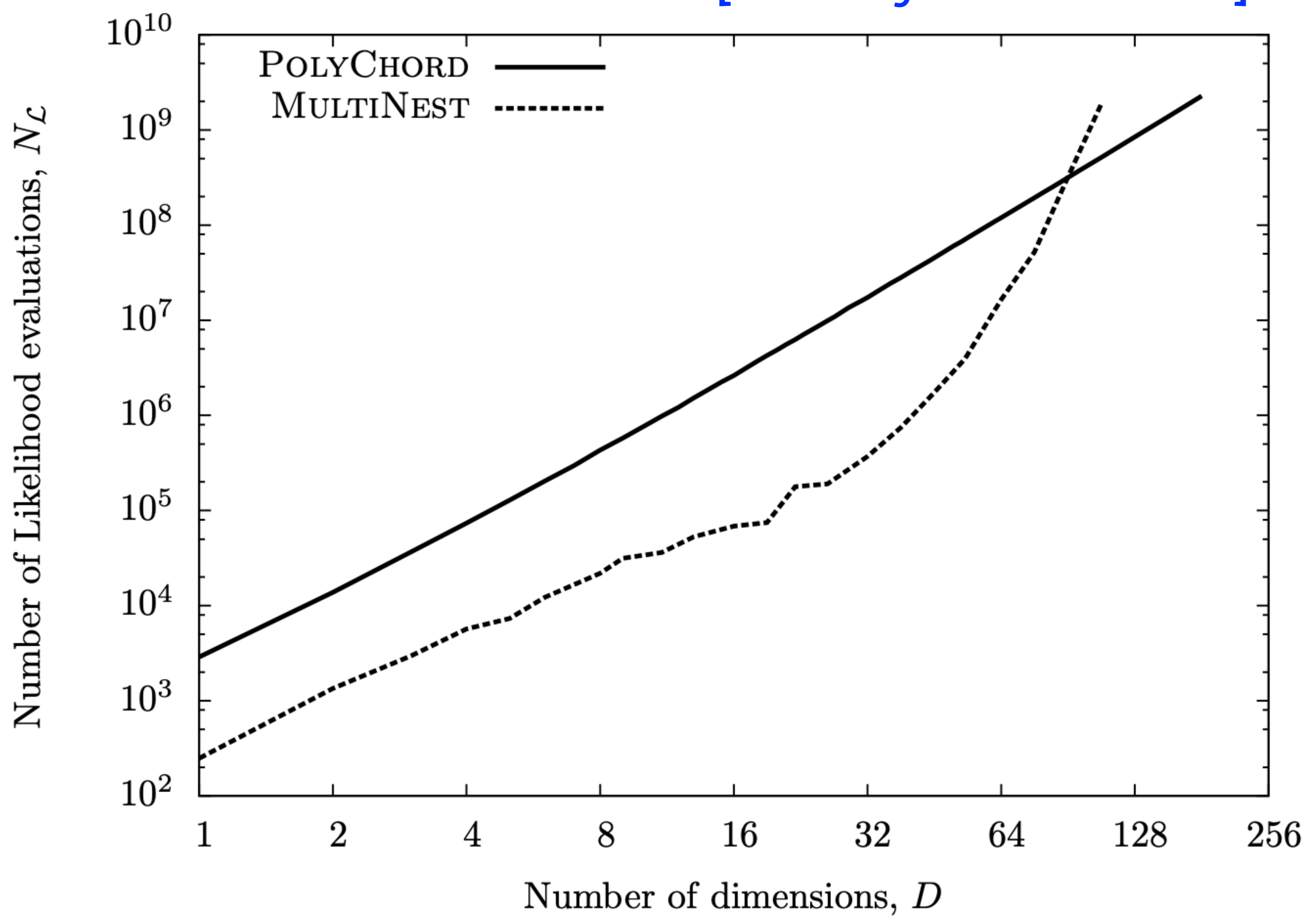
- MCMC methods scale poorly with the dimensionality

↓

For surveys like Euclid we expect $\sim 50 - 100$ nuisance params!

- Often the full likelihood function is intractable

[Handley+ 1506.00171]



Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)

"Can we still do Bayesian inference if all we can do is simulate the data?"

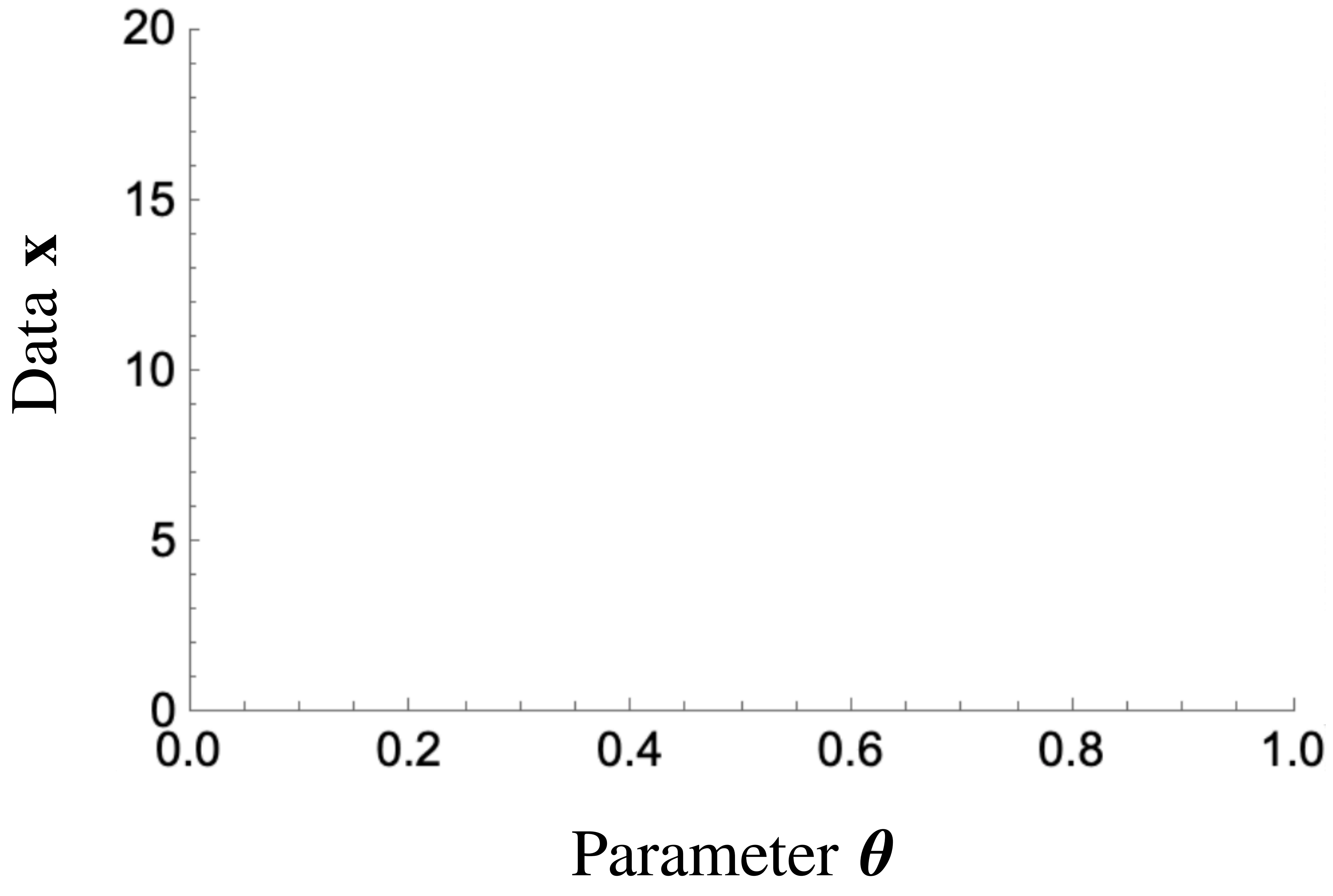
Simulation-based inference (SBI)

(a.k.a. likelihood-free or implicit inference)

"Can we still do Bayesian inference if all we can do is simulate the data?"

Simulator mapping from parameters $\boldsymbol{\theta}$ to data $\mathbf{x}$ ⟷ $\mathbf{x} \sim p(\mathbf{x} \,|\, \boldsymbol{\theta})$ **(implicit likelihood)**

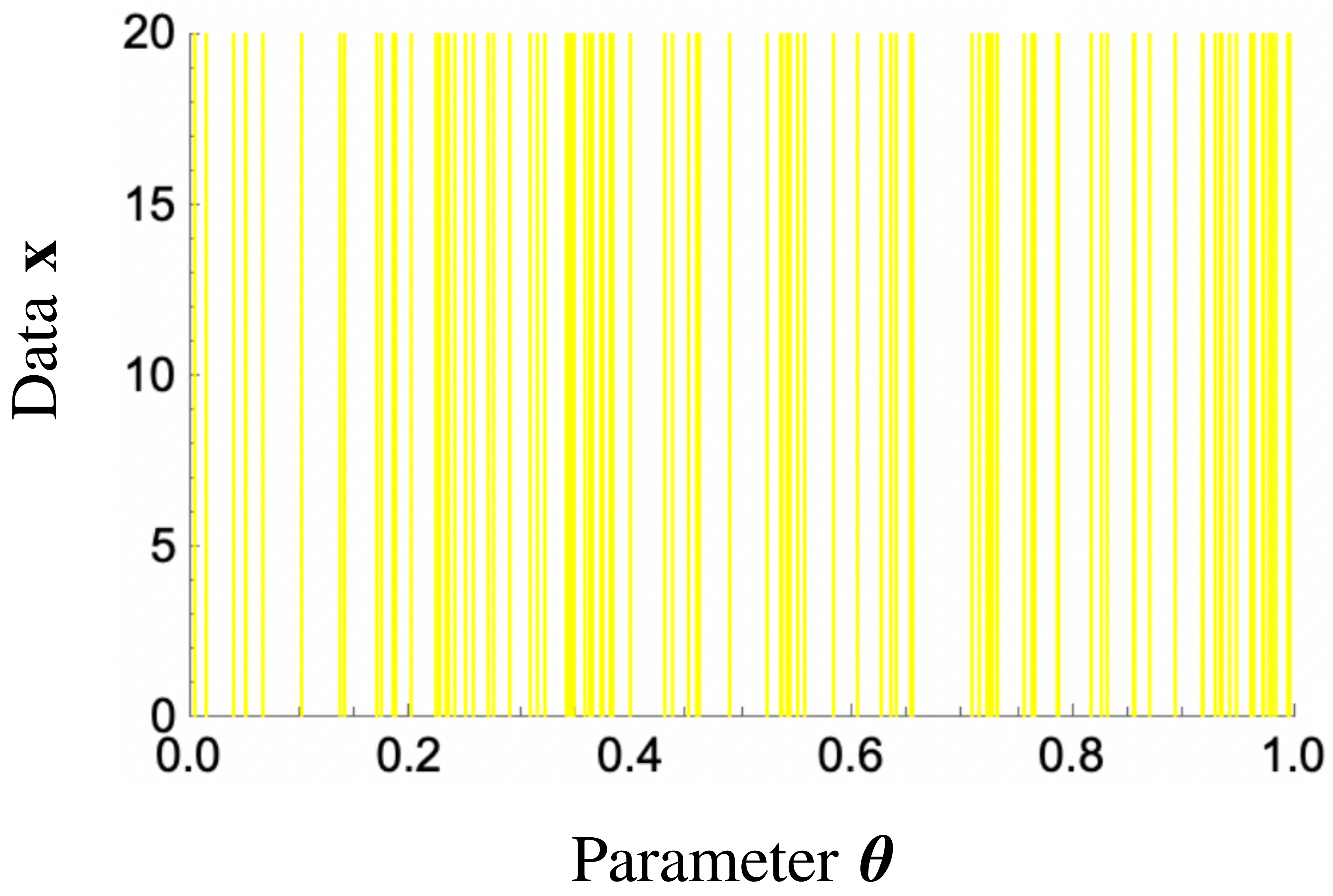
Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)



[Credit: B. Wandelt]

Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)

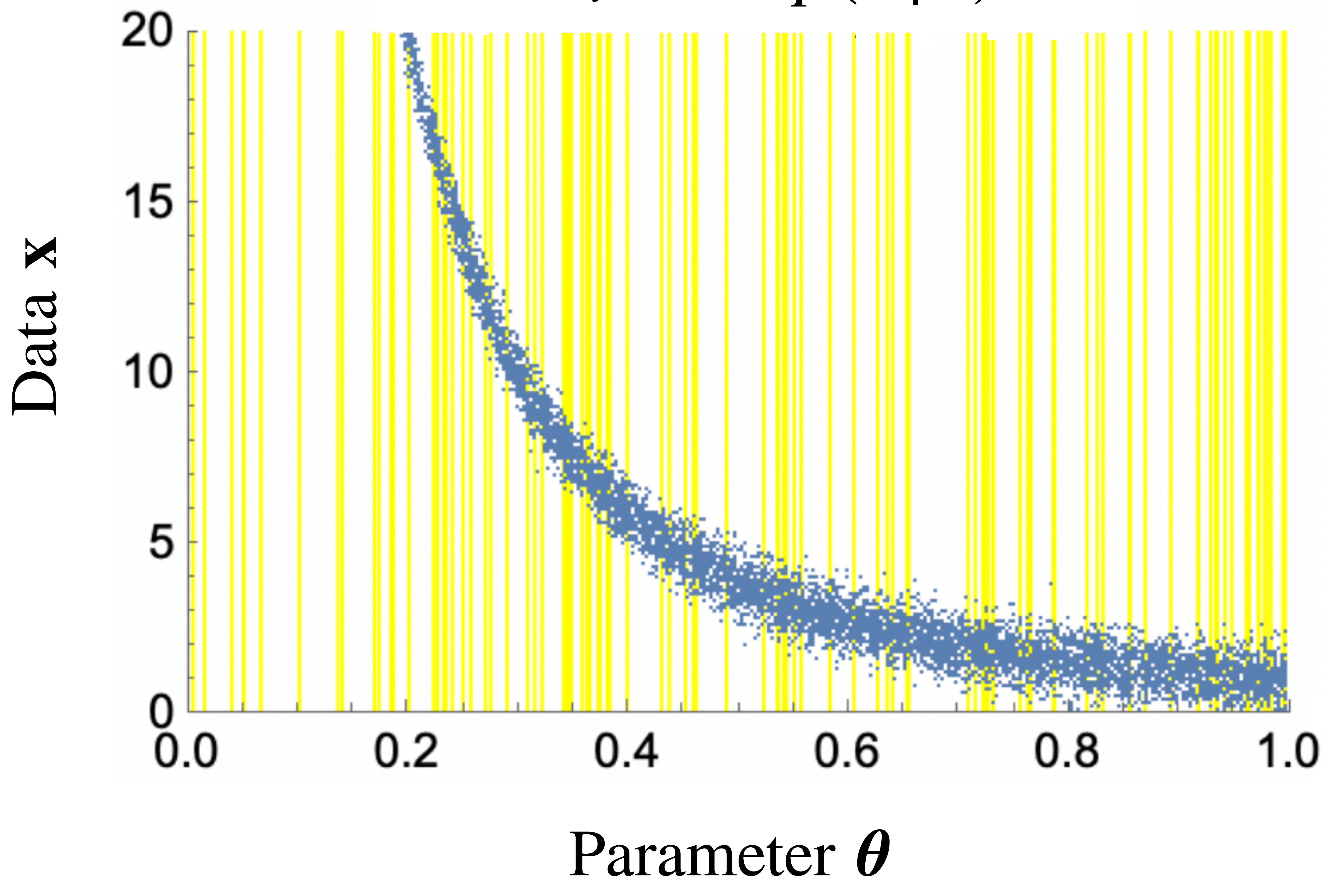
Generate $\boldsymbol{\theta}$ from prior



[Credit: B. Wandelt]

Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)

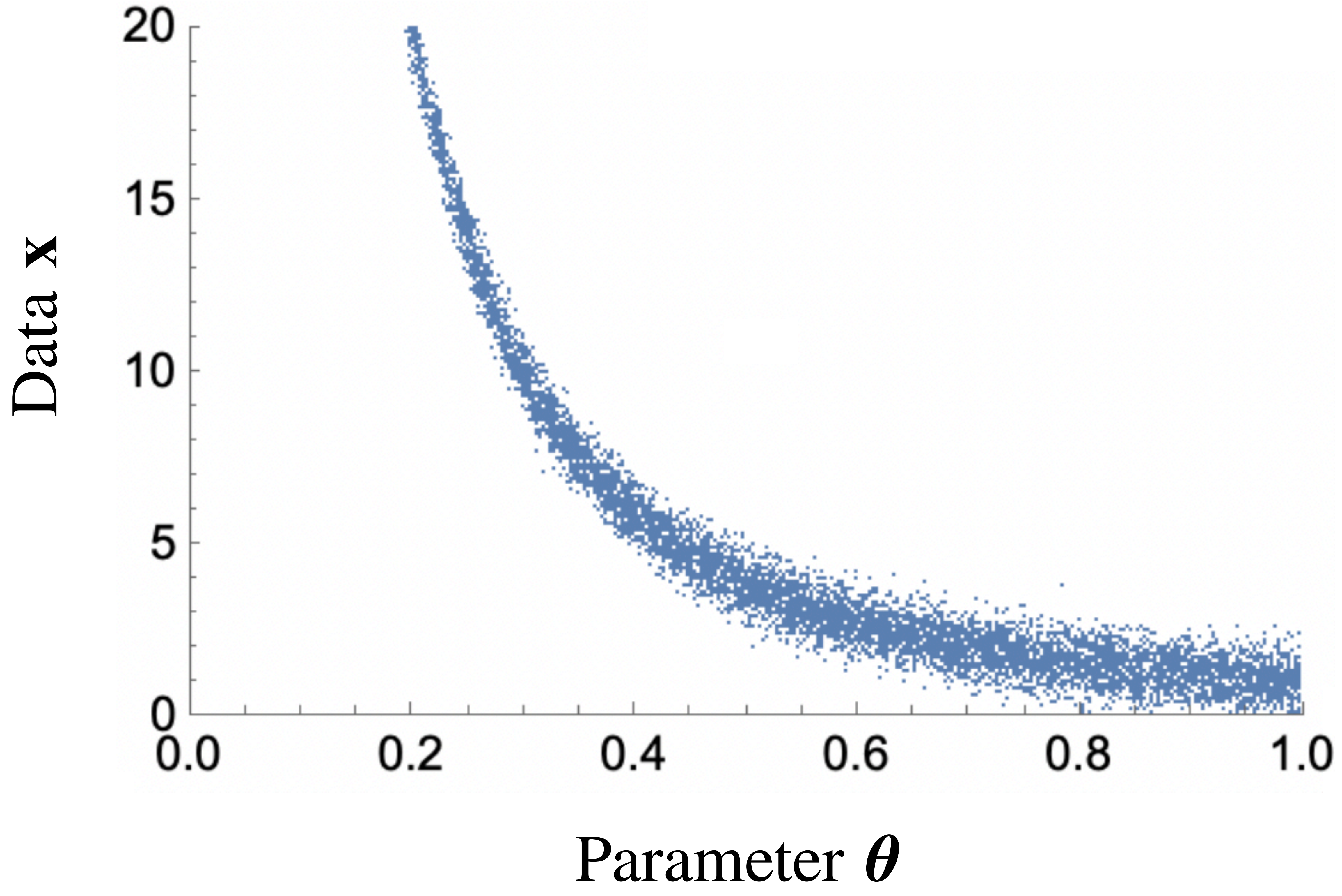
Simulate $\mathbf{x}$ for each $\boldsymbol{\theta}$
i.e., $\mathbf{x} \sim p(\mathbf{x} \mid \boldsymbol{\theta})$



[Credit: B. Wandelt]

Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)

This is a sample from $p(\boldsymbol{\theta}, \mathbf{x})$

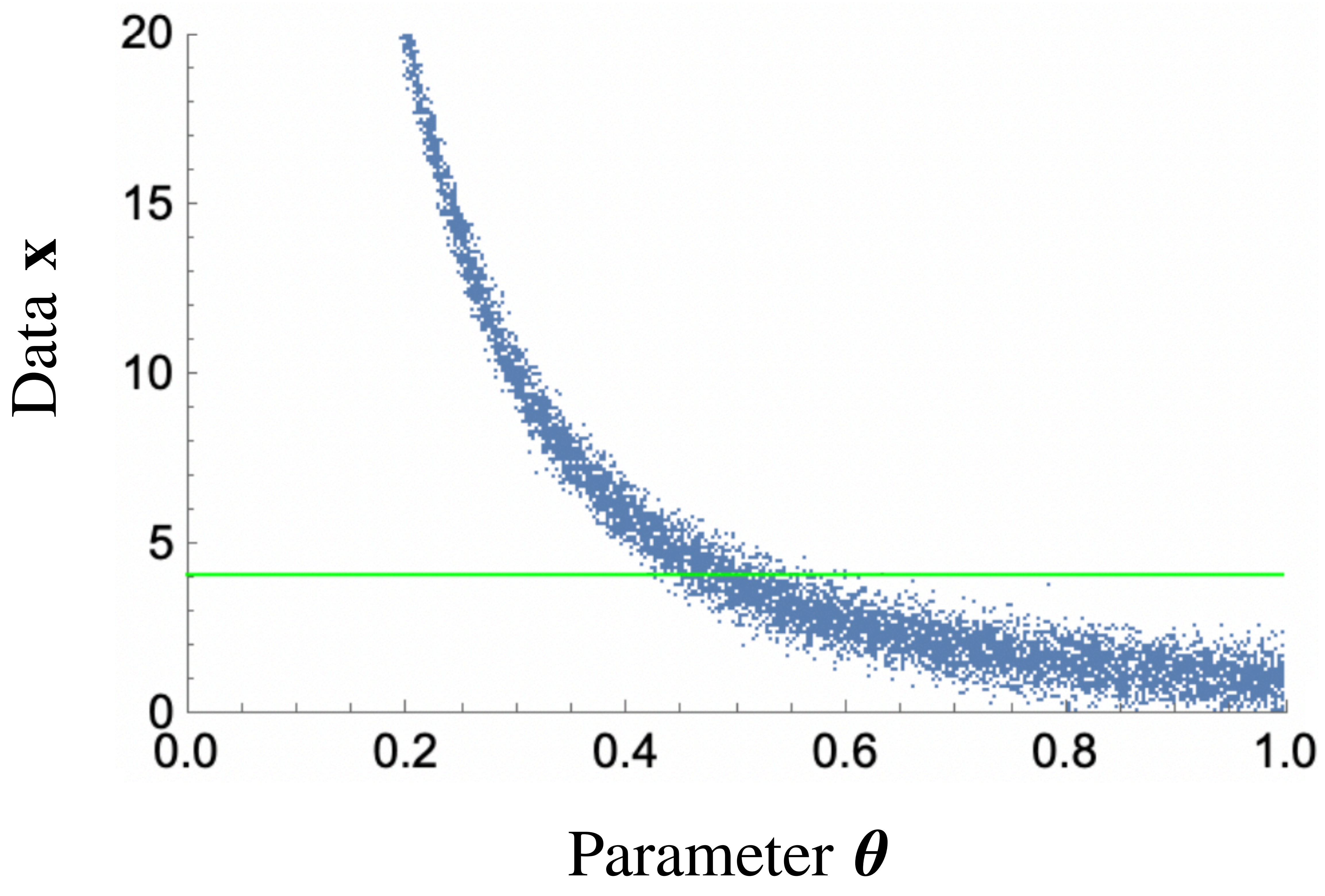


[Credit: B. Wandelt]

Simulation-based inference (SBI)

(a.k.a. likelihood-free or implicit inference)

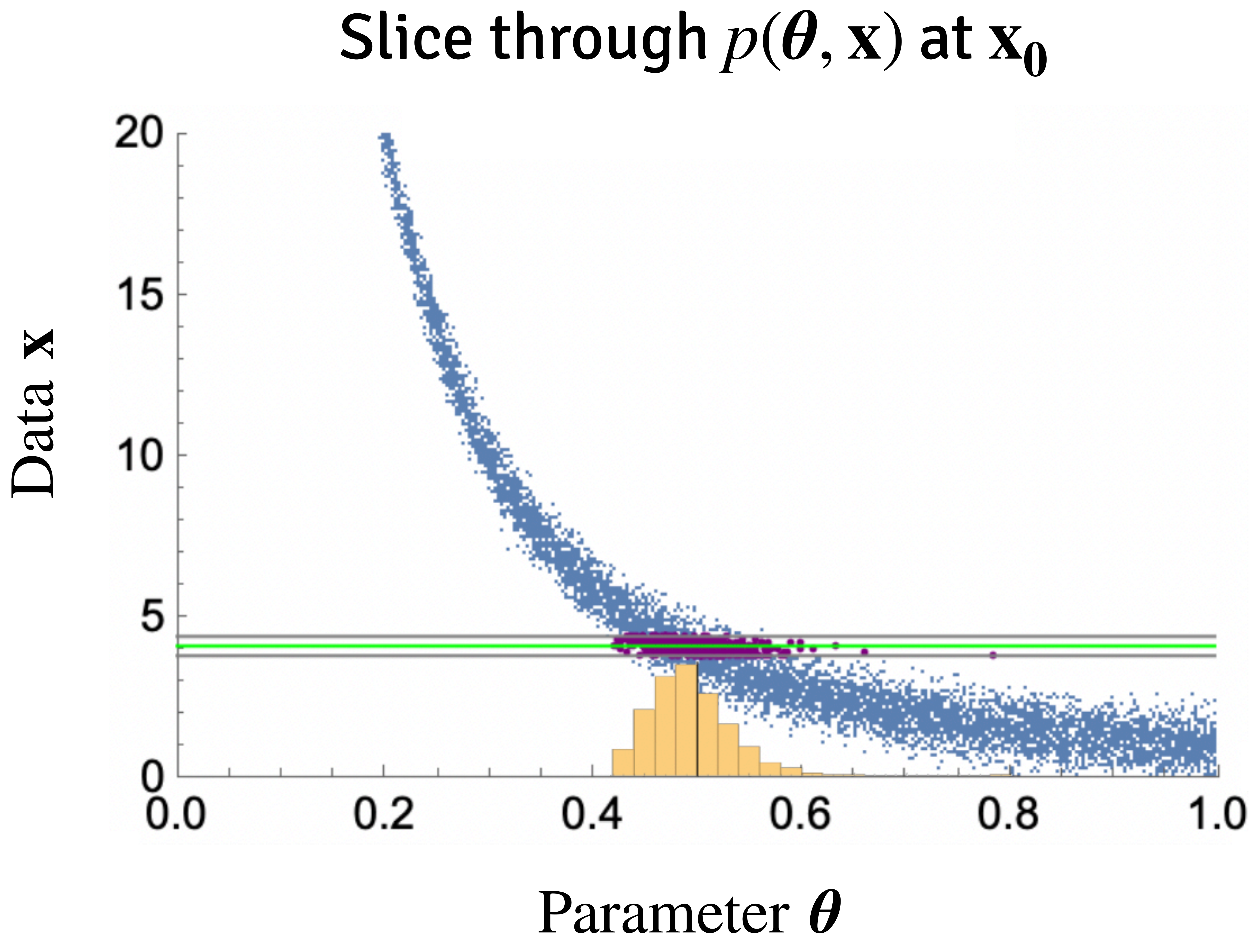
Measure actual $\mathbf{x}_0$



[Credit: B. Wandelt]

Simulation-based inference (SBI)

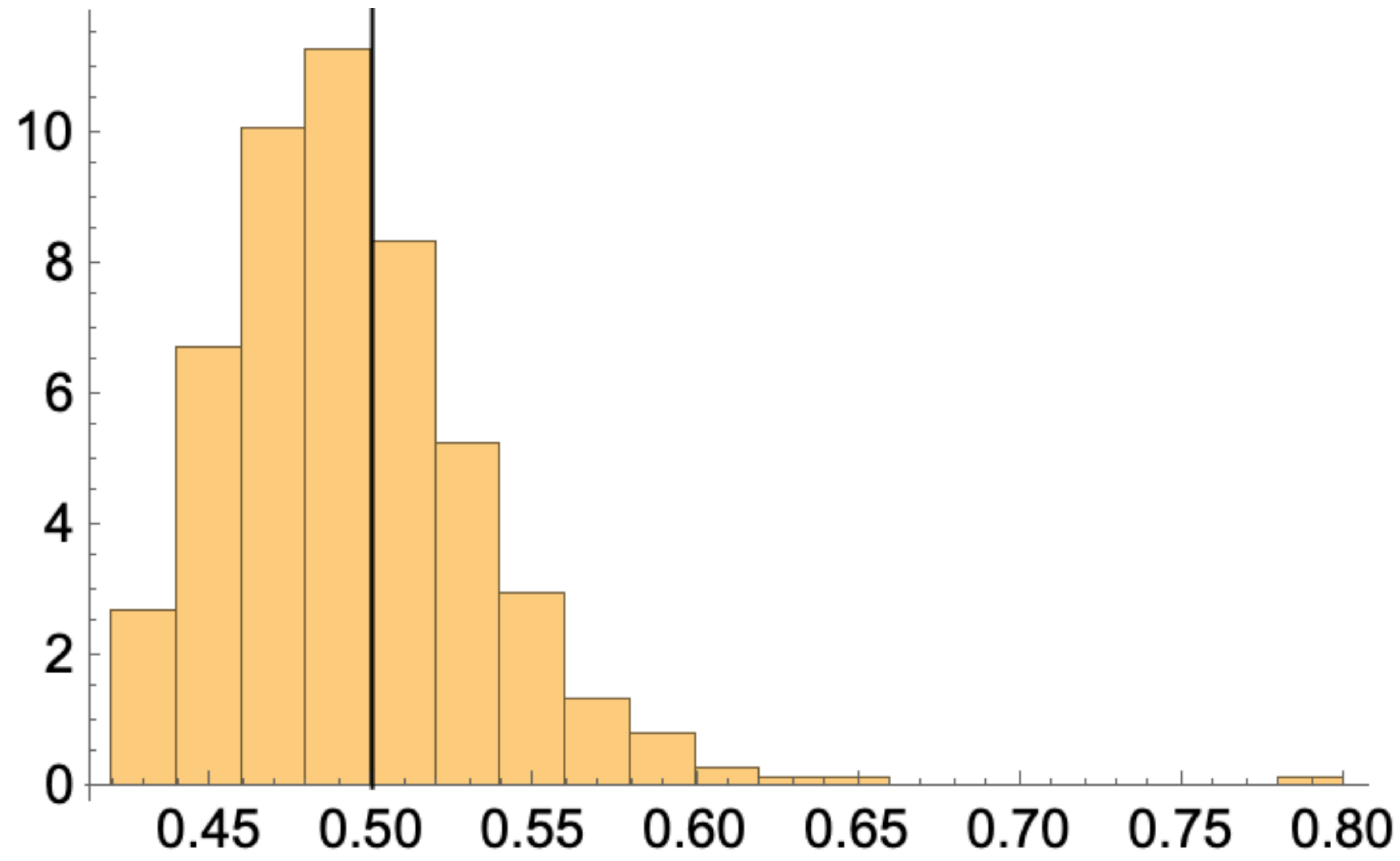
(a.k.a. likelihood-free or implicit inference)



[Credit: B. Wandelt]

Simulation-based inference (SBI)
(a.k.a. likelihood-free or implicit inference)

Posterior $p(\boldsymbol{\theta} \mid \mathbf{x}_0)$



Parameter $\boldsymbol{\theta}$

[Credit: B. Wandelt]

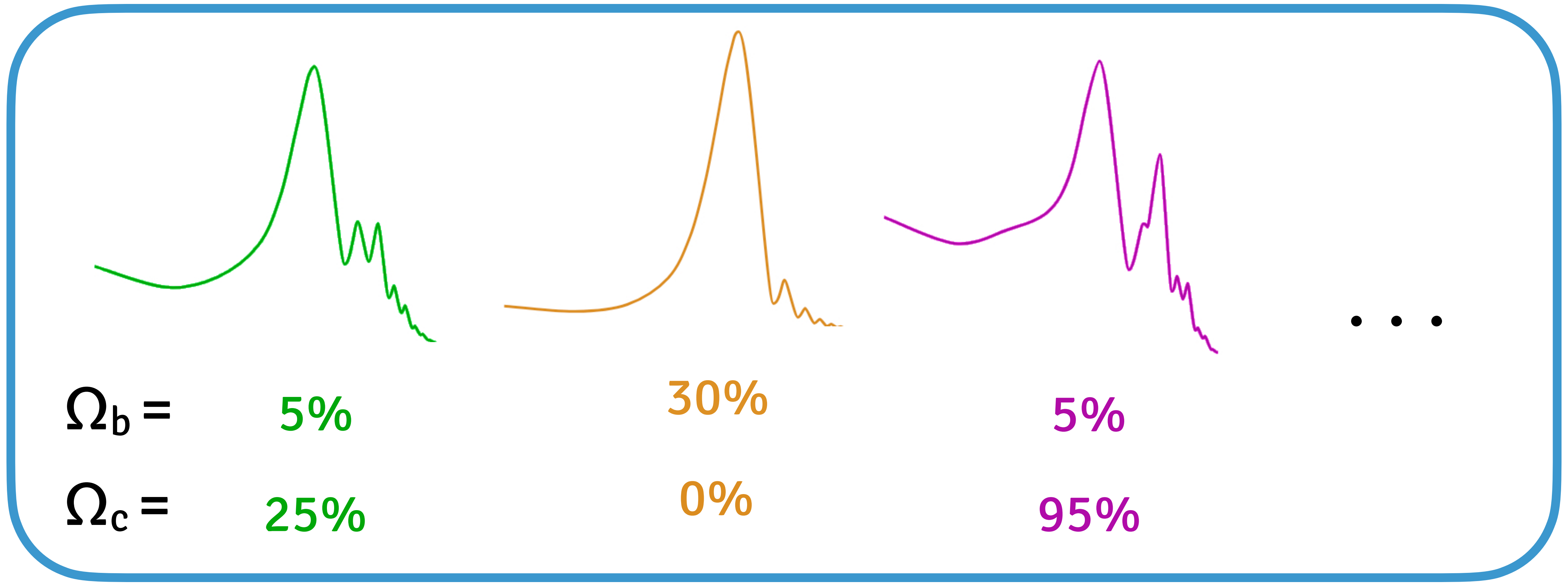
Usual steps in SBI

I. **Simulation:** get N samples $\{(\mathbf{x}^{(1)}, \boldsymbol{\theta}^{(1)}), (\mathbf{x}^{(2)}, \boldsymbol{\theta}^{(2)}), \ldots, (\mathbf{x}^{(N)}, \boldsymbol{\theta}^{(N)})\}$

Usual steps in SBI

I. Simulation: get N samples $\{(\mathbf{x}^{(1)}, \boldsymbol{\theta}^{(1)}), (\mathbf{x}^{(2)}, \boldsymbol{\theta}^{(2)}), \ldots, (\mathbf{x}^{(N)}, \boldsymbol{\theta}^{(N)})\}$

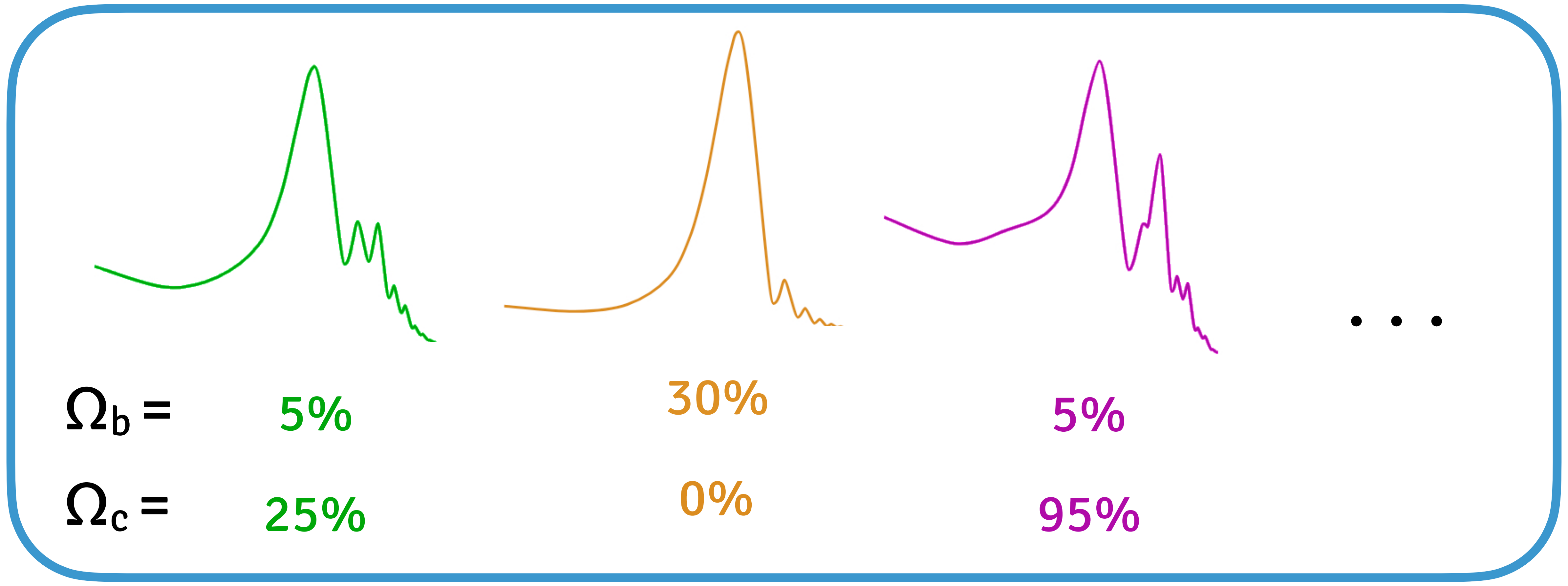
Ex: CMB simulator



Usual steps in SBI

I. Simulation: get N samples $\{(\mathbf{x}^{(1)}, \boldsymbol{\theta}^{(1)}), (\mathbf{x}^{(2)}, \boldsymbol{\theta}^{(2)}), \ldots, (\mathbf{x}^{(N)}, \boldsymbol{\theta}^{(N)})\}$

Ex: CMB simulator

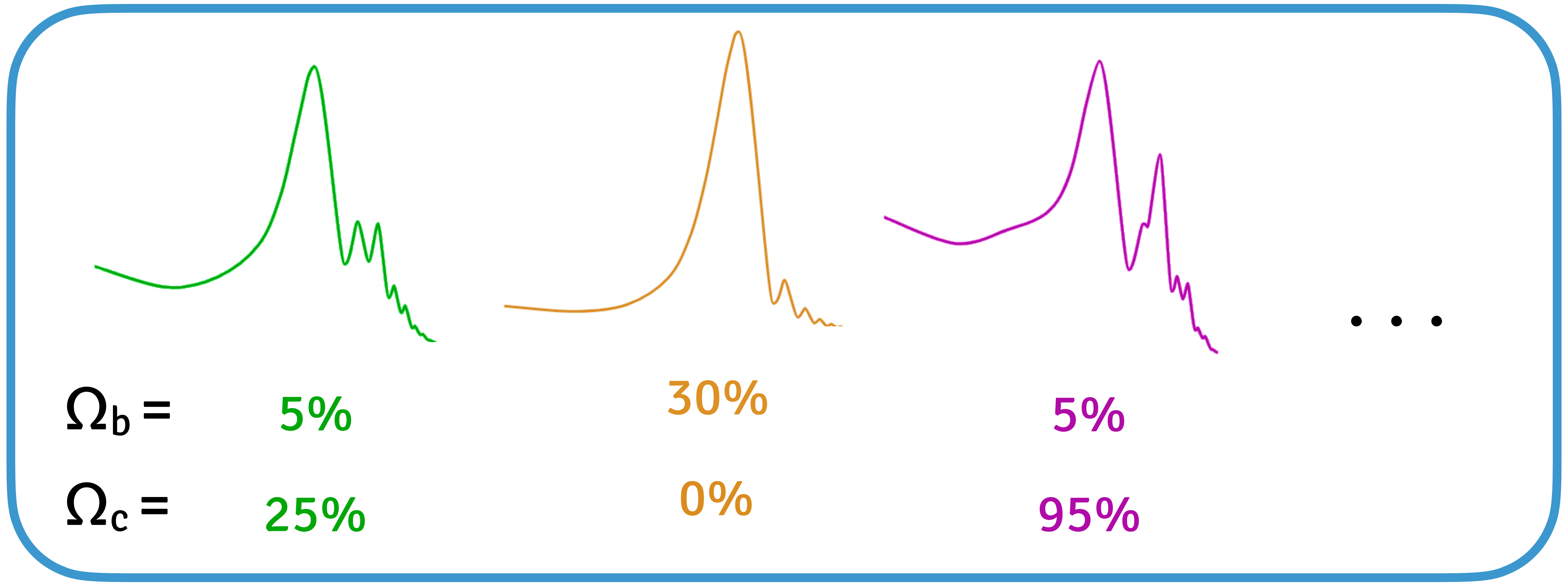


II. Training: train a NN to learn posterior $f_\phi(\boldsymbol{\theta}, \mathbf{x}) \simeq p(\boldsymbol{\theta} \,|\, \mathbf{x})$

Usual steps in SBI

I. Simulation: get N samples $\{(\mathbf{x}^{(1)}, \boldsymbol{\theta}^{(1)}), (\mathbf{x}^{(2)}, \boldsymbol{\theta}^{(2)}), \ldots, (\mathbf{x}^{(N)}, \boldsymbol{\theta}^{(N)})\}$

Ex: CMB simulator



II. Training: train a NN to learn posterior $f_\phi(\boldsymbol{\theta}, \mathbf{x}) \simeq p(\boldsymbol{\theta} \,|\, \mathbf{x})$

III. Inference: evaluate trained network at $\mathbf{x} = \mathbf{x_0}$ to get $p(\boldsymbol{\theta} \,|\, \mathbf{x_0})$

SBI comes in many “flavors”:

Neural posterior estimation (**NPE**) $\longrightarrow p(\boldsymbol{\theta} \,|\, \mathbf{x})$

Neural likelihood estimation (**NLE**) $\longrightarrow p(\mathbf{x} \,|\, \boldsymbol{\theta})$

Neural ratio estimation (**NRE**) $\longrightarrow p(\boldsymbol{\theta} \,|\, \mathbf{x})/p(\boldsymbol{\theta})$

SBI comes in many "flavors":

Neural posterior estimation **(NPE)** $\longrightarrow p(\boldsymbol{\theta}\,|\,\mathbf{x})$

Neural likelihood estimation **(NLE)** $\longrightarrow p(\mathbf{x}\,|\,\boldsymbol{\theta})$

Neural ratio estimation **(NRE)** $\longrightarrow p(\boldsymbol{\theta}\,|\,\mathbf{x})/p(\boldsymbol{\theta})$

We focus on a recent algorithm:

MNRE = **M**arginal **N**eural **R**atio **E**stimation

Implemented in Swyft [Miller et al. 2011.13951]

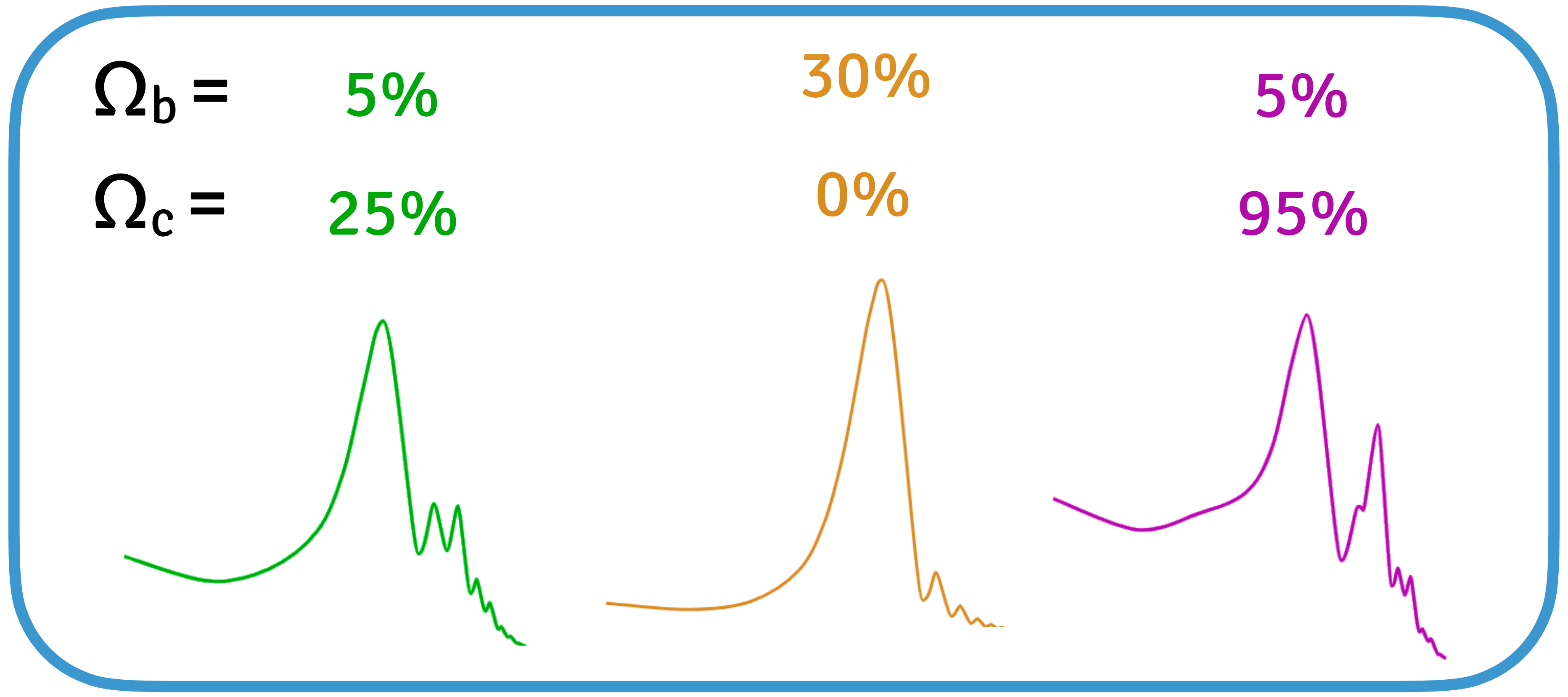
Neural Ratio Estimation

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

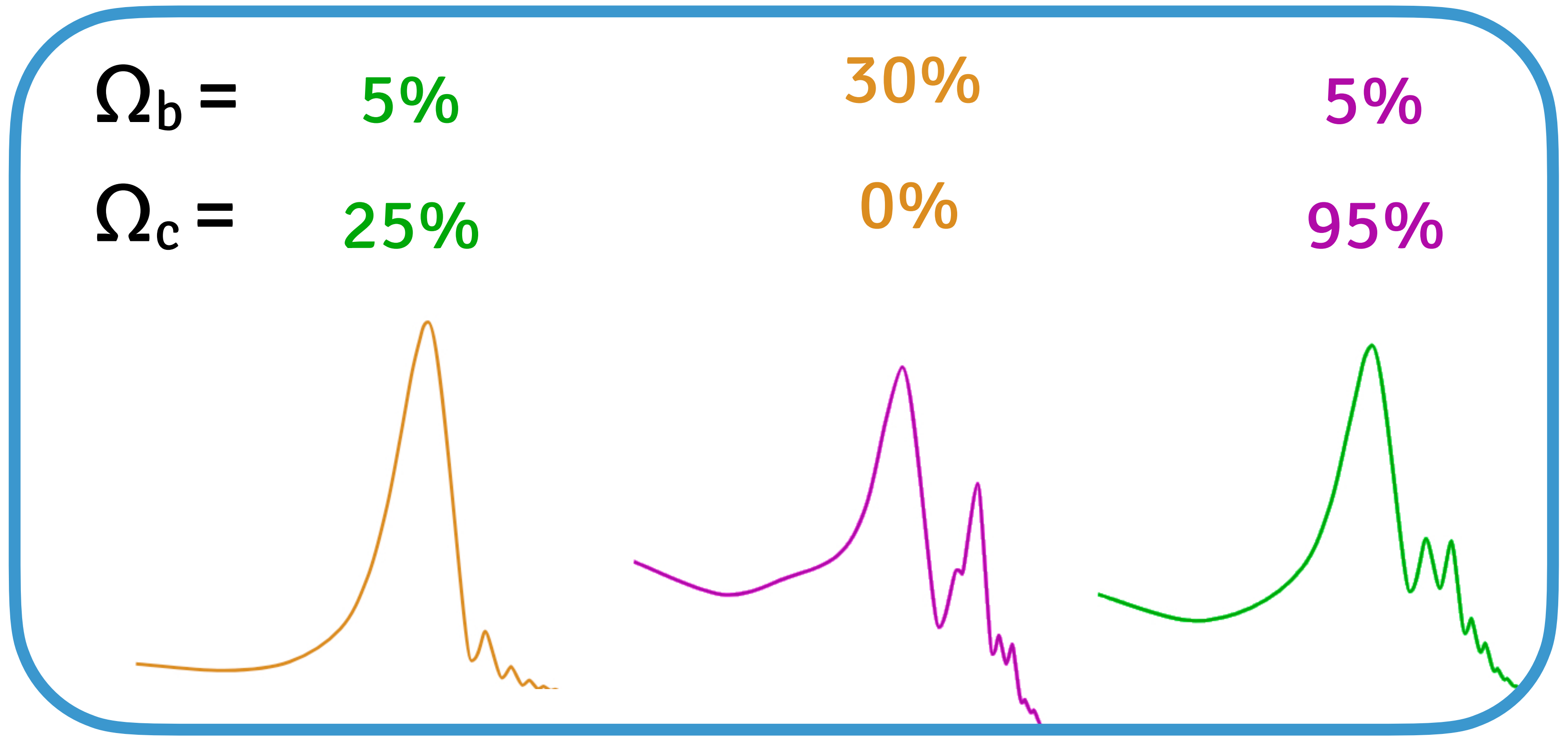
MNRE in a nutshell

Train NNs to solve a binary classification problem:

$$(\mathbf{x}, \boldsymbol{\theta}) \sim p(\mathbf{x}, \boldsymbol{\theta})$$



$$(\mathbf{x}, \boldsymbol{\theta}) \sim p(\mathbf{x})p(\boldsymbol{\theta})$$



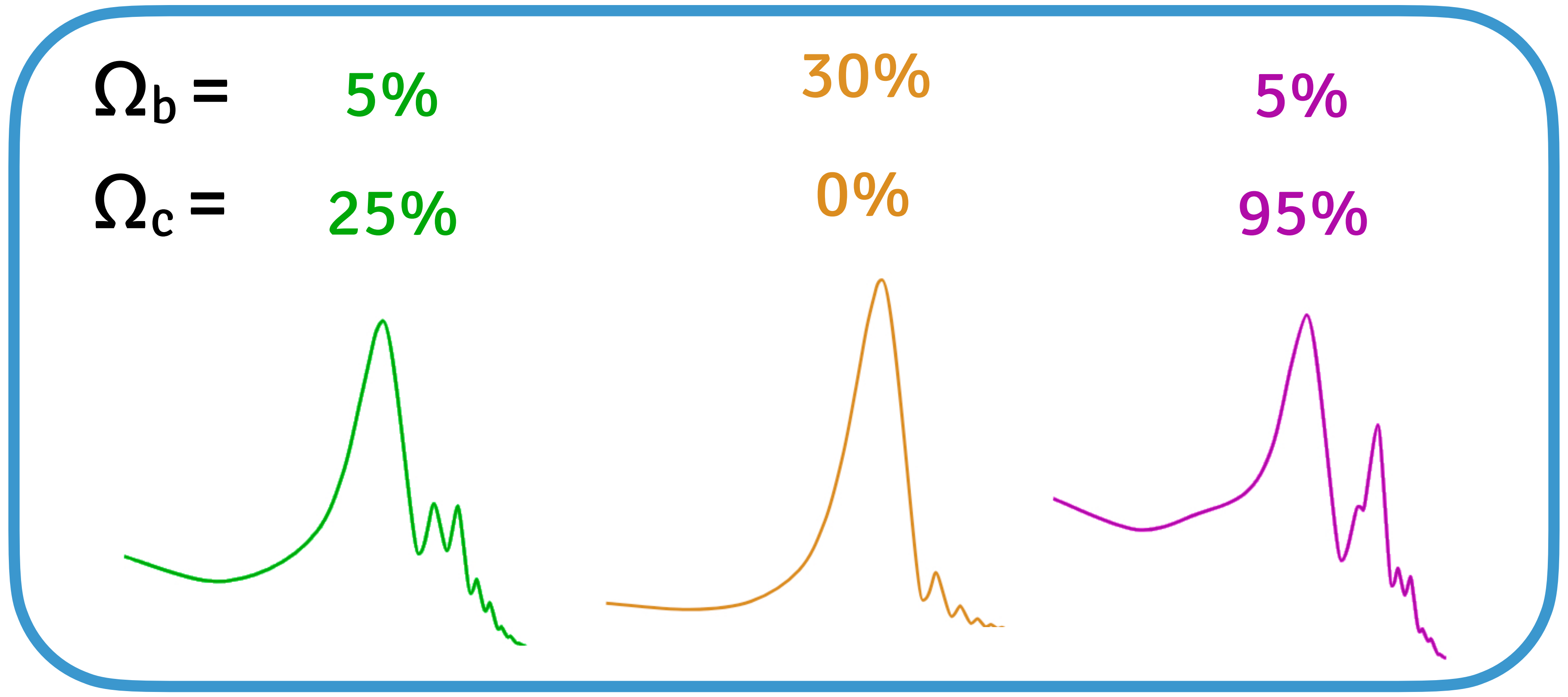
MNRE in a nutshell

Neural Ratio Estimation

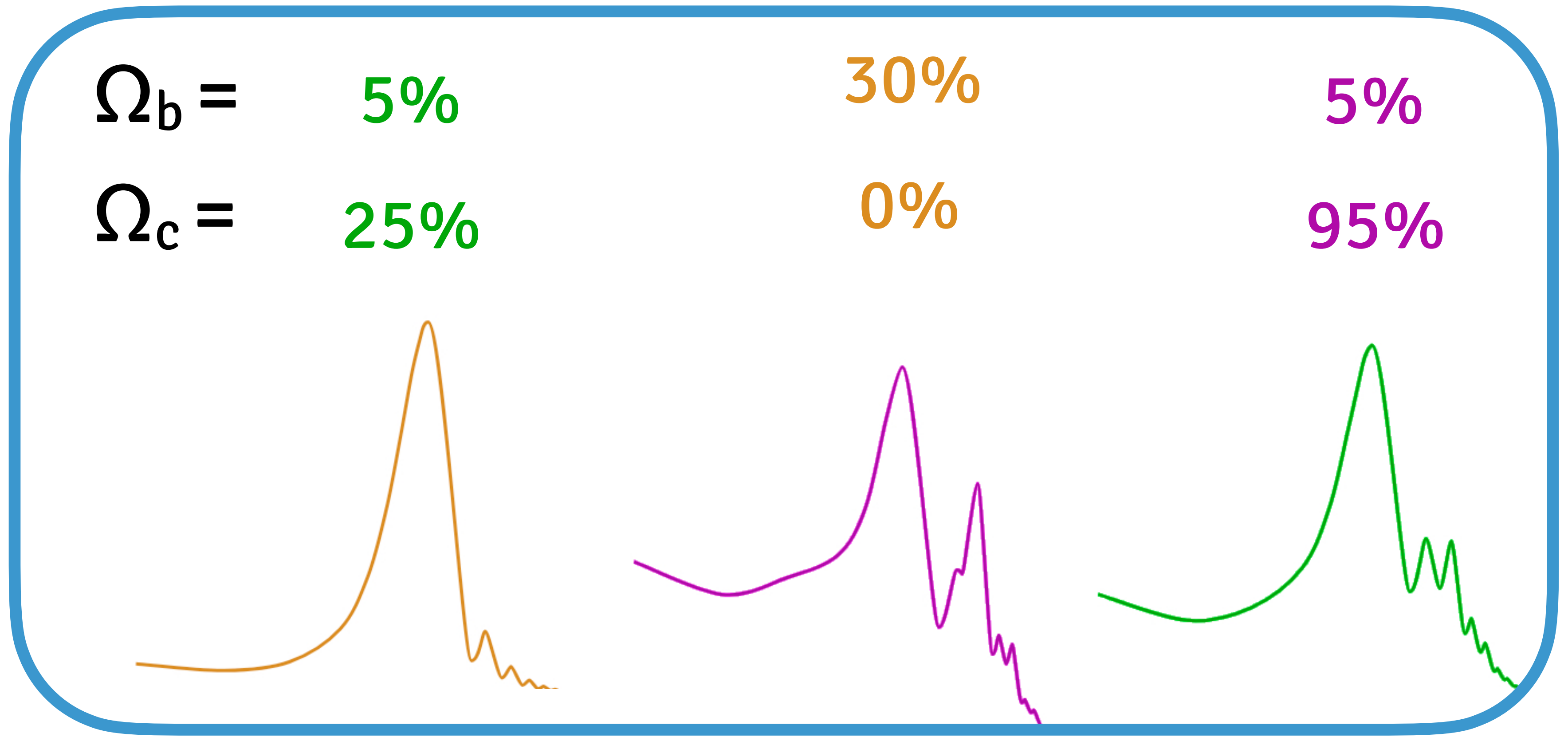
$$\frac{p(\mathbf{x},\boldsymbol{\theta})}{p(\mathbf{x})p(\boldsymbol{\theta})} = \frac{p(\boldsymbol{\theta}\,|\,\mathbf{x})}{p(\boldsymbol{\theta})}$$

Train NNs to solve a binary classification problem:

$$(\mathbf{x},\boldsymbol{\theta}) \sim p(\mathbf{x},\boldsymbol{\theta})$$

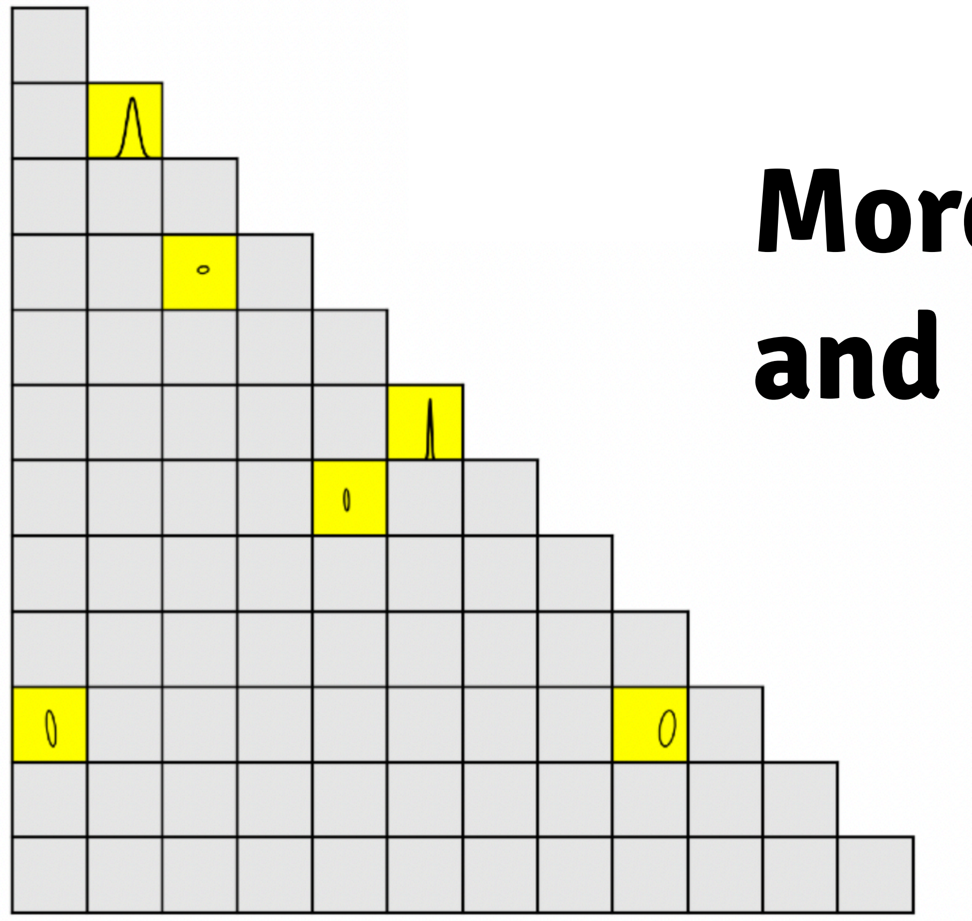


$$(\mathbf{x},\boldsymbol{\theta}) \sim p(\mathbf{x})p(\boldsymbol{\theta})$$



Focus on marginals

Instead of estimating all parameters, cherry-pick the ones we care about



More flexible and efficient!

SBI has already been applied to different LSS surveys:

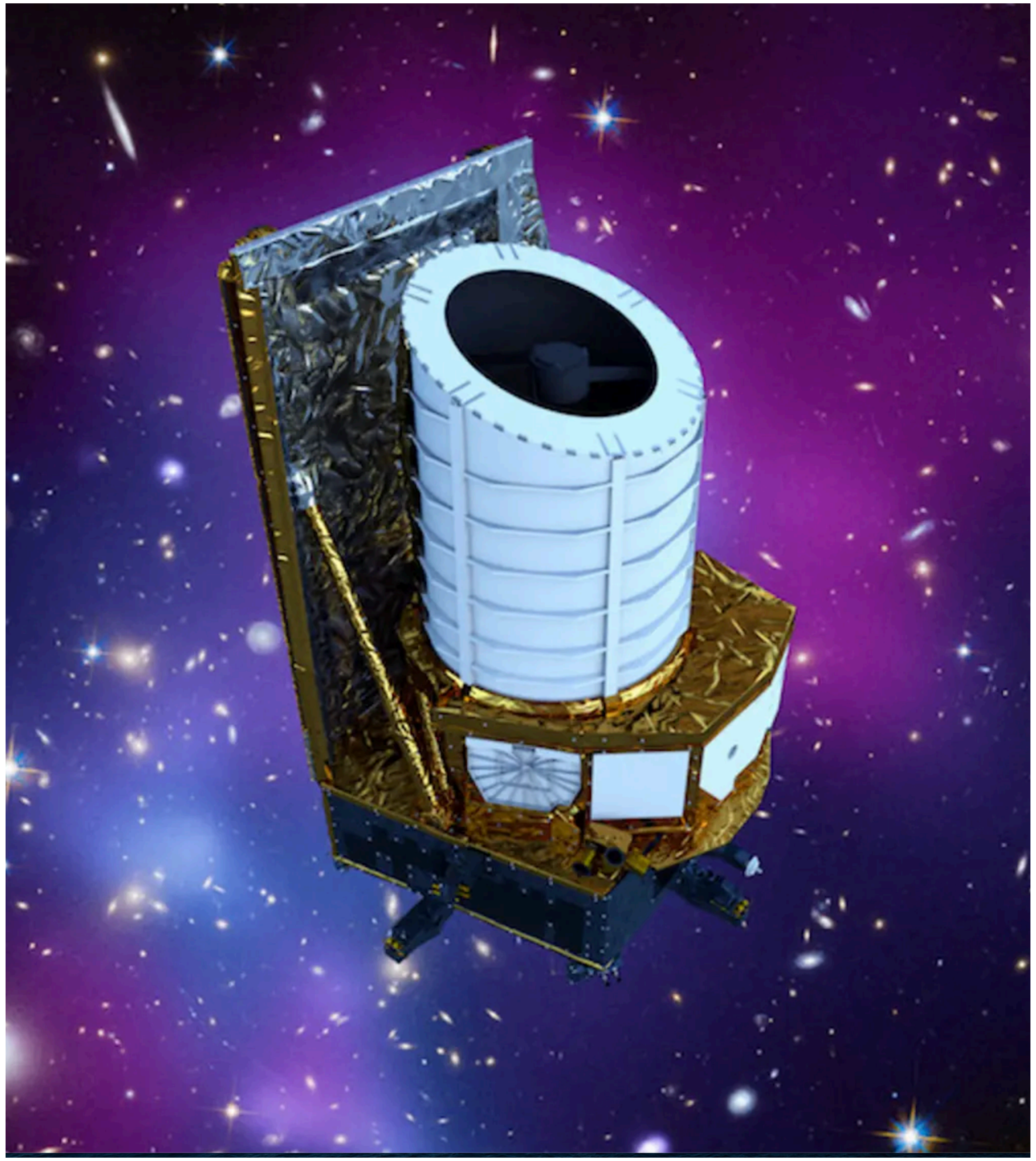
- **BOSS** [Lemos et al. 2310.15256]
- **DES** [Jeffrey et al. 2403.02314]
- **KiDS** [von Wietersheim-Kramsta et al. 2404.15402]
- **HSC** [Novaes et al. 2409.01301]

SBI has already been applied to different LSS surveys:

- **BOSS** [Lemos et al. 2310.15256]
- **DES** [Jeffrey et al. 2403.02314]
- **KiDS** [von Wietersheim-Kramsta et al. 2404.15402]
- **HSC** [Novaes et al. 2409.01301]

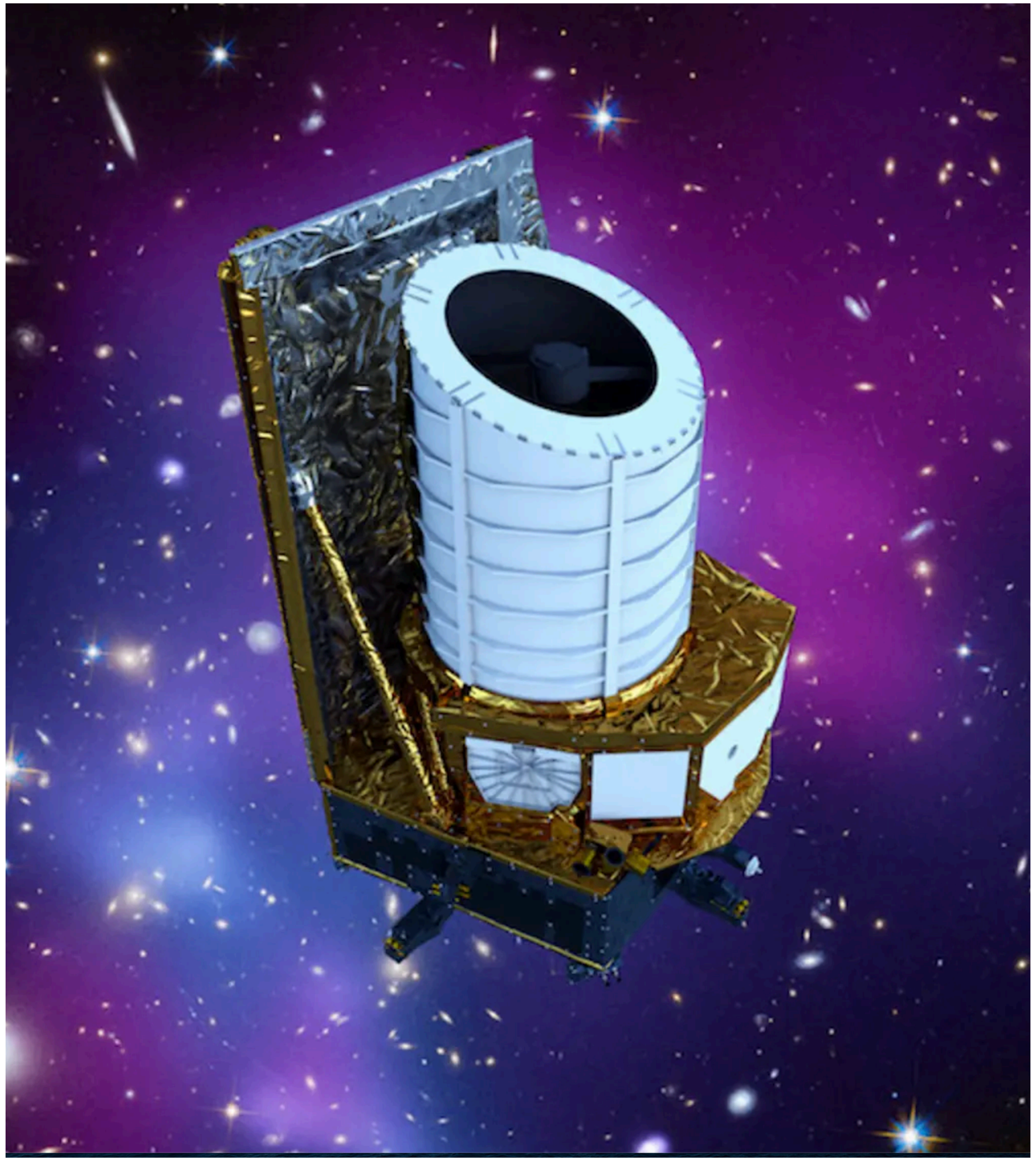
Our goal: apply MNRE to **Euclid** photometric observables

[ESA's Euclid space satellite]



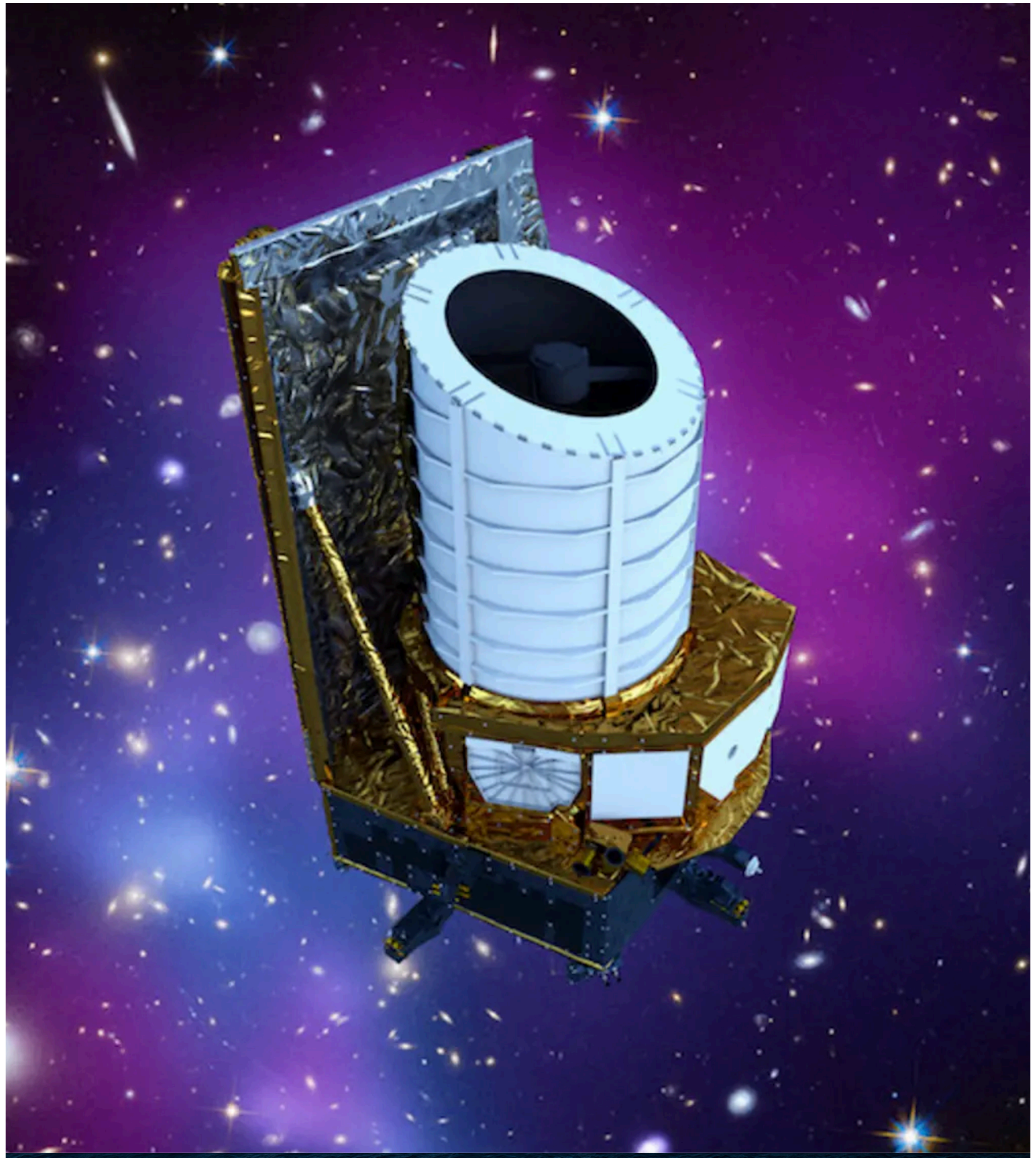
On July 1st 2023, Euclid was launched to L2

[ESA's Euclid space satellite]



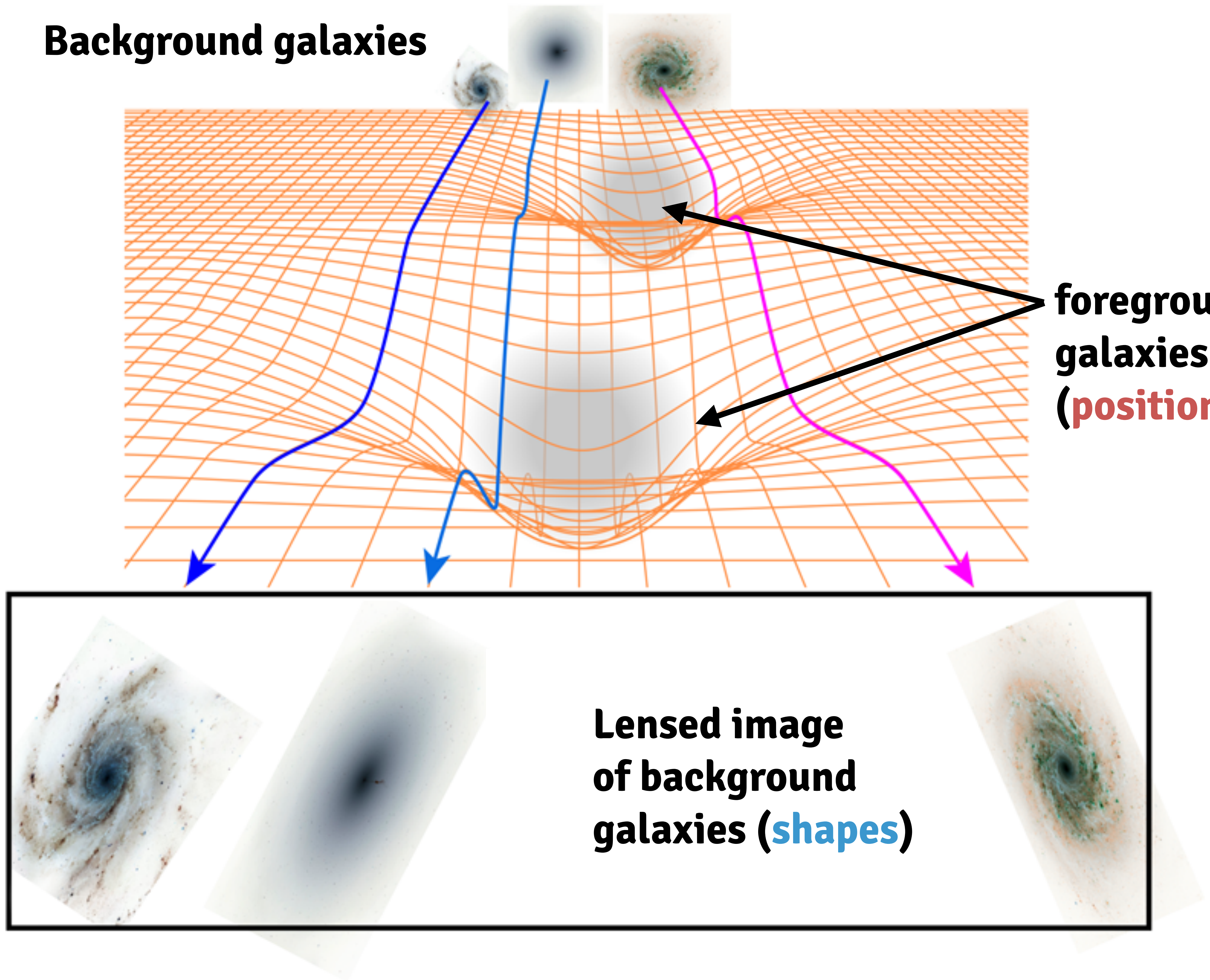
- On July 1st 2023, Euclid was launched to L2
- For 6 years, Euclid will measure the shapes and redshifts of billions of galaxies, across ~1/3 of the sky

[ESA's Euclid space satellite]



- On July 1st 2023, Euclid was launched to L2
- For 6 years, Euclid will measure the shapes and redshifts of billions of galaxies, across ~1/3 of the sky
- First public data release expected in 2026

Which are the Euclid primary observables?



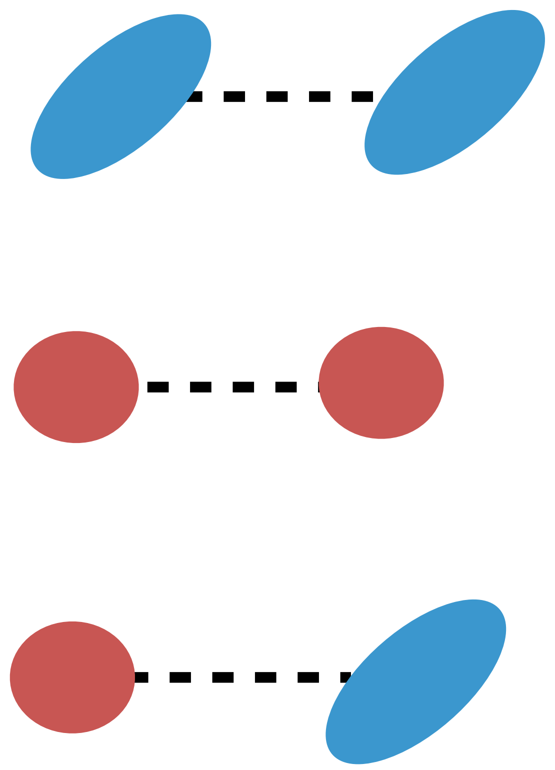
3x2pt photometric probes

Summarise maps of galaxy positions/shapes using three 2-point statistics **(3x2pt)** measured at 10 tomographic redshift bins

Cosmic Shear

Galaxy clustering

Galaxy-Galaxy lensing



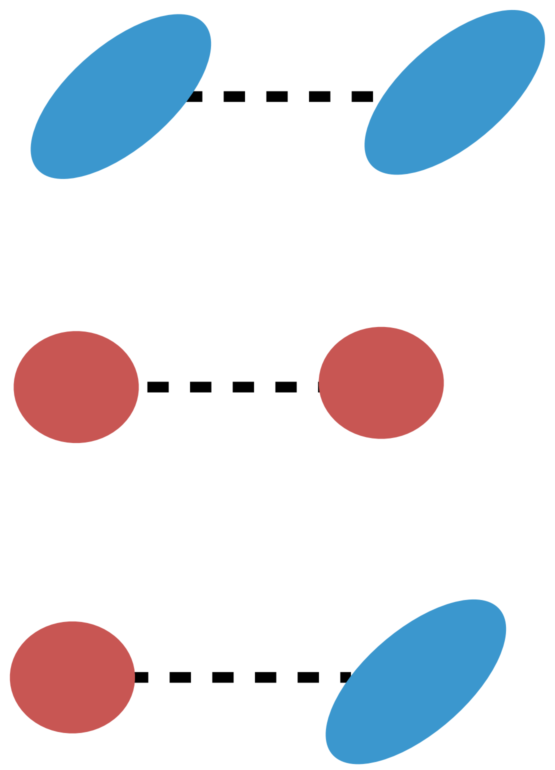
3x2pt photometric probes

Summarise maps of galaxy positions/shapes using three 2-point statistics **(3x2pt)** measured at 10 tomographic redshift bins

Cosmic Shear

Galaxy clustering

Galaxy-Galaxy lensing



...described by angular power spectra $C_{ij}^{XY}(\ell) = \int dz\ W_i^X(z) W_j^Y(z)\ P_m(k_\ell, z)$

NOTE: We use a simplified simulator/likelihood (12 nuisance params)

Swyft analysis of Euclid 3x2pt

1. Simulator:

We generate 50k realisations of 3x2pt spectra with gaussian noise

$$\hat{C}_{ij}^{AB}(\ell) = C_{ij}^{AB}(\ell) + n_{ij}^{AB}(\ell)$$

$n_{ij}^{AB}(\ell) \leftarrow \mathcal{N}(0,\mathbf{C})$

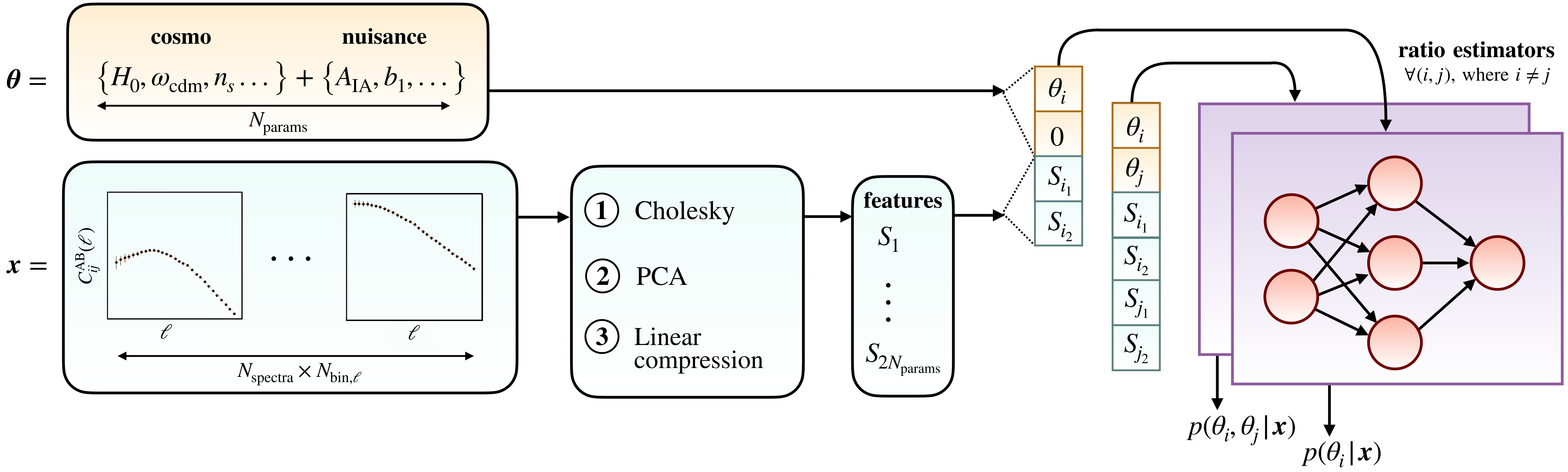
Swyft analysis of Euclid 3x2pt

1. Simulator:

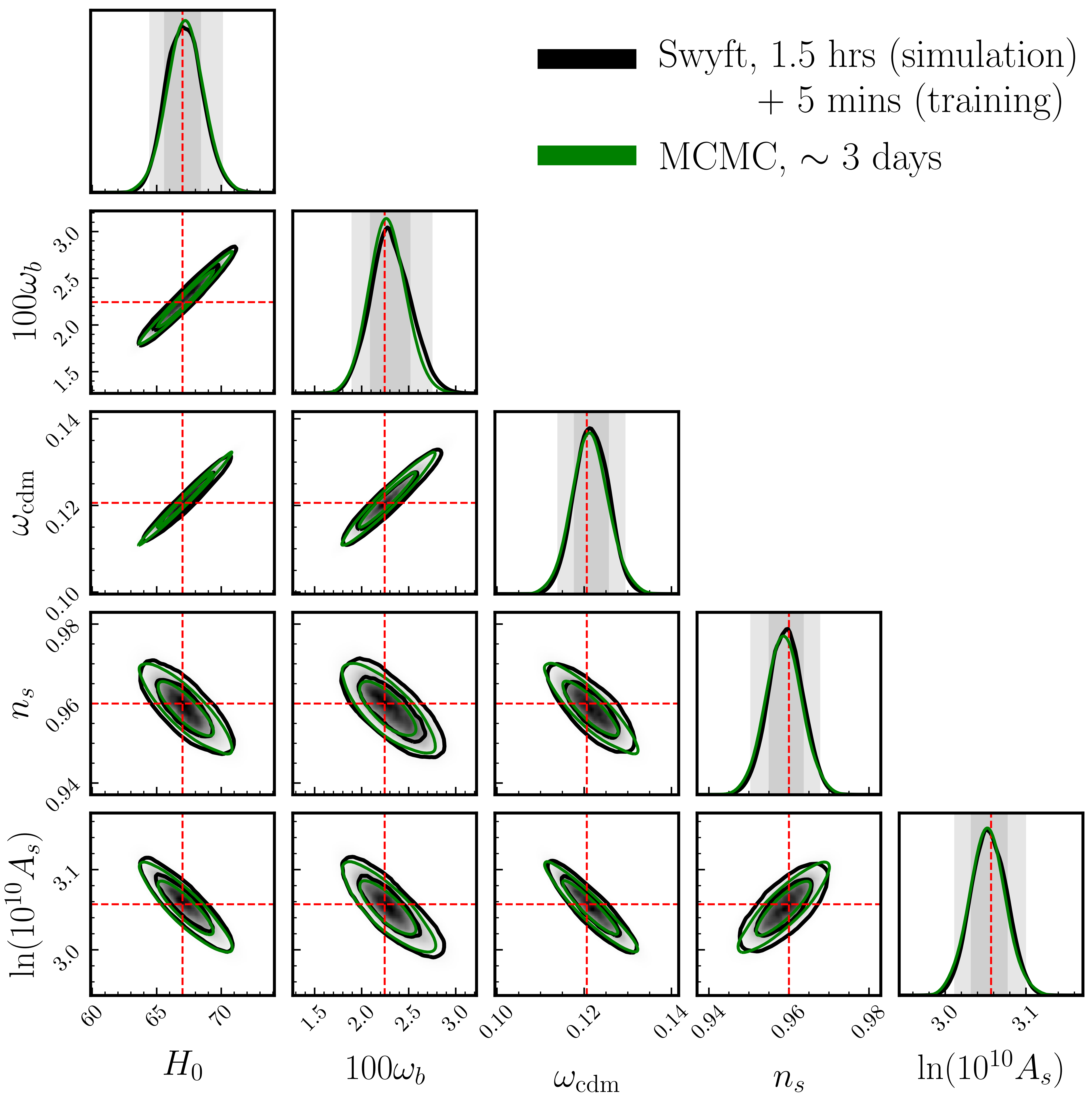
We generate 50k realisations of 3x2pt spectra with gaussian noise

$$\hat{C}_{ij}^{AB}(\ell) = C_{ij}^{AB}(\ell) + n_{ij}^{AB}(\ell) \quad \leftarrow \mathcal{N}(0,\mathbf{C})$$

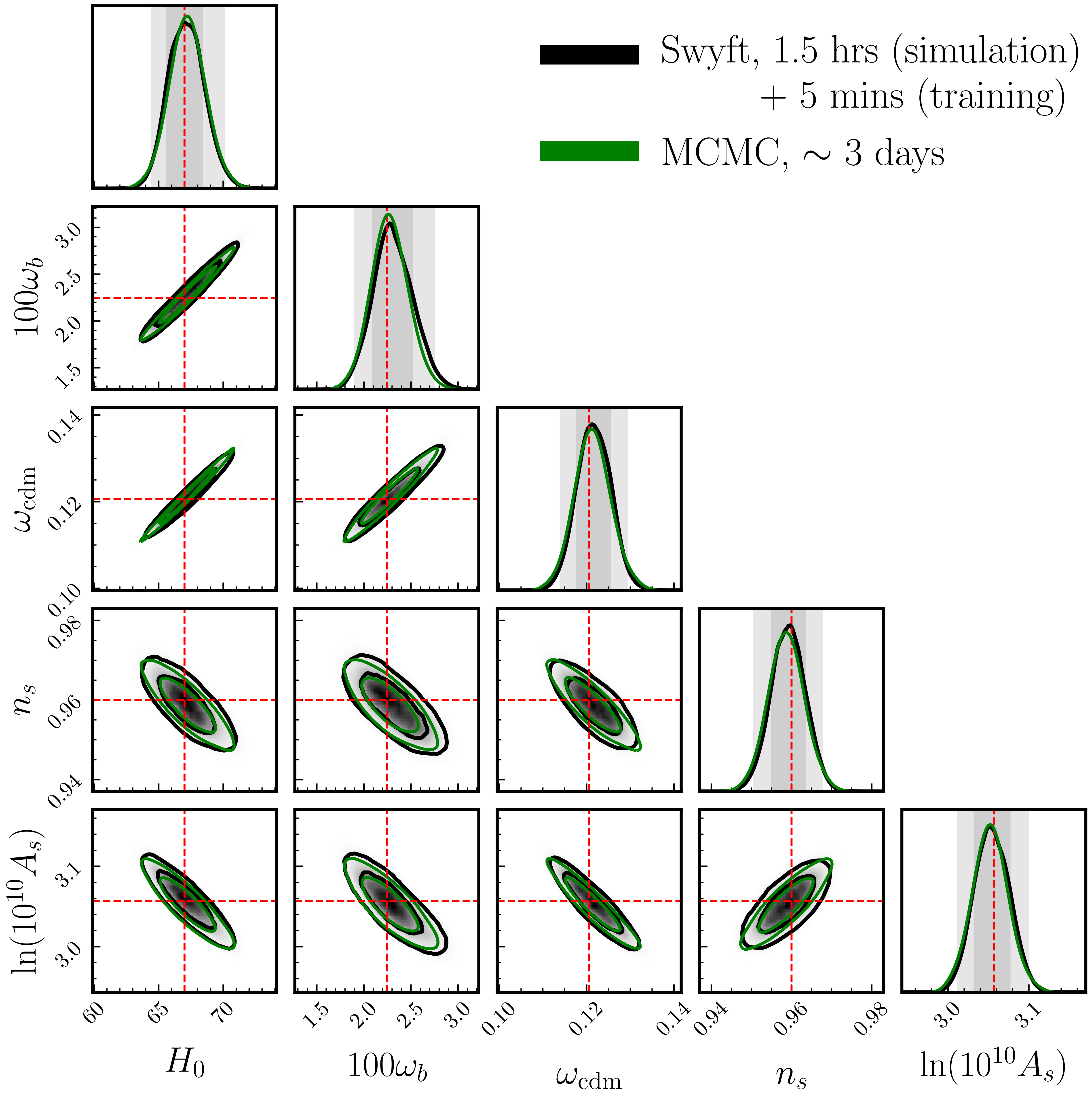
2. Network: We pre-compress spectra into features



Forecast ΛCDM posteriors

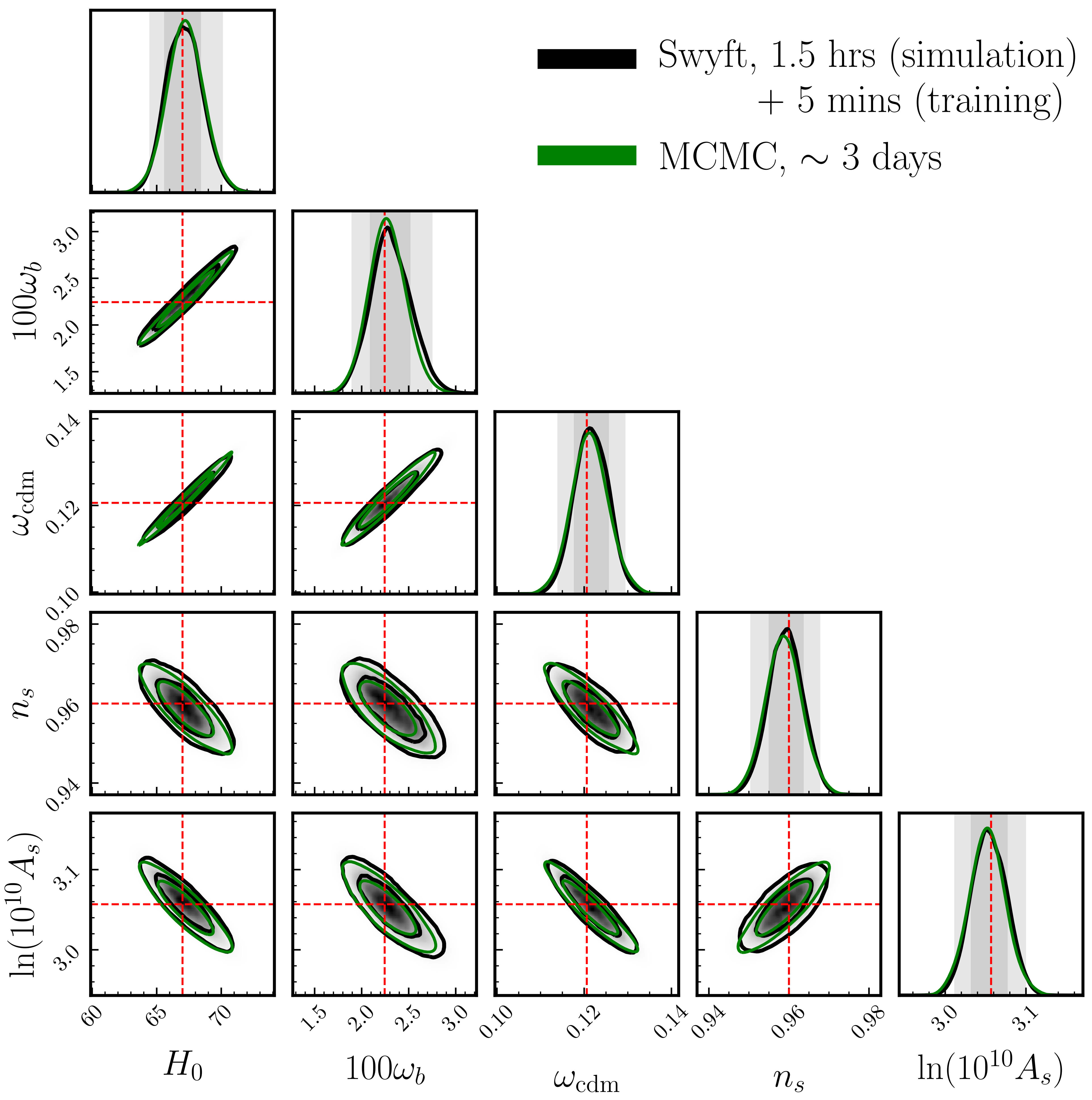


Forecast ΛCDM posteriors



MNRE & MCMC are in excellent agreement!

Forecast ΛCDM posteriors



Dramatic reduction in CPU time!

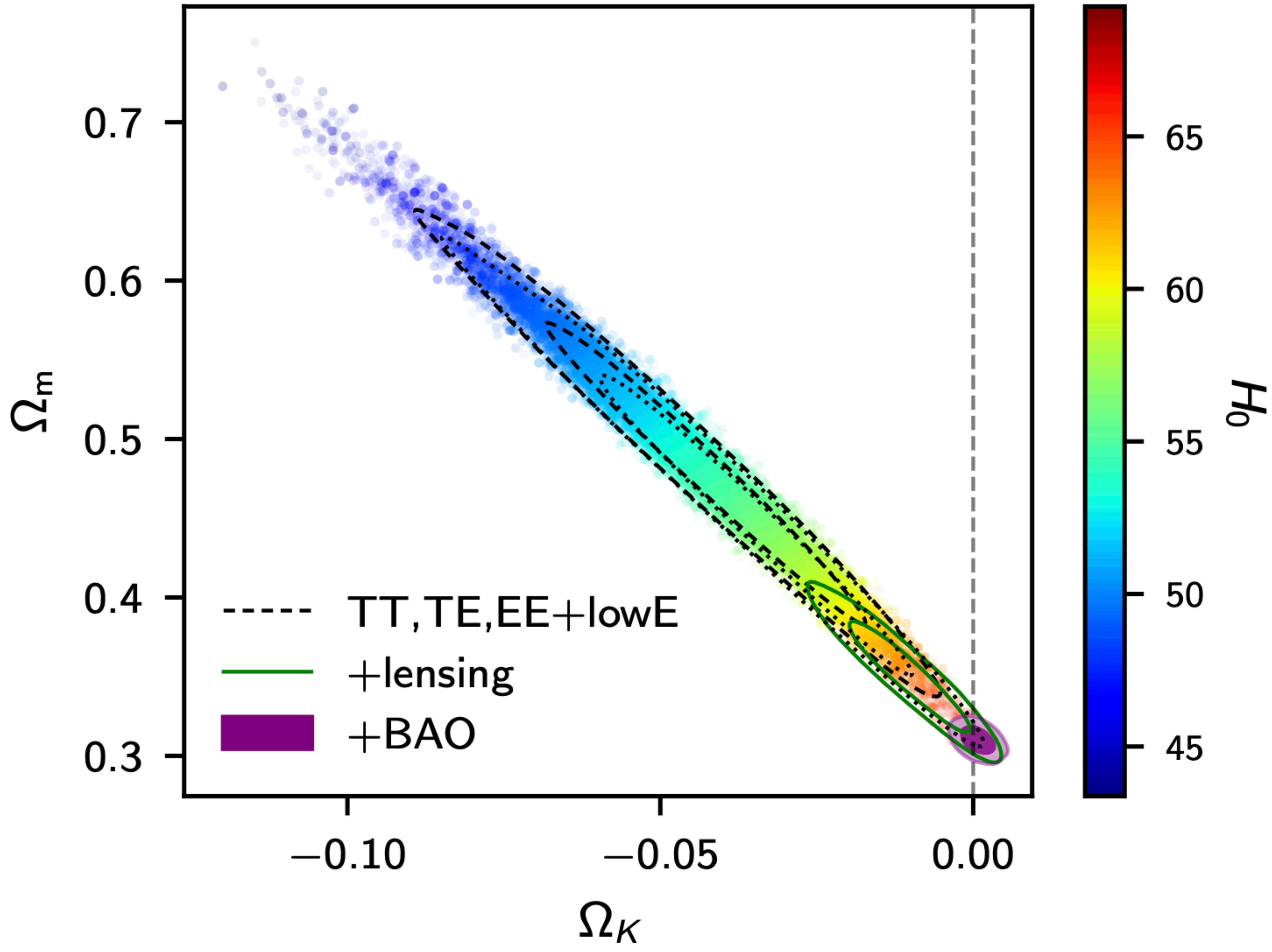
MNRE & MCMC are in excellent agreement!

What about combining different cosmological probes?

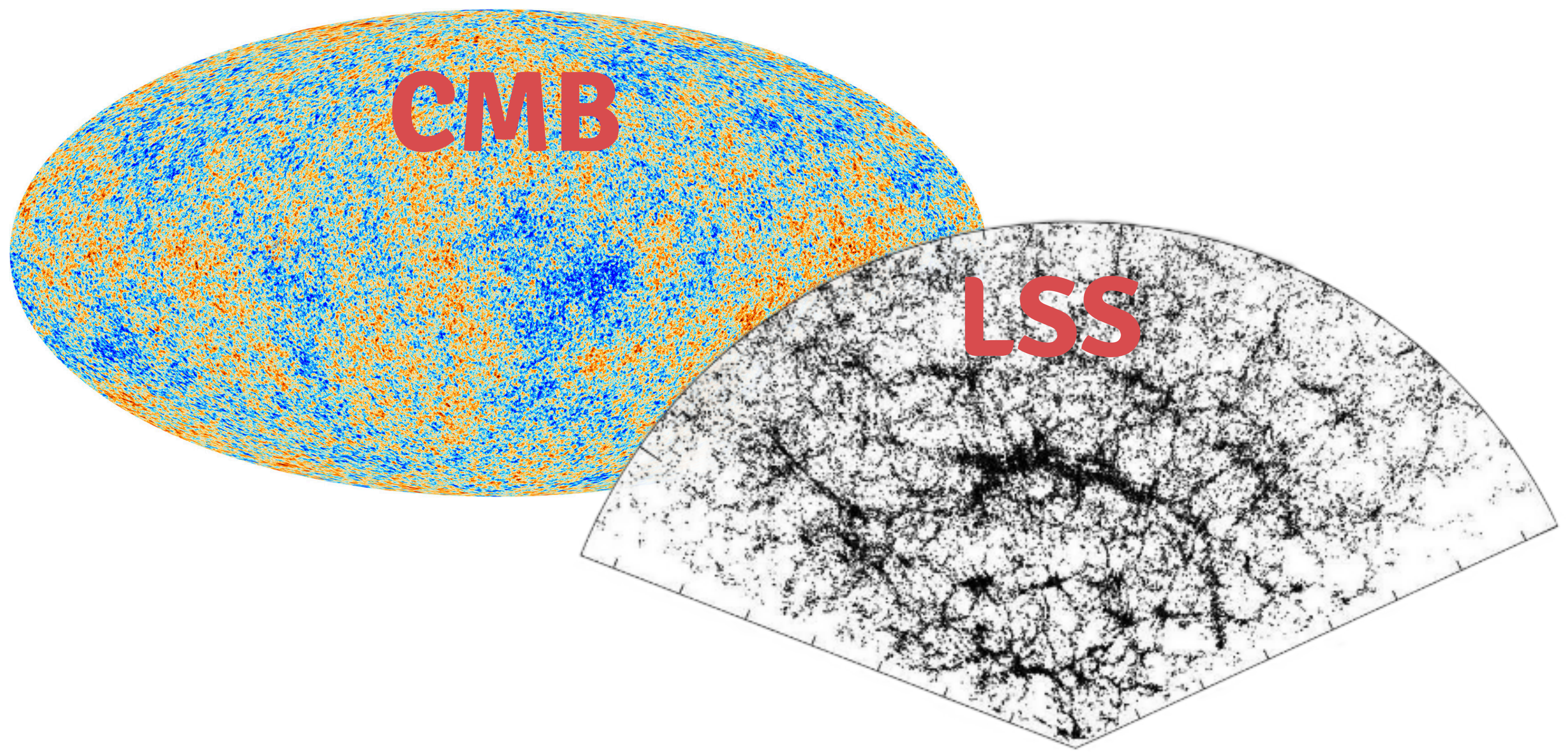
What about combining different cosmological probes?

Canonical example:

Planck 2018 Results. VI. [1807.06209]

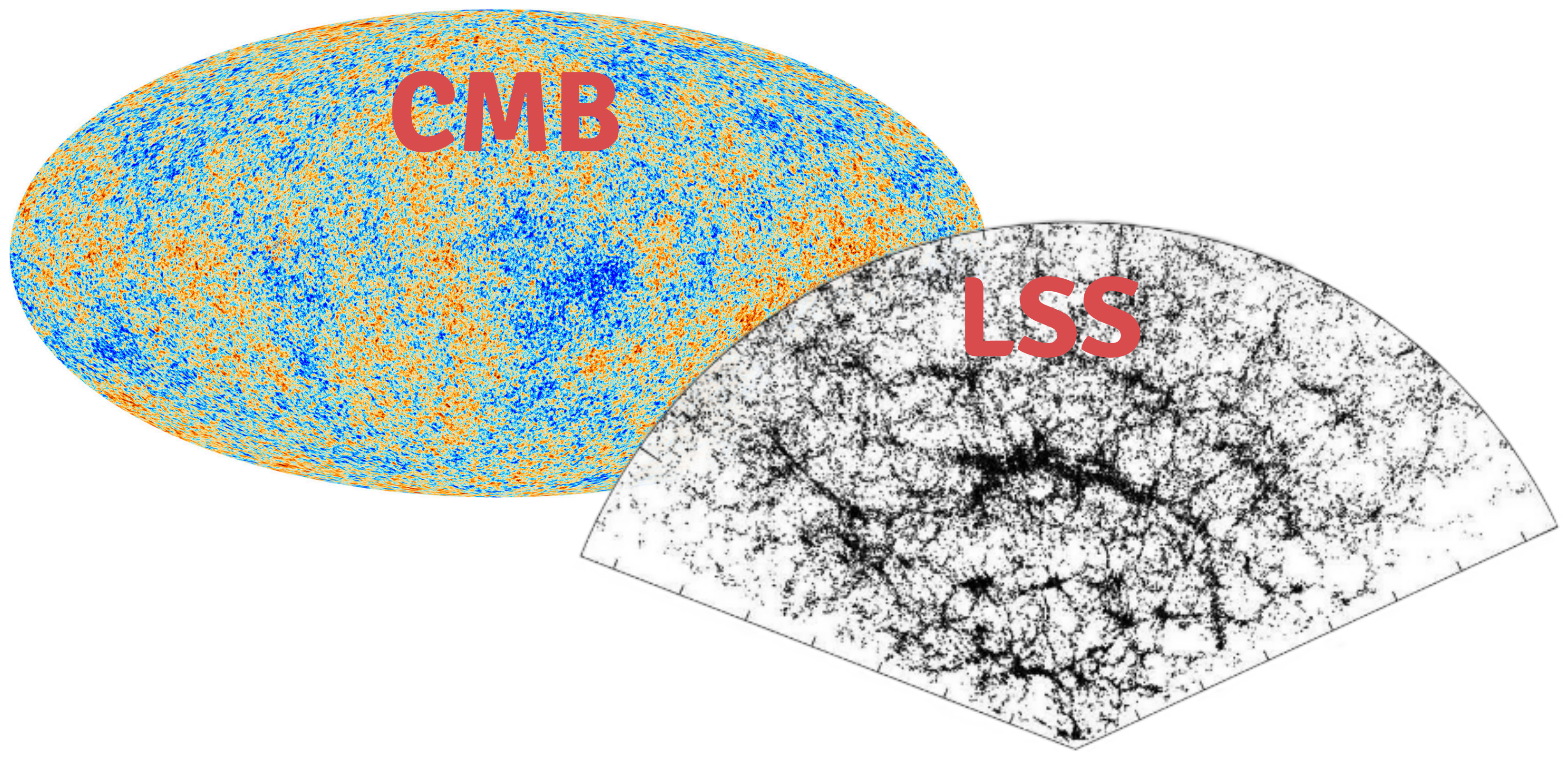


Joint analyses of CMB & LSS



Very complementary
(high-z vs. low-z, linear vs. non-linear)

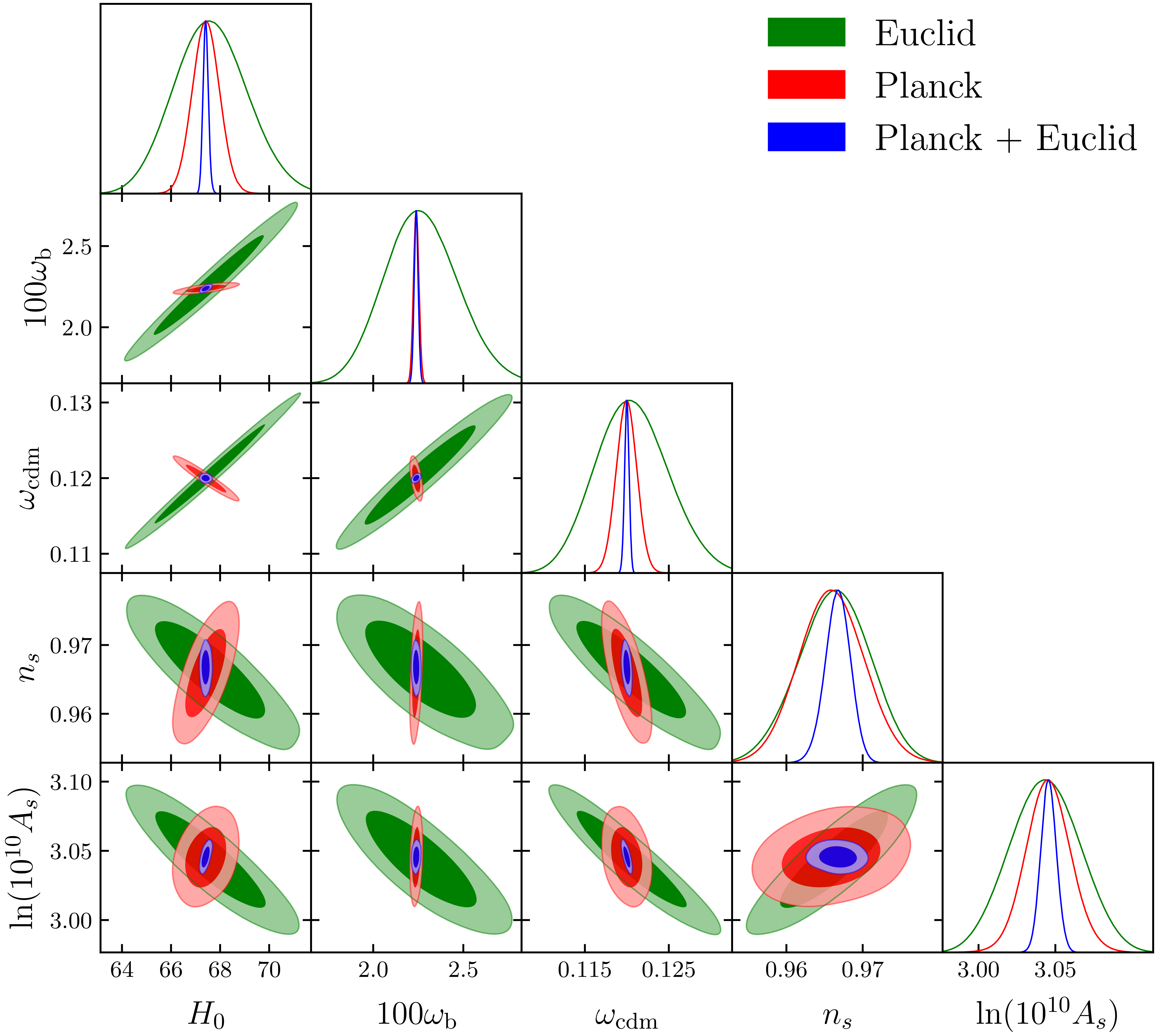
Joint analyses of CMB & LSS



Very complementary
(high-z vs. low-z, linear vs. non-linear)

↓

Break parameter degeneracies



Can we do CMB+LSS analyses in SBI?

Can we do CMB+LSS analyses in SBI?

Unfortunately, building a realistic Planck simulator seems very challenging

Can we do CMB+LSS analyses in SBI?

Unfortunately, building a realistic Planck simulator seems very challenging

BUT we can easily build an "effective" simulator using posterior samples from a previous MCMC run

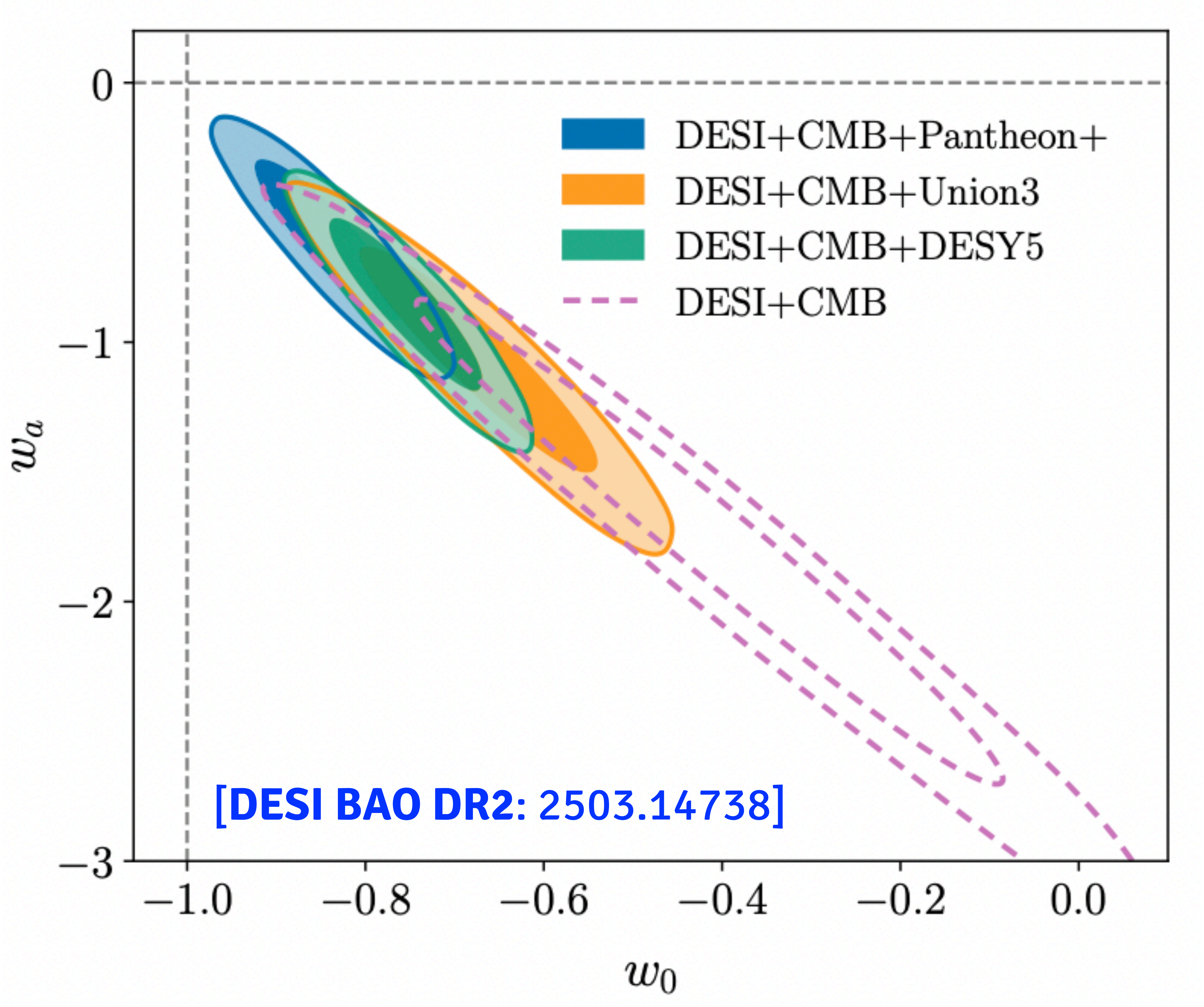
$$\boldsymbol{a} = \boldsymbol{\theta} - \boldsymbol{\theta}'$$

prior samples ↗ ↖ posterior samples

$$f_\phi(\boldsymbol{\theta} \,|\, \boldsymbol{a} = 0) \simeq p(\boldsymbol{\theta} \,|\, \mathbf{x_0})$$

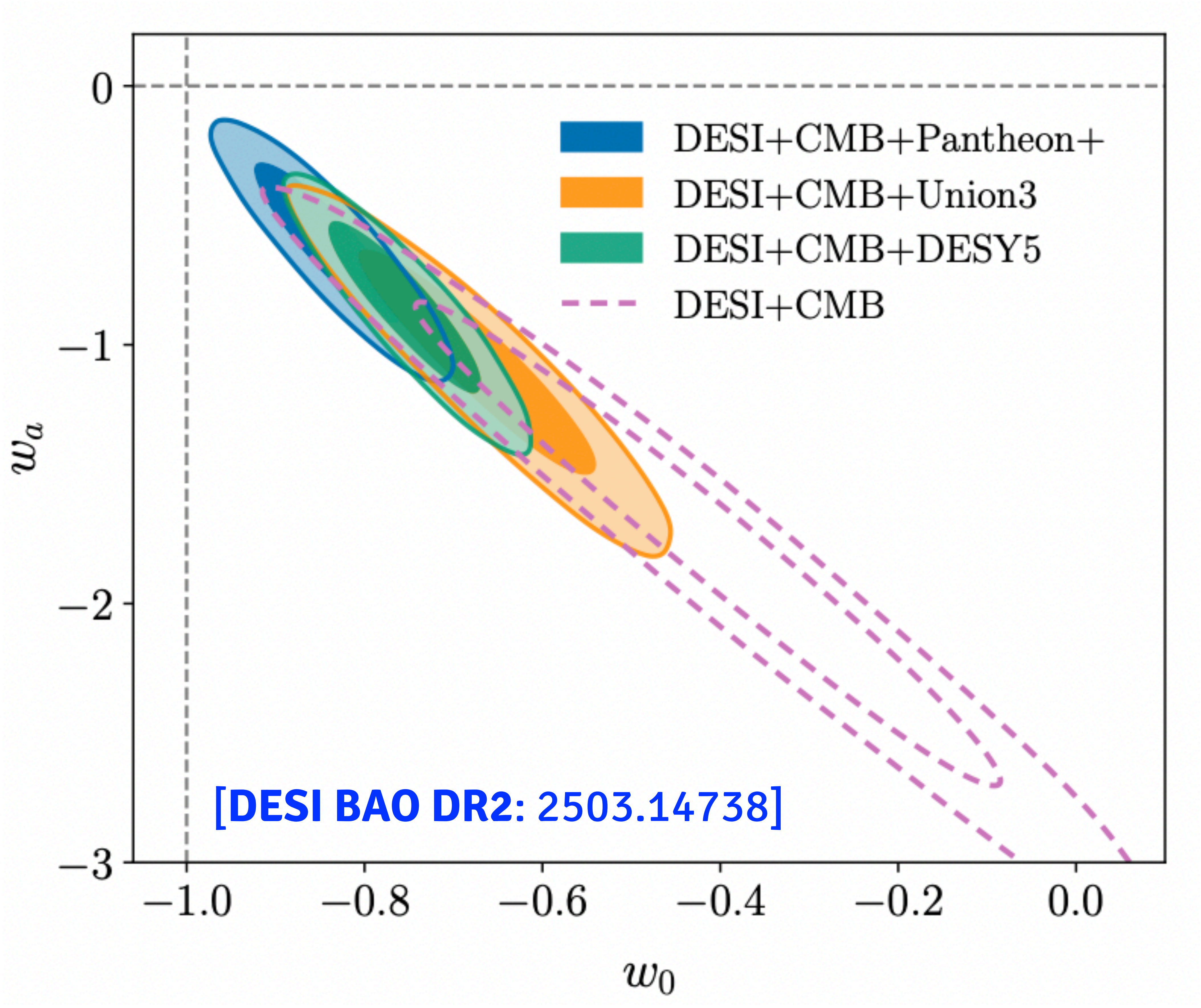
Application to evolving dark energy

$$w_{\rm CPL}(a) = w_0 + w_a(1 - a)$$



Application to evolving dark energy

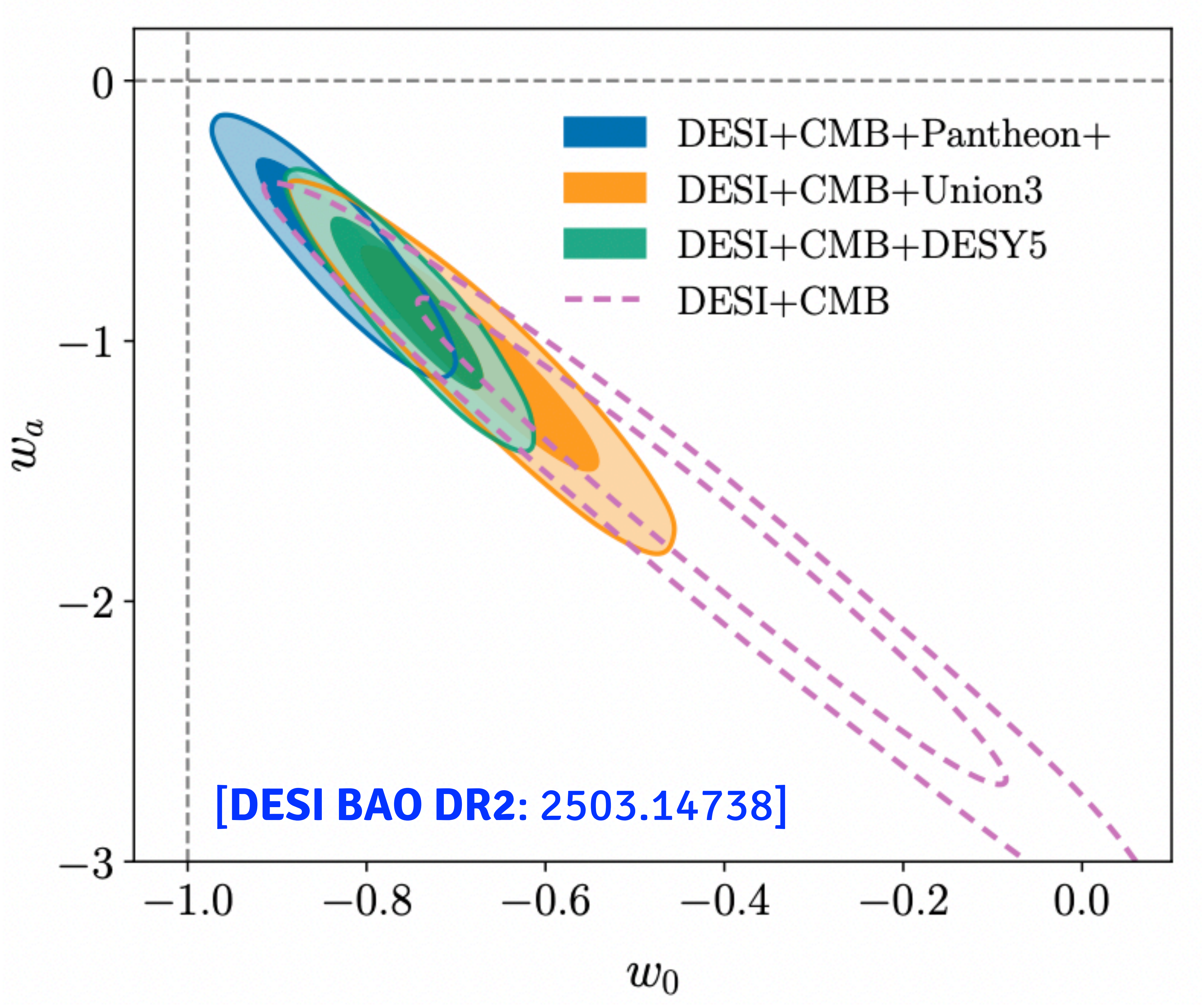
$$w_{\text{CPL}}(a) = w_0 + w_a(1 - a)$$



What can Euclid data (and its combination with current probes) tell us about this?

Application to evolving dark energy

$$w_{\rm CPL}(a) = w_0 + w_a(1 - a)$$



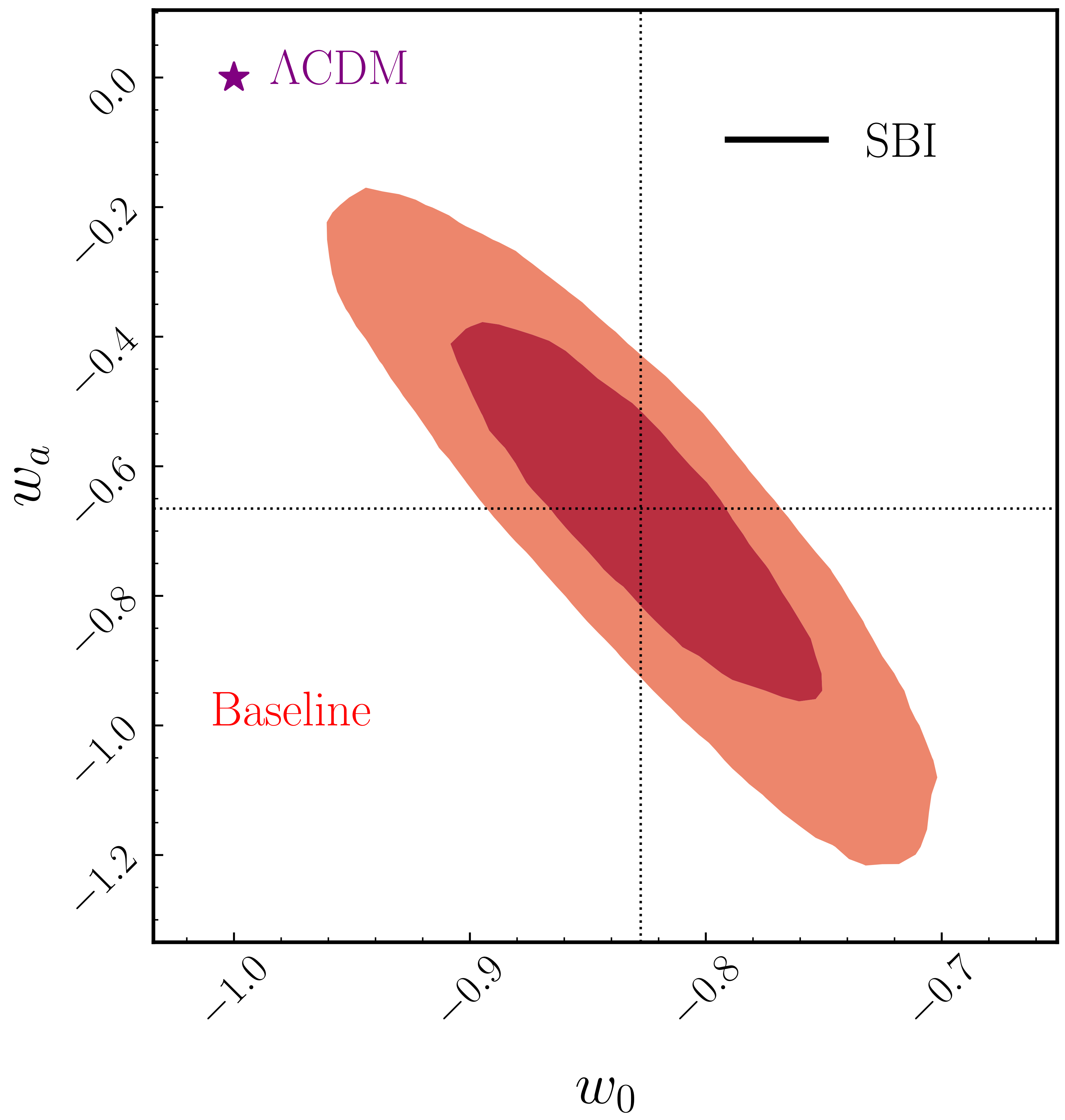
What can Euclid data (and its combination with current probes) tell us about this?

↓

Perform forecast assuming the bestfit w_0w_aCDM model hinted by DESI + CMB + Pantheon+

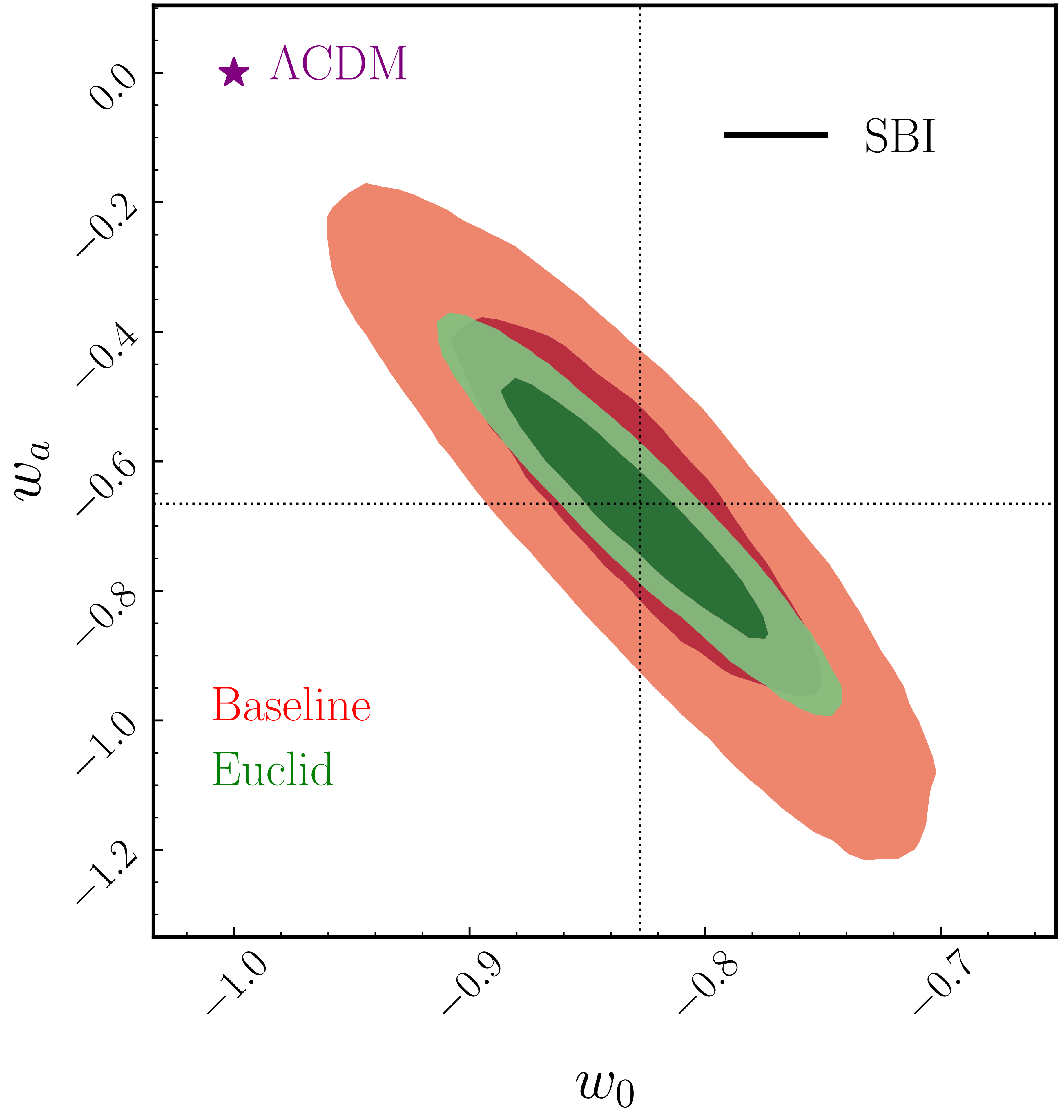
Forecast w_0w_aCDM posteriors

Baseline = DESI + Planck + Pantheon+



Forecast w_0w_aCDM posteriors

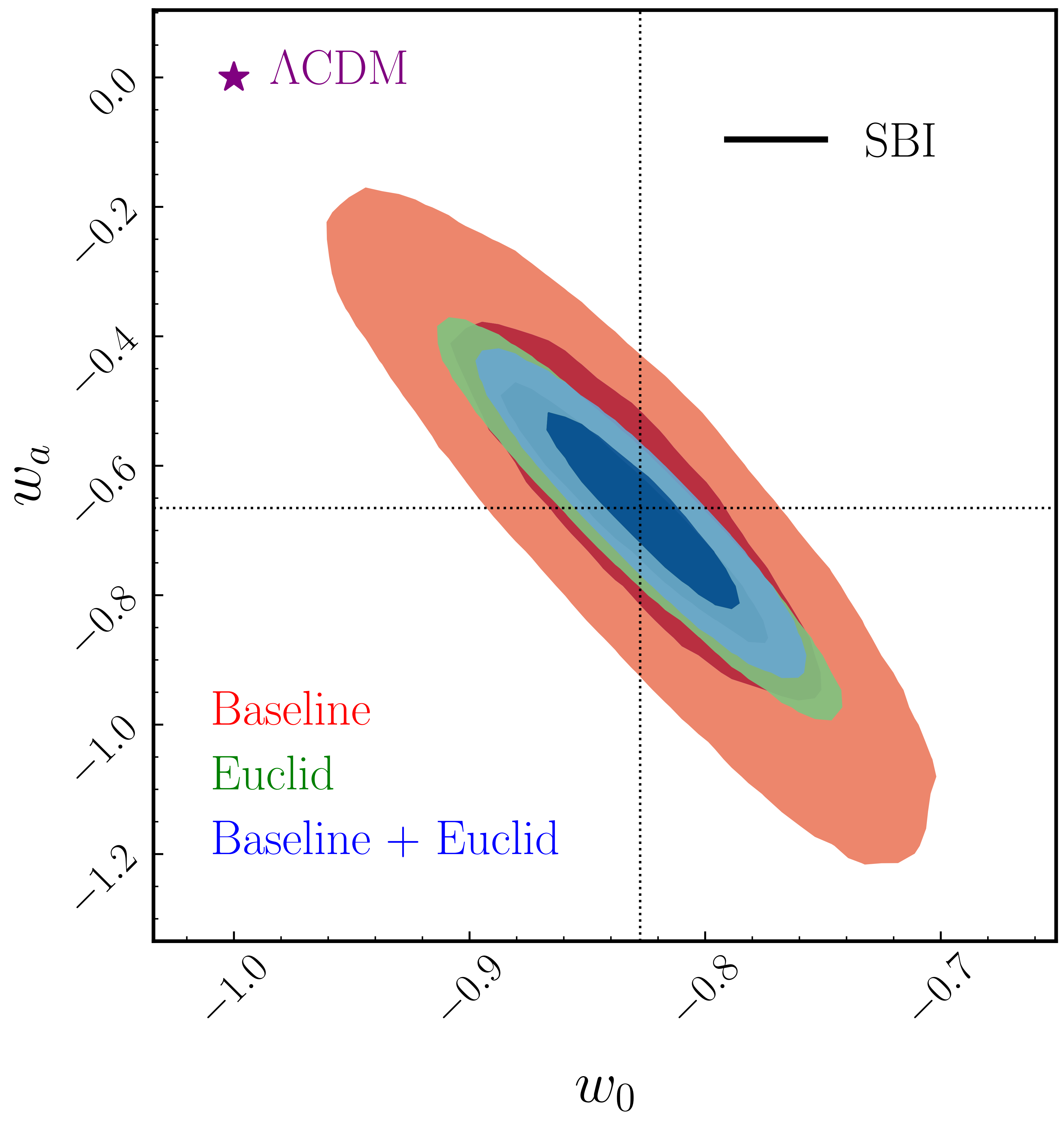
Baseline = DESI + Planck + Pantheon+



Euclid alone could detect the fiducial w_0w_aCDM model at the **$\sim 5\sigma$ level**

Forecast w_0w_aCDM posteriors

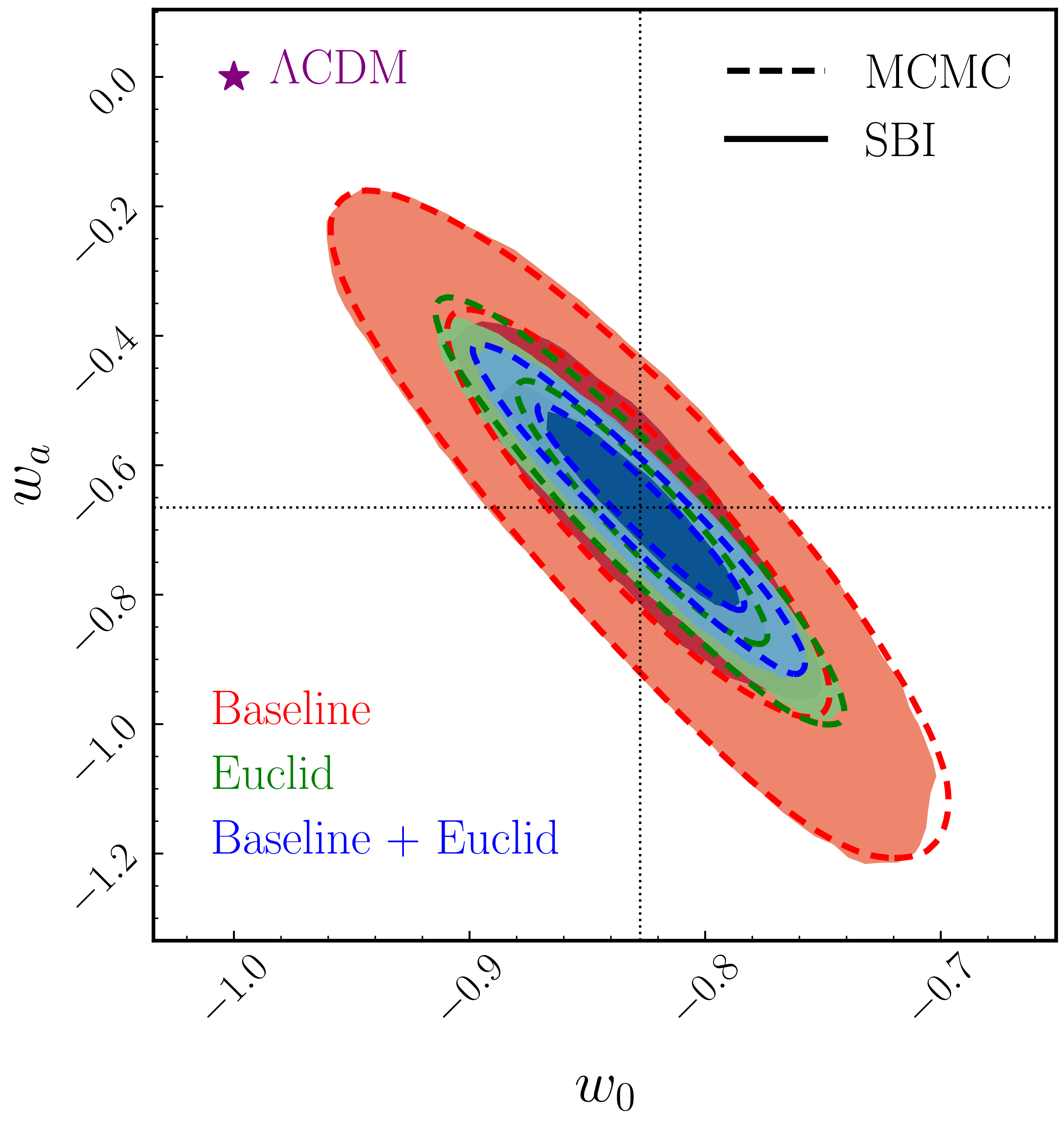
Baseline = DESI + Planck + Pantheon+



- **Euclid alone** could detect the fiducial w_0w_aCDM model at the **$\sim 5\sigma$ level**
- The **combination with Planck+DESI** data would rise the detection to $\sim 7\sigma$

Forecast $w_0 w_a$CDM posteriors

Baseline = DESI + Planck + Pantheon+



SBI is in good agreement with MCMC, while requiring only ~3% of the number of model evaluations used by MCMC

Conclusions

- Modern deep learning techniques will be key to learn as much as we can about the dark sector and neutrinos from future data

- Emulators achieve ultra-fast model evaluations → applied to decaying neutrino models, these can relax current mass bounds up to $\sum m_\nu < 0.24$ eV

- SBI methods provide a simulator-efficient and scalable alternative to MCMC → applied to Euclid+Planck, these datasets could detect evolving dark energy at $\sim 7\sigma$

Conclusions

- Modern deep learning techniques will be key to learn as much as we can about the dark sector and neutrinos from future data

- Emulators achieve ultra-fast model evaluations → applied to decaying neutrino models, these can relax current mass bounds up to $\sum m_\nu < 0.24$ eV

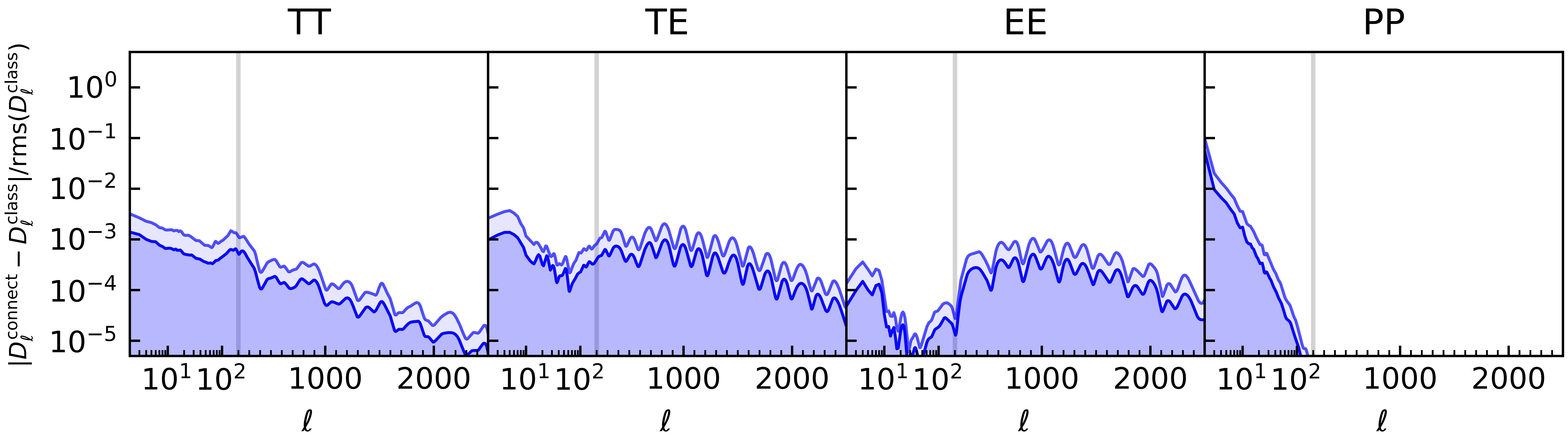
- SBI methods provide a simulator-efficient and scalable alternative to MCMC → applied to Euclid+Planck, these datasets could detect evolving dark energy at $\sim 7\sigma$

THANK YOU!

g.francoabellan@uva.nl

BACK-UP

Errors in emulated CMB spectra for neutrino decays



The 1σ and 2σ percentile errors in the CMB spectra (from the test set) are almost always below 1%

Strategy: train a neural network $d_\phi(\mathbf{x}, \boldsymbol{\theta}) \in [0,1]$ as a binary classifier, so that

- $d_\phi(\mathbf{x}, \boldsymbol{\theta}) \simeq 1$ if $(\mathbf{x}, \boldsymbol{\theta}) \sim p(\mathbf{x}, \boldsymbol{\theta}) = p(\mathbf{x} \,|\, \boldsymbol{\theta}) p(\boldsymbol{\theta})$

- $d_\phi(\mathbf{x}, \boldsymbol{\theta}) \simeq 0$ if $(\mathbf{x}, \boldsymbol{\theta}) \sim p(\mathbf{x}) p(\boldsymbol{\theta})$

Note: Φ denotes all the network parameters

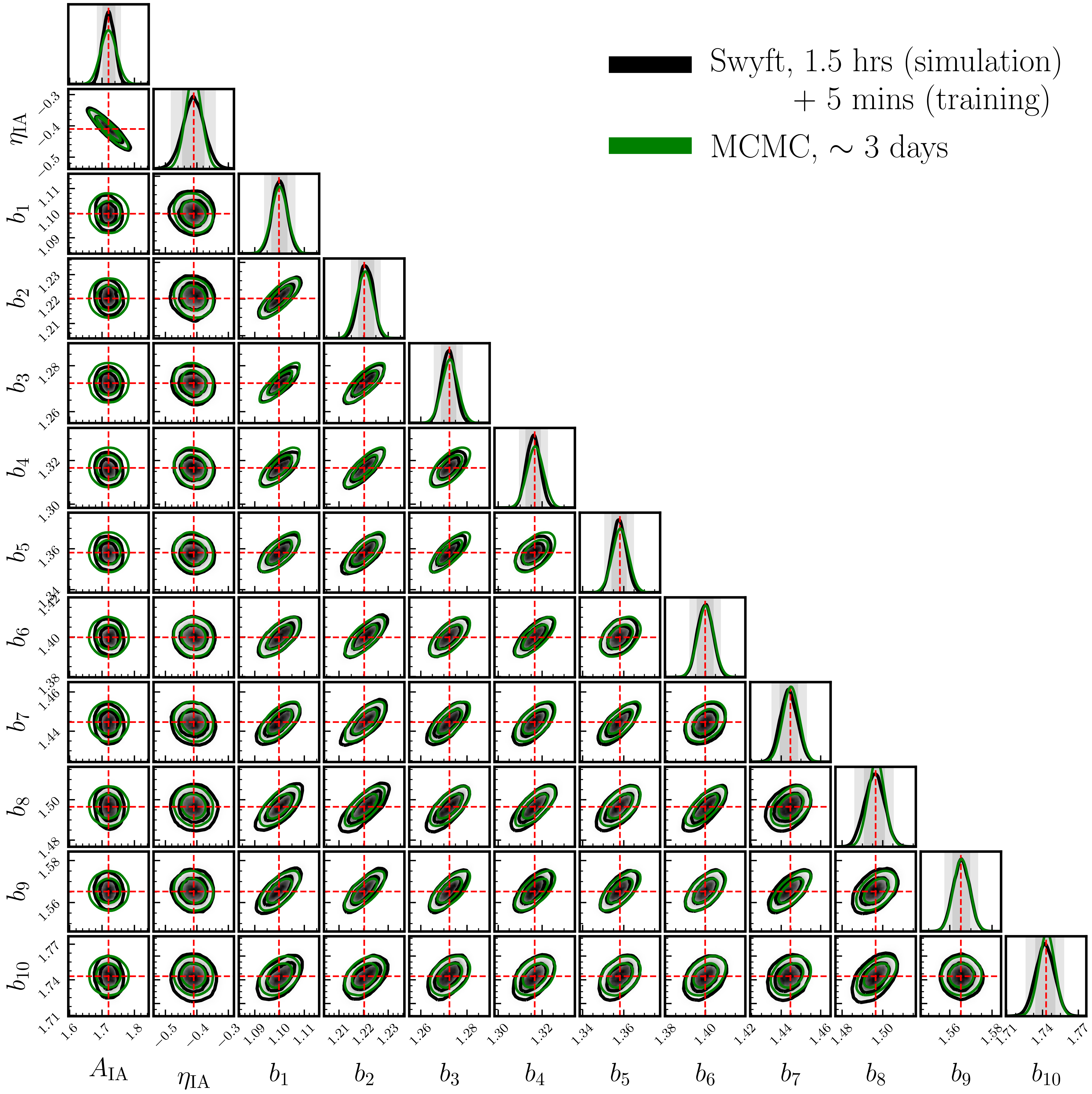
We have to minimise a **loss function** w.r.t. the network params. Φ

$$L[d_\phi(\mathbf{x},\boldsymbol{\theta})] = -\int d\mathbf{x}d\boldsymbol{\theta}\left[p(\mathbf{x},\boldsymbol{\theta})\ln(d_\phi(\mathbf{x},\boldsymbol{\theta})) + p(\mathbf{x})p(\boldsymbol{\theta})\ln(1-d_\phi(\mathbf{x},\boldsymbol{\theta}))\right]$$

which yields

$$d_\phi(\mathbf{x},\boldsymbol{\theta}) \simeq \frac{p(\mathbf{x},\boldsymbol{\theta})}{p(\mathbf{x},\boldsymbol{\theta}) + p(\mathbf{x})p(\boldsymbol{\theta})} = \frac{r(\mathbf{x};\boldsymbol{\theta})}{r(\mathbf{x};\boldsymbol{\theta})+1}$$

Posteriors for 3x2pt nuisance parameters



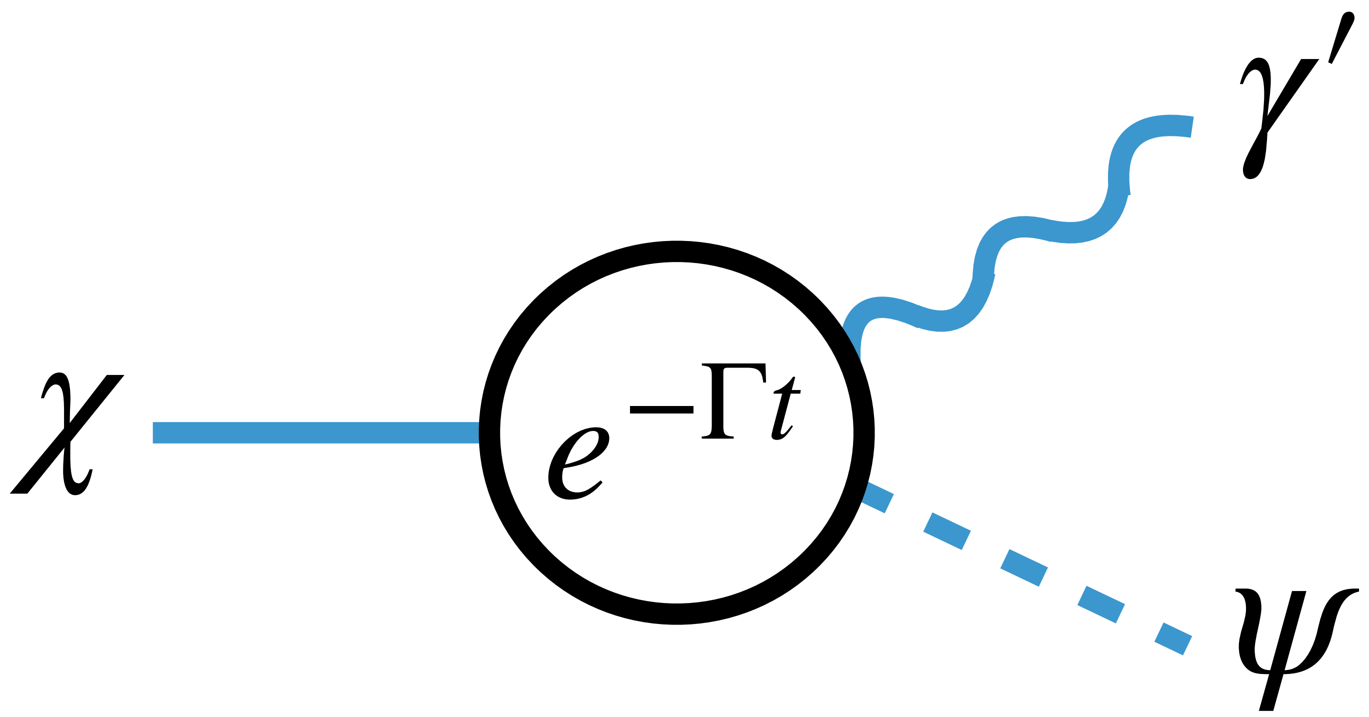
MNRE & MCMC are again in good agreement

- Does MNRE perform well with highly non-Gaussian posteriors?

- As an example, we test a model of **CDM decaying to DR + WDM** (proposed to explain the S_8 tension)

[Abellán et al. 2102.12498]

[Bucko et al. 2307.03222]

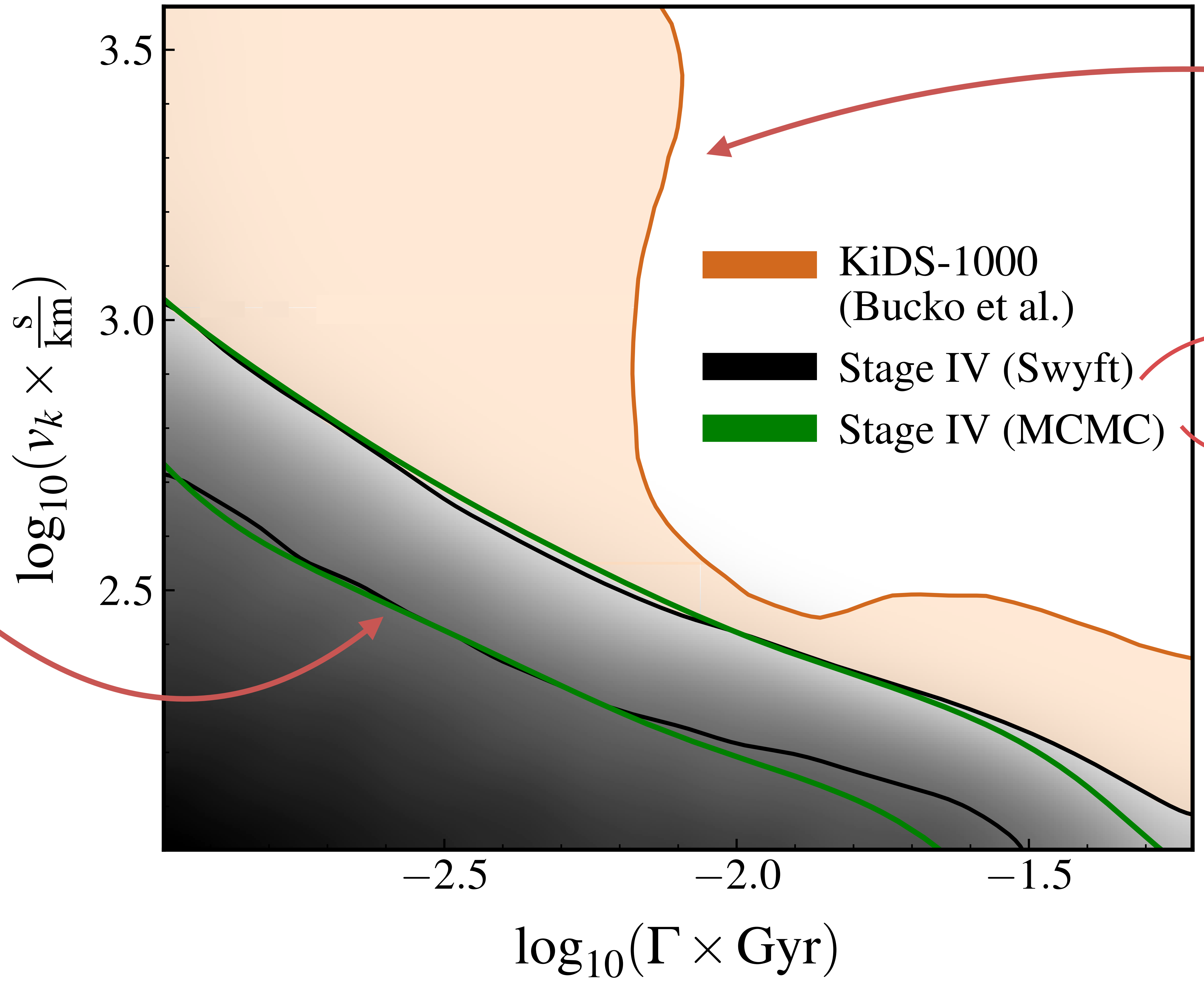


Decay rate Γ

WDM velocity kick v_k

Forecast constraints on decaying DM

MNRE & MCMC in good agreement



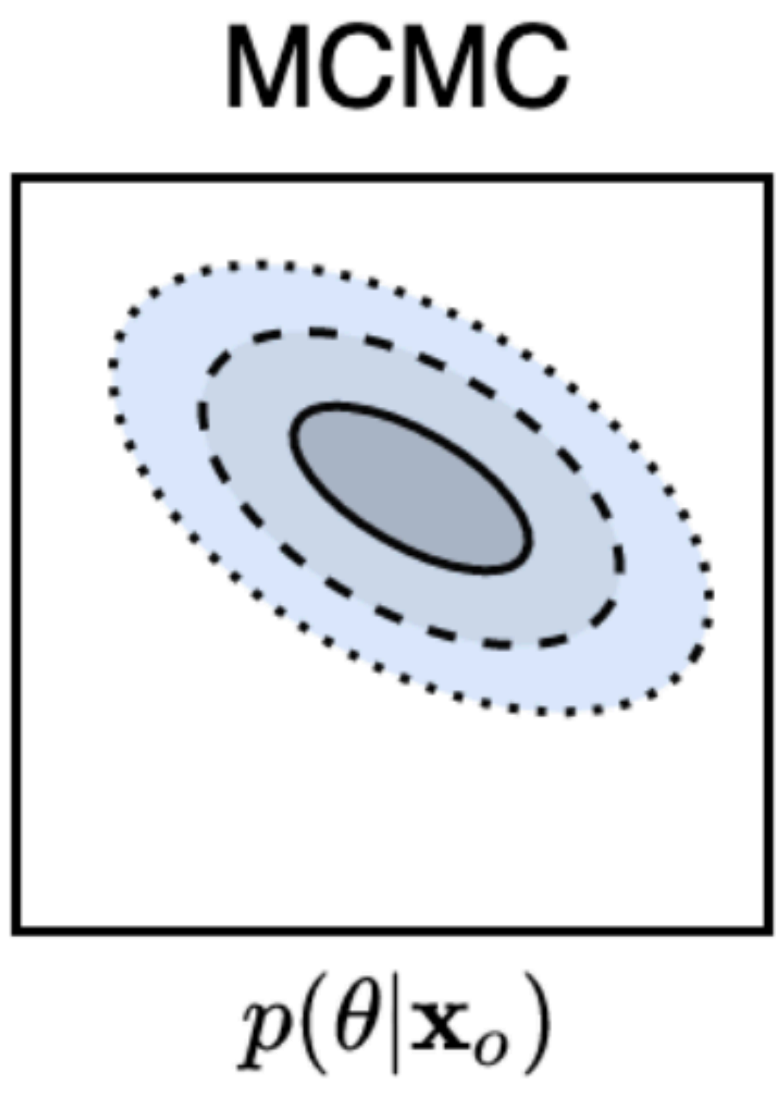
Improve current limits by ~1 order of magnitude!

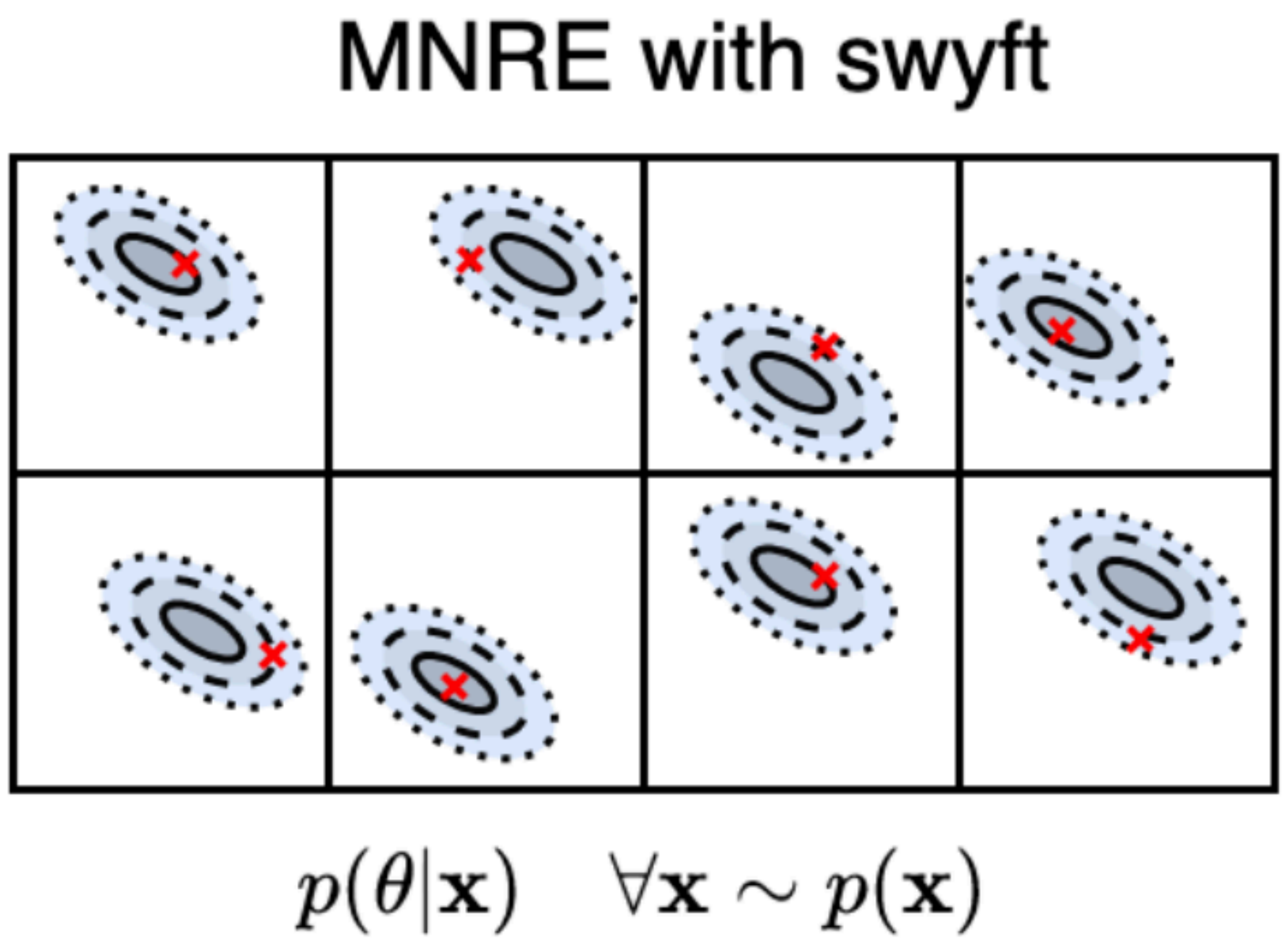
3 hours vs. 8 days!

But can we trust our results?

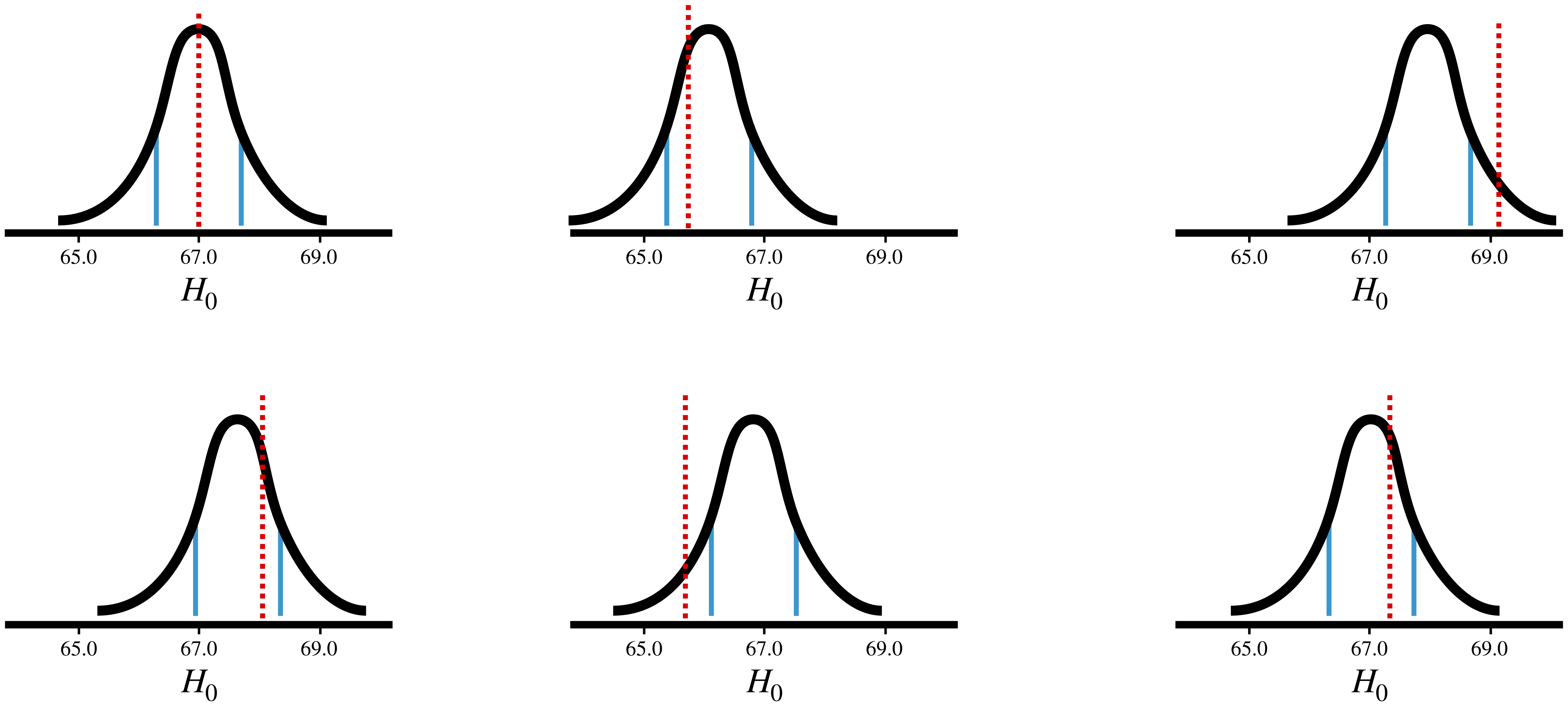
...even if NNs are often seen as "black boxes”, it is possible to perform statistical consistency tests which are impossible with MCMC

Trained networks can estimate effortlessly
the posteriors for all simulated observations



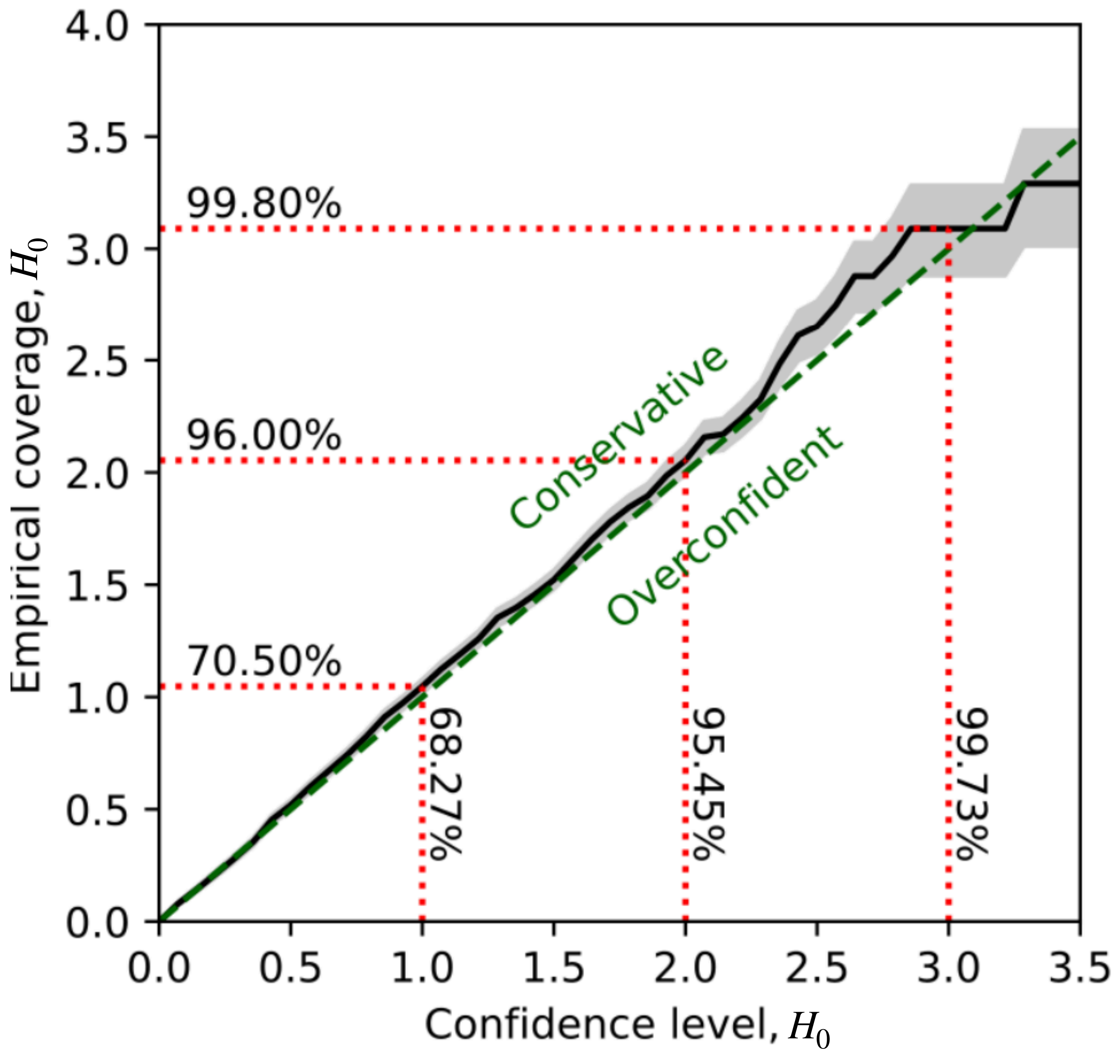


Ex: Is the estimated 68.27% interval covering the ground truth in ~68% of the cases?

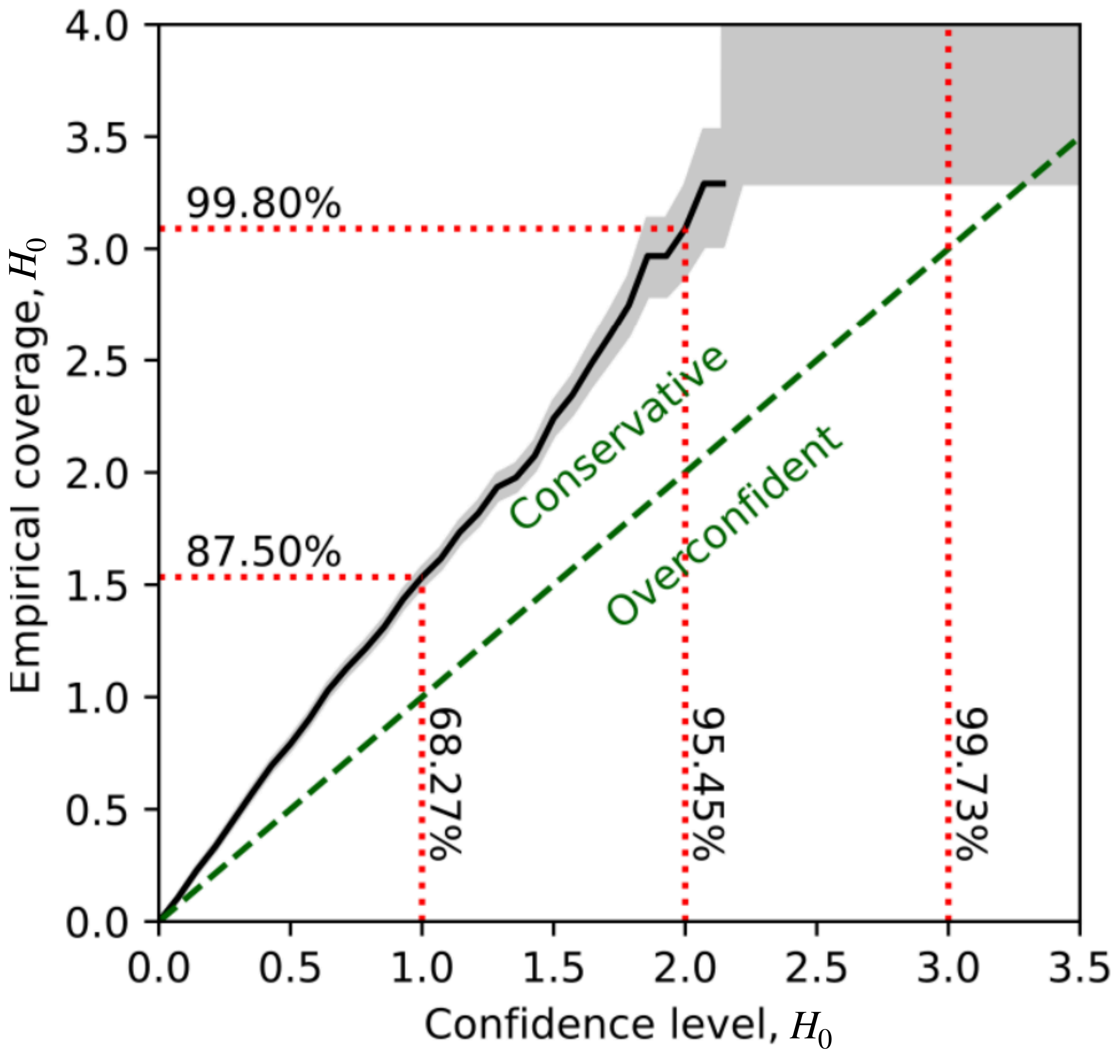


We can empirically estimate the Bayesian coverage

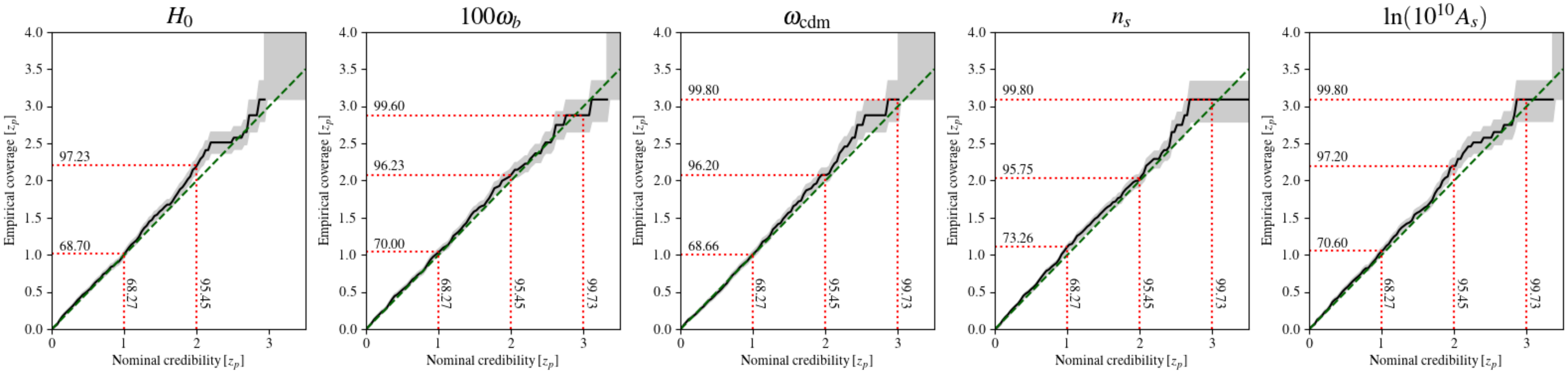
Converged network



Non-converged network



Coverage test for Stage-IV 3x2pt



GFA++ 2403.14750

Empirical coverage and confidence level match to excellent precision!