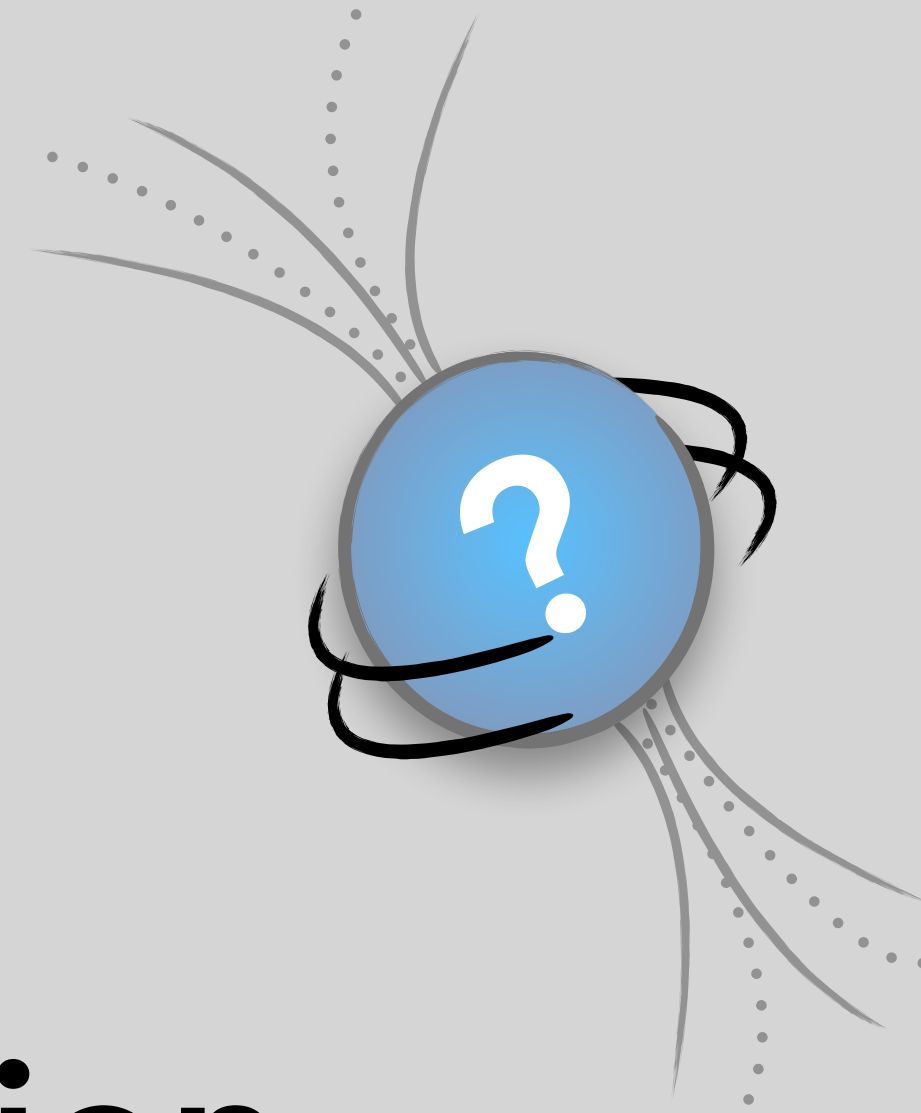


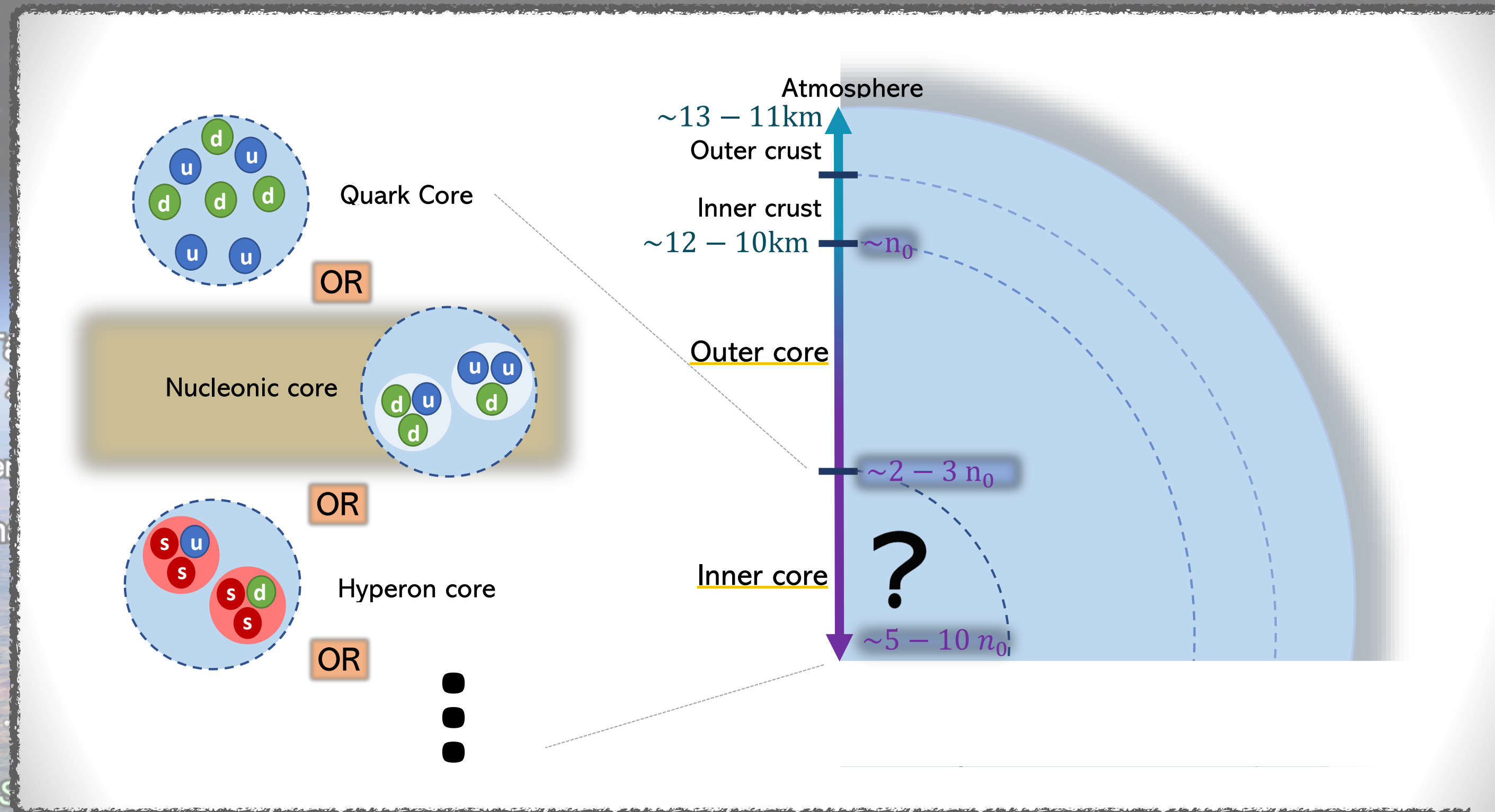


Can Neural Posterior Estimation Infer the Neutron Star Equation of State



Valéria Carvalho, Constança Providência, Márcio Ferreira, Michał Bejger

What's inside a neutron star?



$$M = 1 \sim 2 M_{\odot}$$

$$R \approx 10^{-5} R_{\odot}$$

$$T=0$$



The equation of state: A bridge between micro and macro

.....

Equation of State(EoS)

$$p(\varepsilon)$$

Speed of Sound

$$c_s^2(n) = \frac{dp(n)}{d\varepsilon(n)}$$

Trace Anomaly

$$\Delta(n) = \frac{1}{3} - \frac{p(n)}{\varepsilon(n)}$$

Tolman-Oppenheimer-Volkoff equations

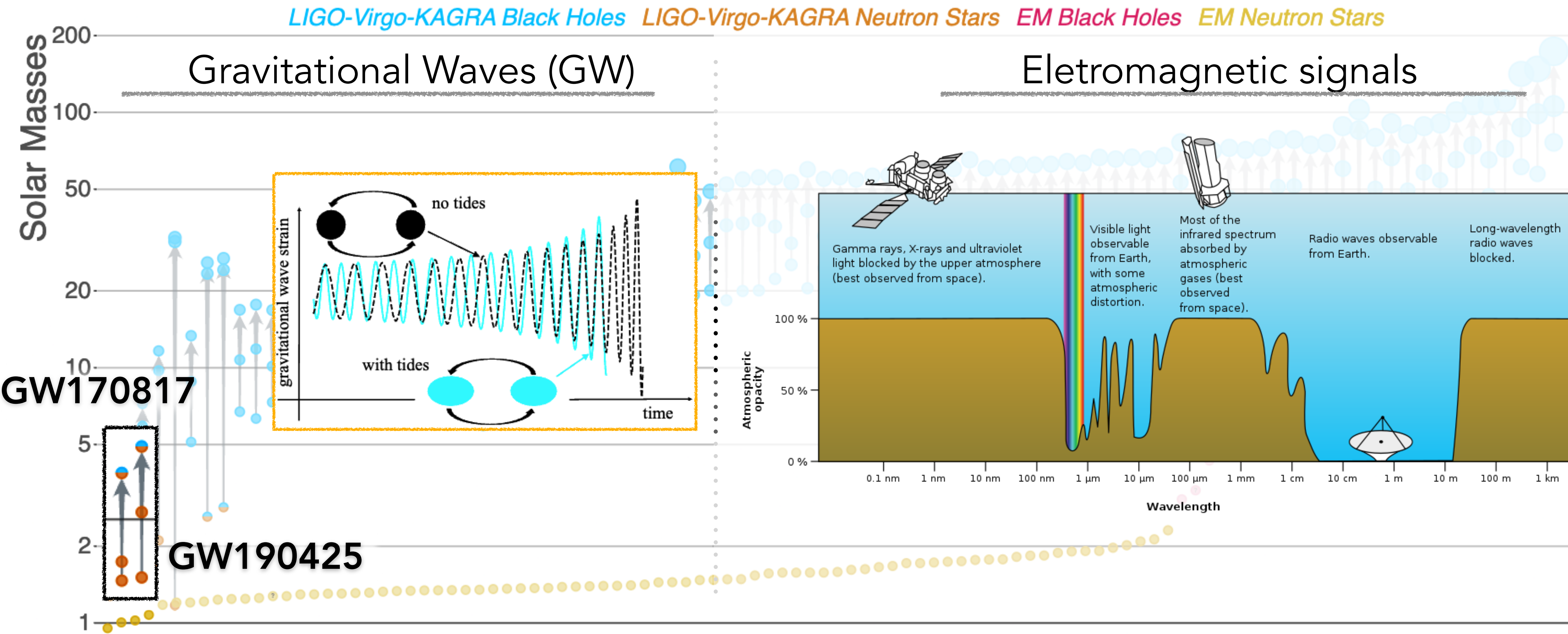
$$\frac{dP(r)}{dr} = -\frac{\varepsilon(r)m(r)}{r^2} \left(1 + \frac{P(r)}{\varepsilon(r)}\right) \left(1 + \frac{4\pi r^3 P(r)}{m(r)}\right) \left(1 - \frac{2m(r)}{r}\right)^{-1}$$

$$\frac{dm(r)}{dr} = 4\pi r^2 \varepsilon(r)$$

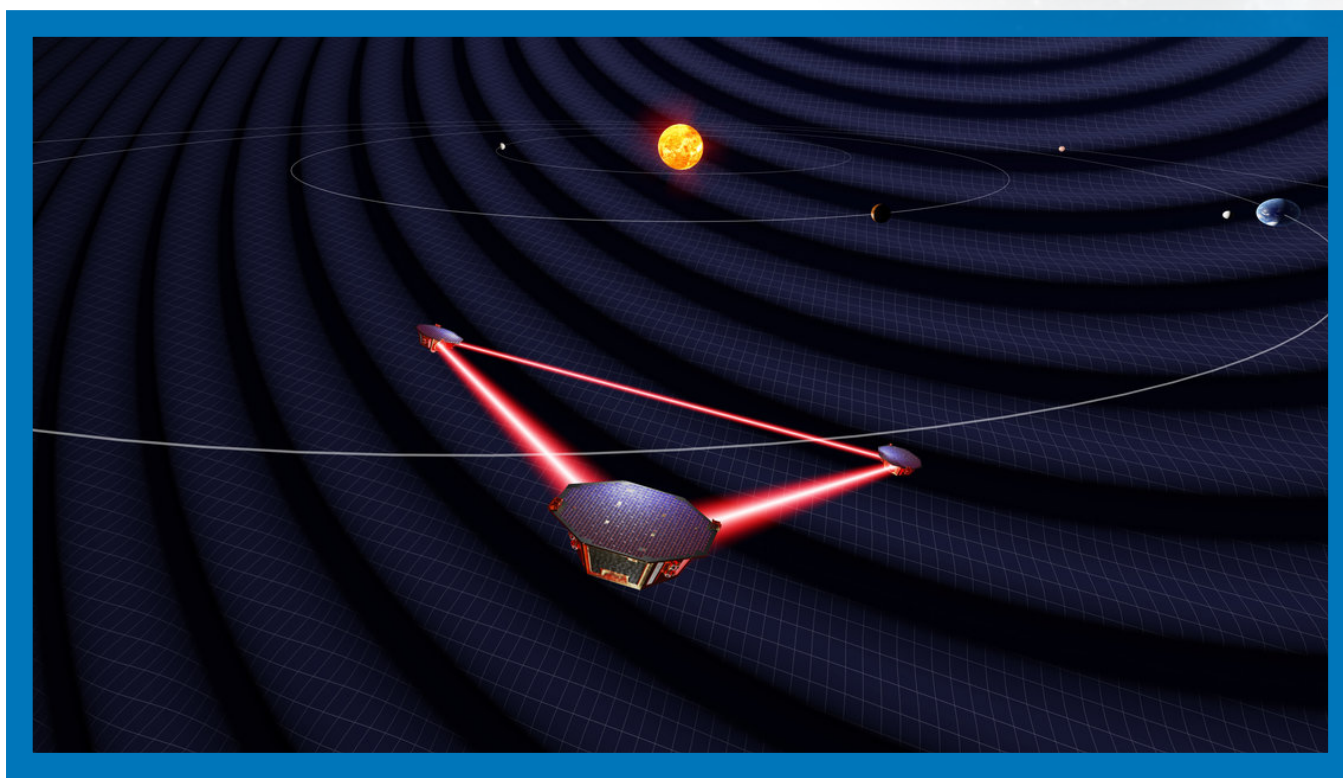
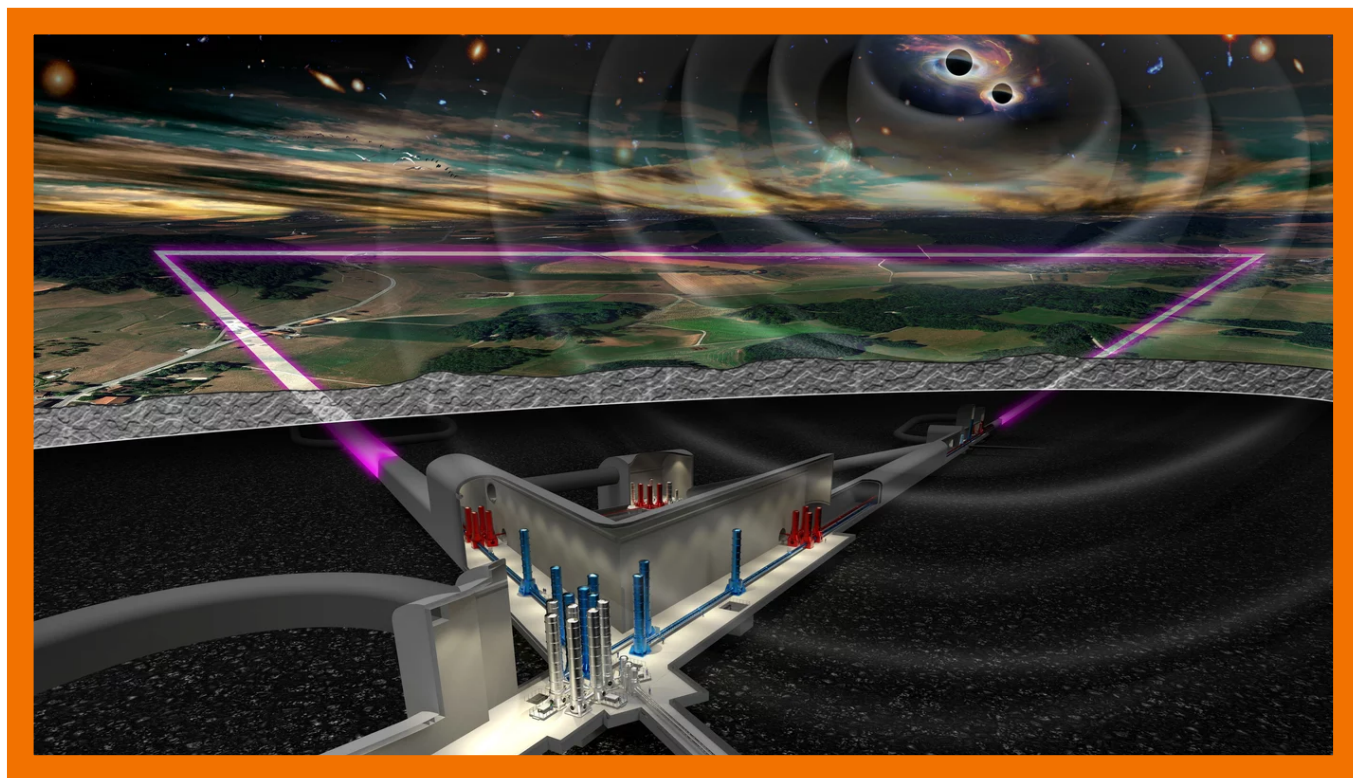
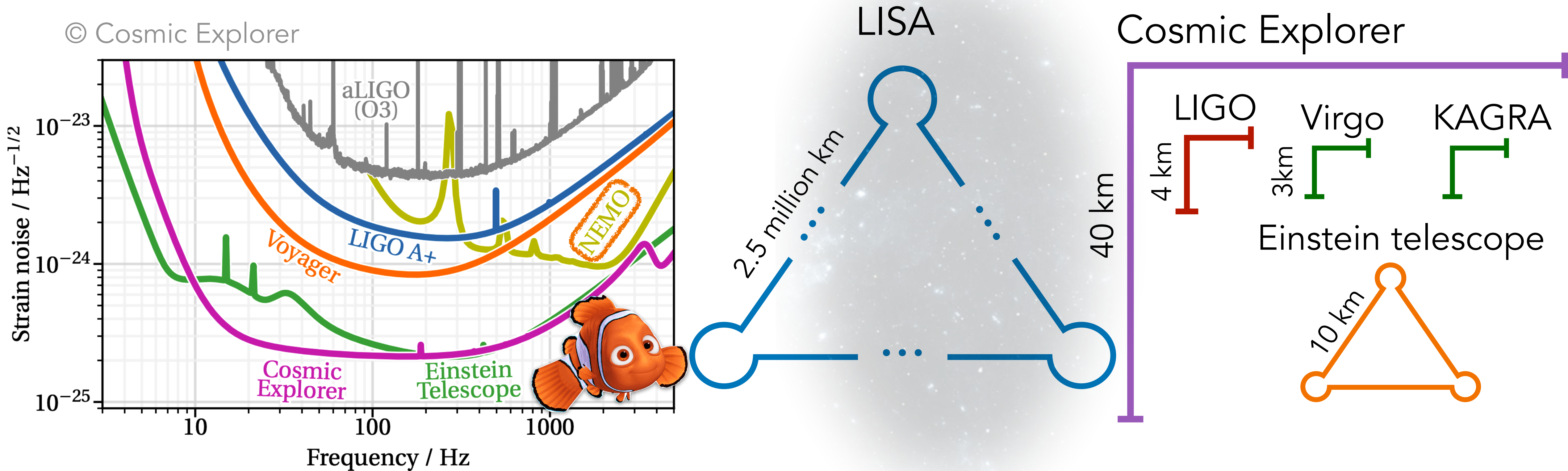
Tidal deformability

$$\Lambda = \frac{2}{3} k_2 C^{-5}, \quad C = \frac{M}{R}$$

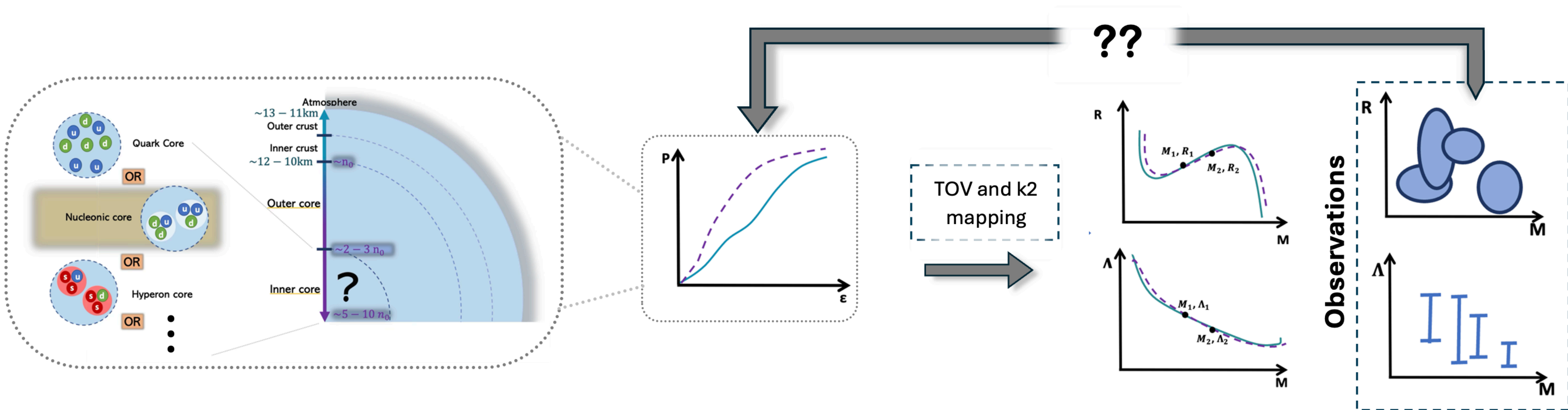
Current observations



A golden age of data is coming- GW future detectors



The challenge: A sparse and noisy inverse problem



For current observations:

- Sparse coverage of observables,
- Uncertainties and degeneracies.

For future observations:

- Big amount of data,
- Smaller uncertainties, less degeneracies.

Deep Learning pipeline

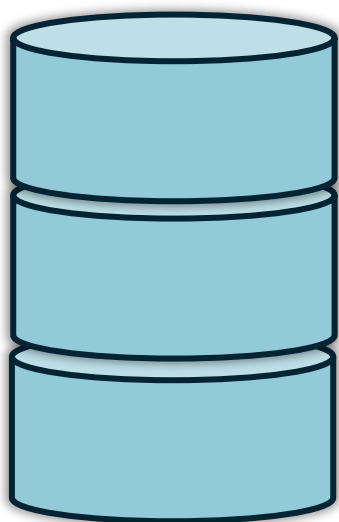
Deep Learning $\subset$ Machine Learning $\subset$ AI

Benefits :

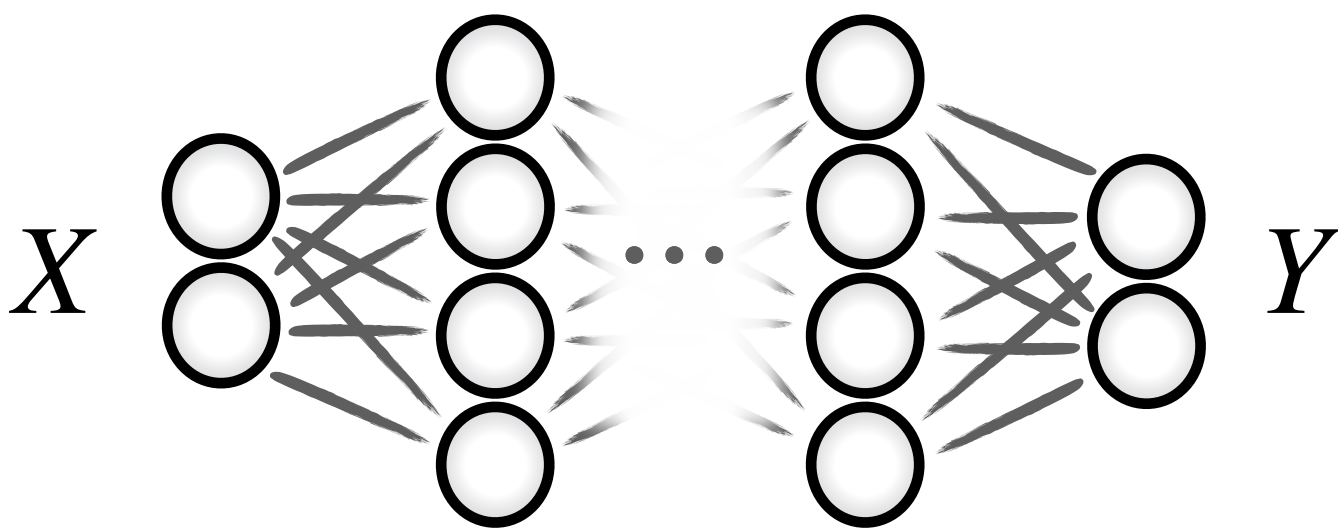
- ▶ Handles complexity,
- ▶ Extremely Fast,
- ▶ Quantifies Uncertainty.



$\approx$ **70%**
Prepare data



28%
Build and train models

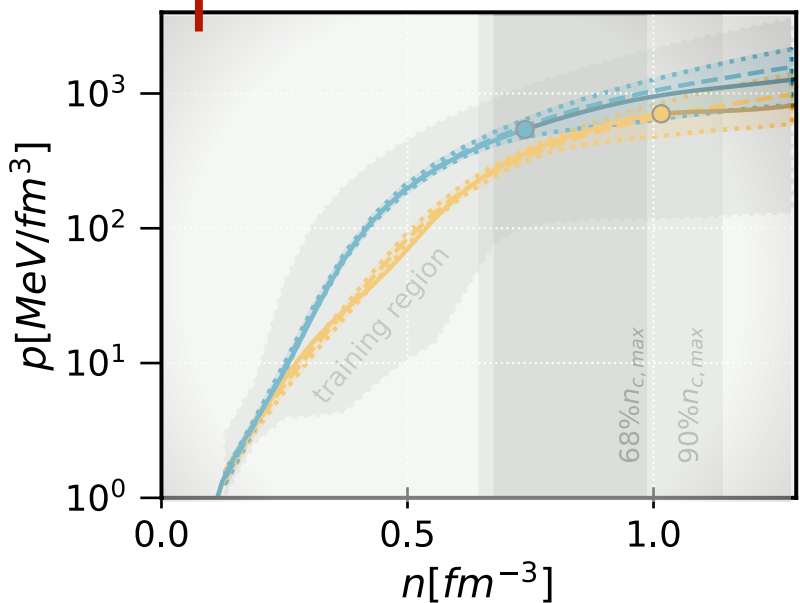


$$f_{\theta}(x) : X \rightarrow Y$$

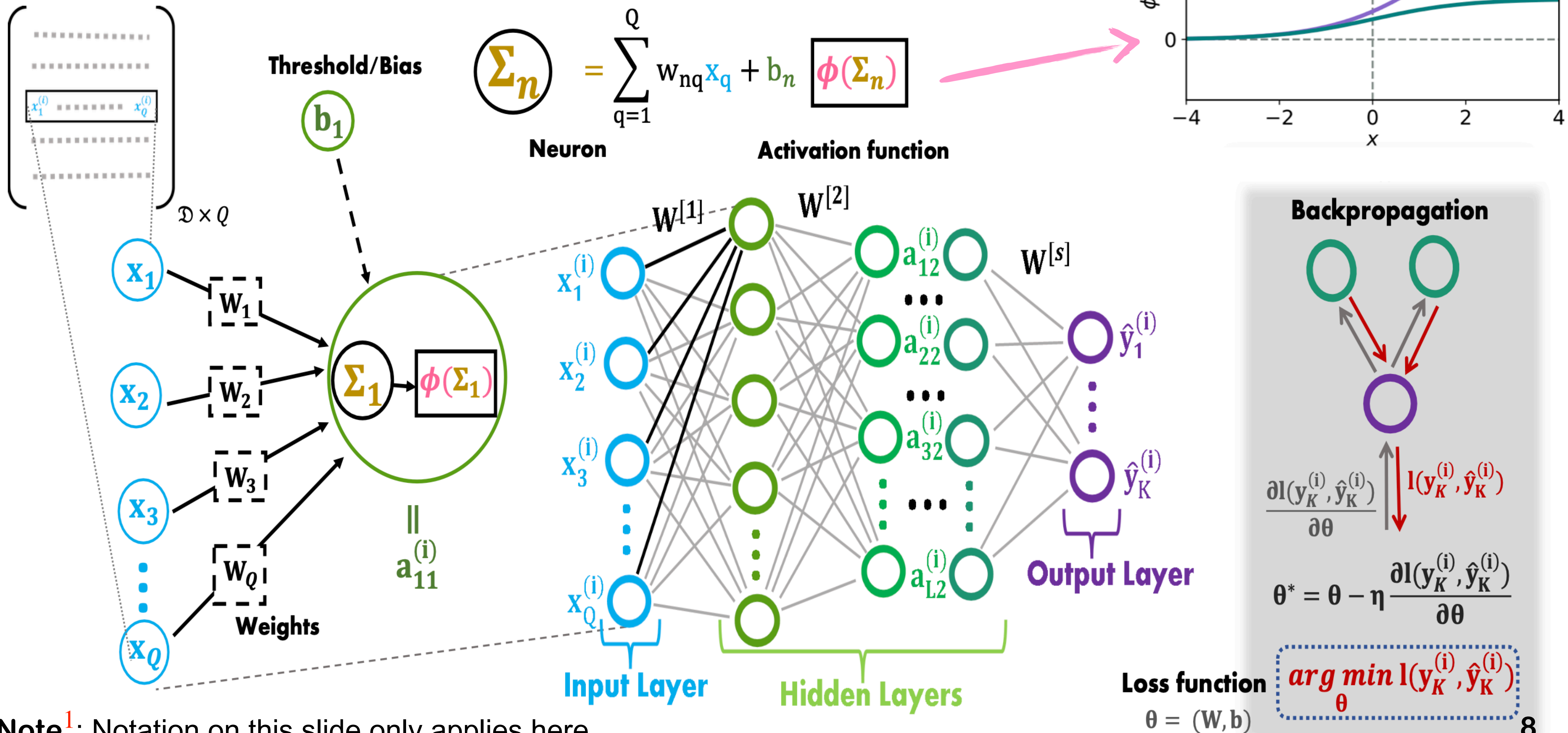


2%
Deploy and predict

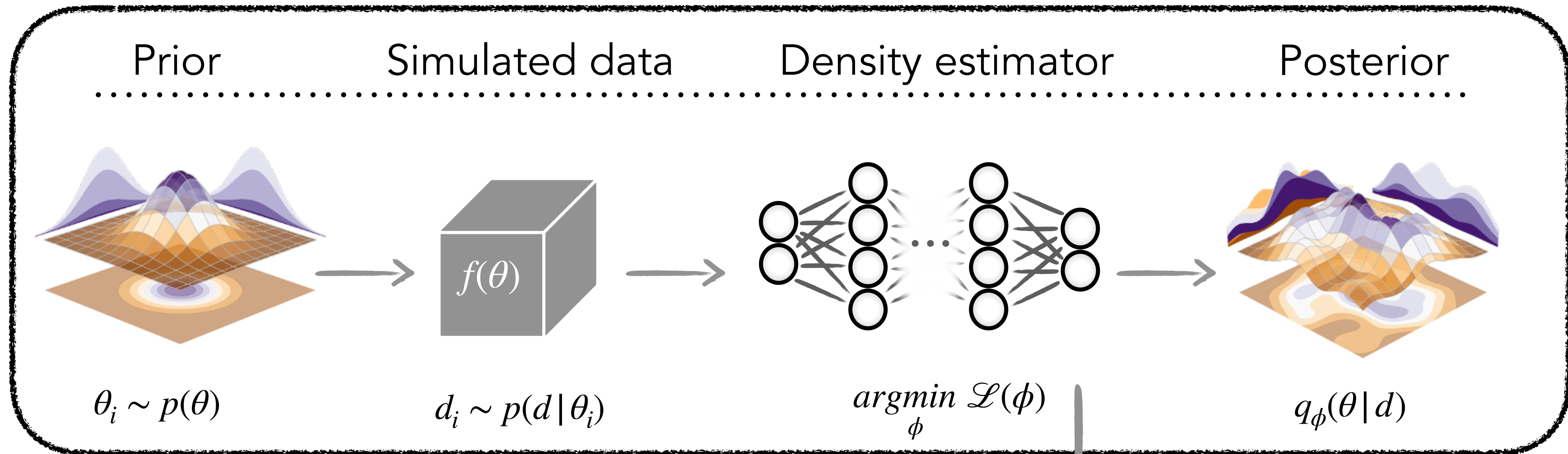
Spoiler alert !



Deep Learning quick recap¹

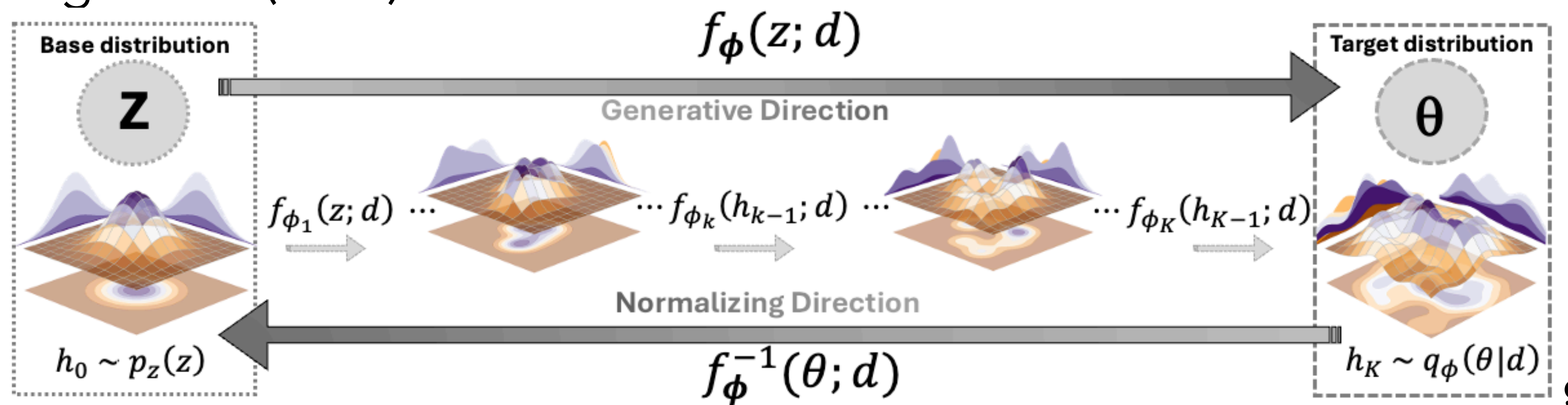


Neural posterior estimation

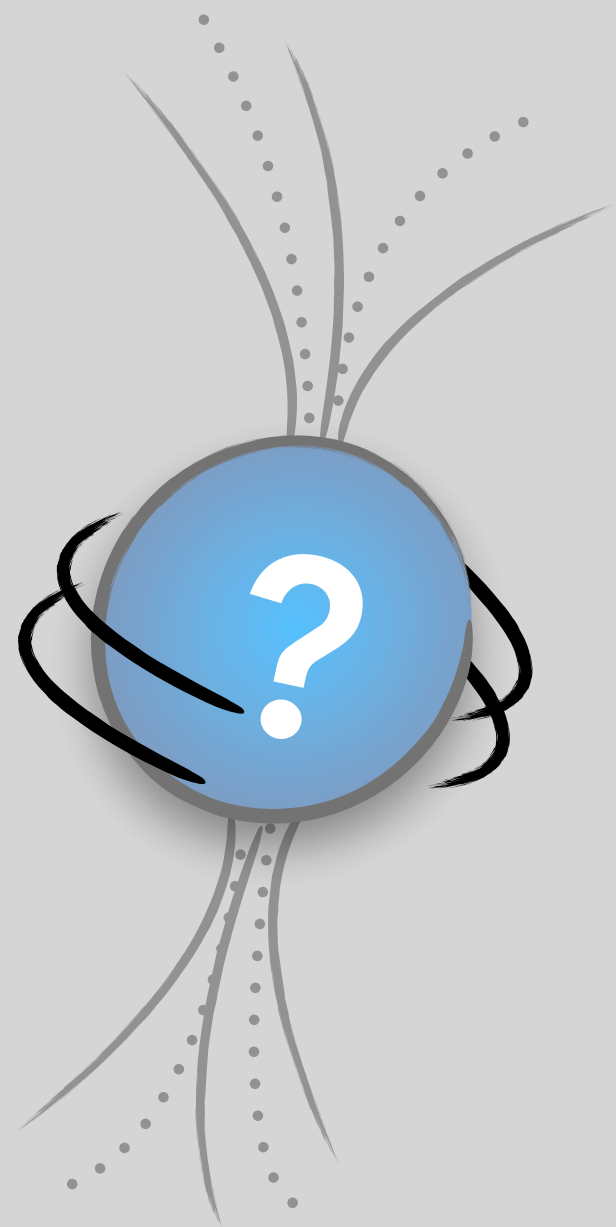


Conditional Normalizing Flows (CNF)

- ☒ Invertible
- ☒ Flexible
- ☒ Bijective



Can Neural Posterior Estimation Infer the Neutron Star Equation of State



How ?



Dataset - Quantities we aim to predict

Main goal: test if neural posterior estimation can recover the neutron star EoS from observations, obtaining the posterior $p(EoS | O)$.

The predicted physical quantities :

$$\begin{cases} \mathbf{p}(\mathbf{n}) = [p(n_1), p(n_2), \dots, p(n_{20})], \\ \mathbf{c}_s^2(\mathbf{n}) = [c_s^2(n_1), c_s^2(n_2), \dots, c_s^2(n_{20})], \\ \mathbf{\Delta}(\mathbf{n}) = [\Delta(n_1), \Delta(n_2), \dots, \Delta(n_{20})], \end{cases}$$

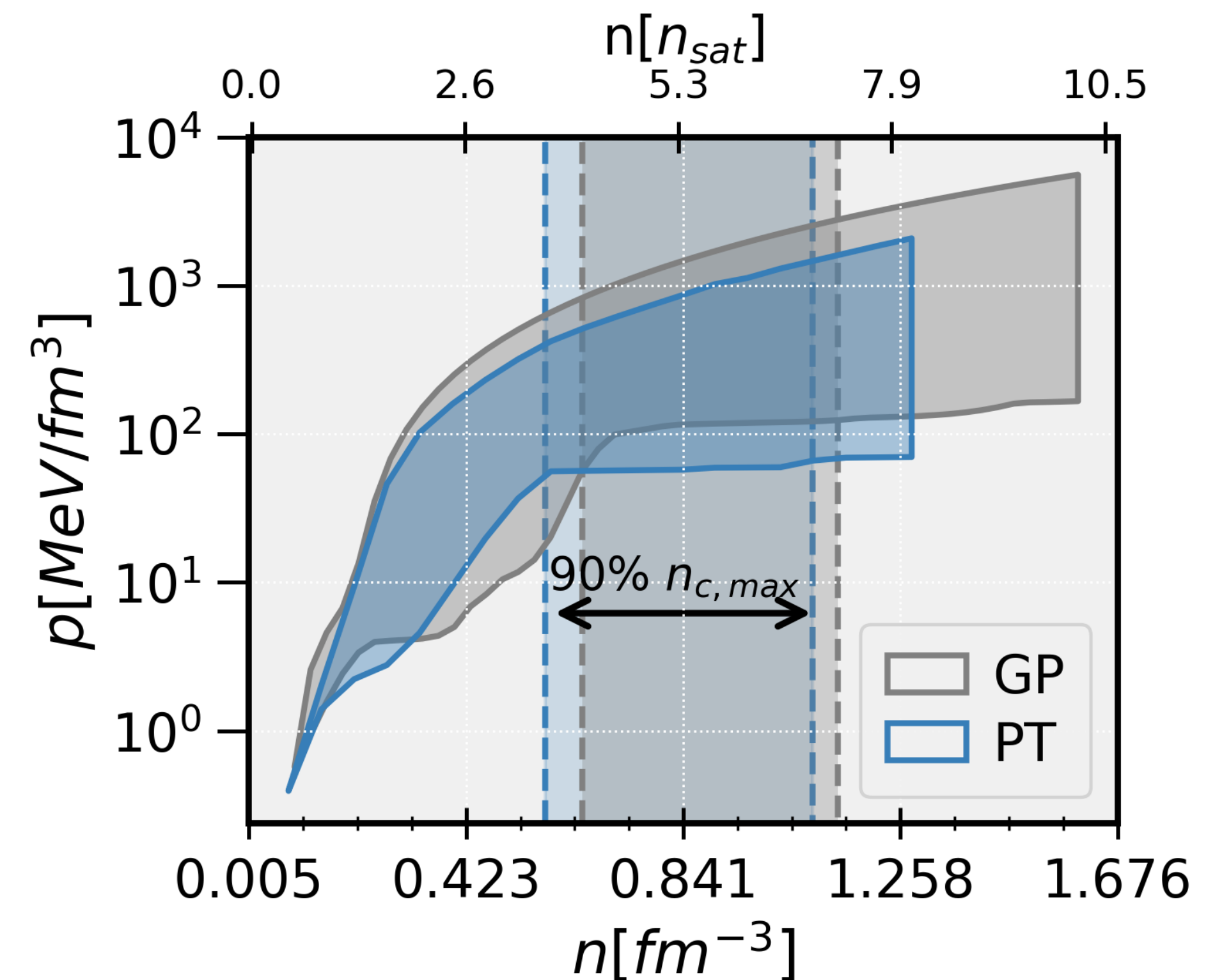
Two agnostic models :

Piece-wise Polytropics (PT)

PRD 111,023035 (2025)

Gaussian Processes (GP)

Nat Commun 14, 8451 (2023)



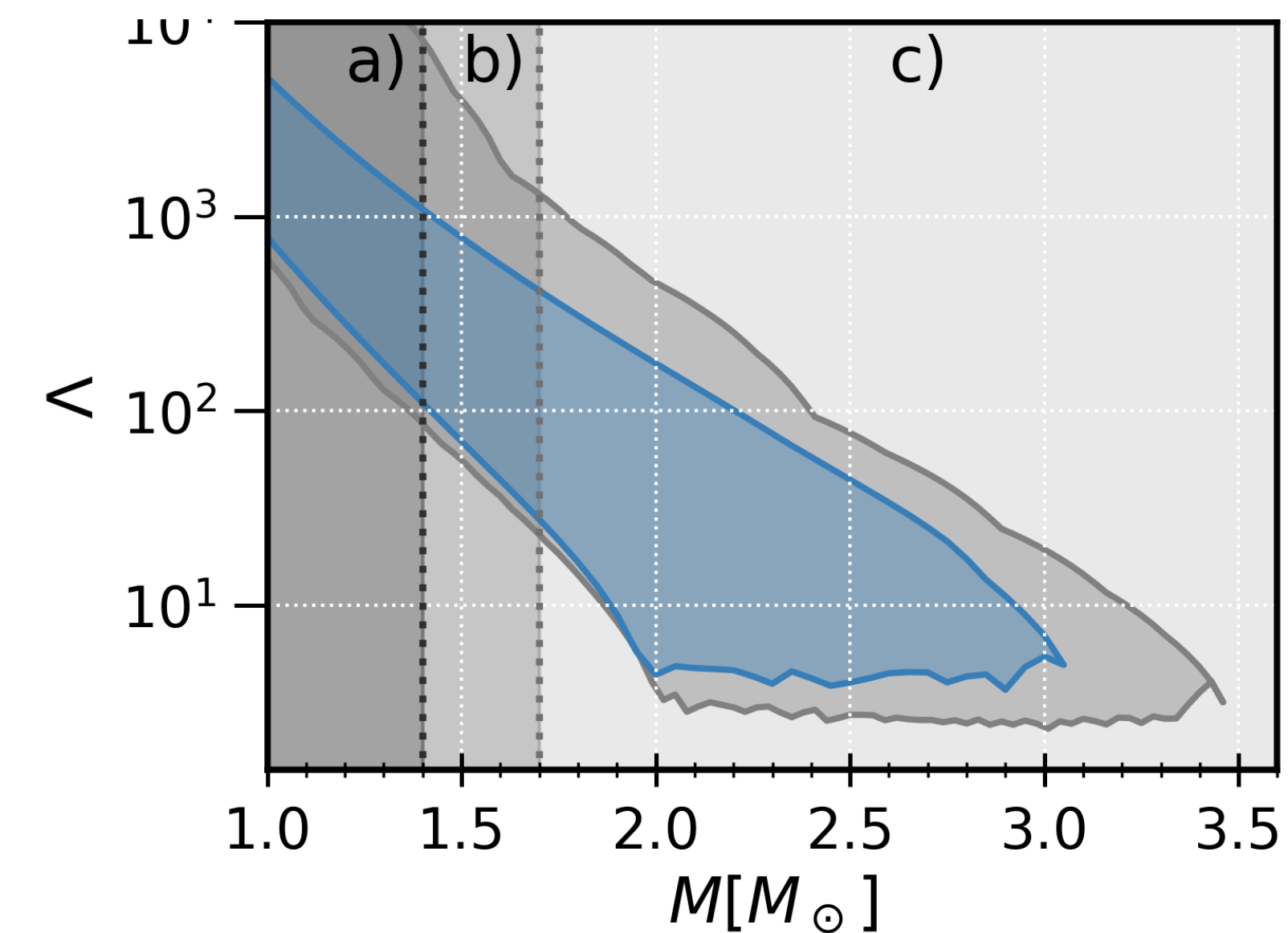
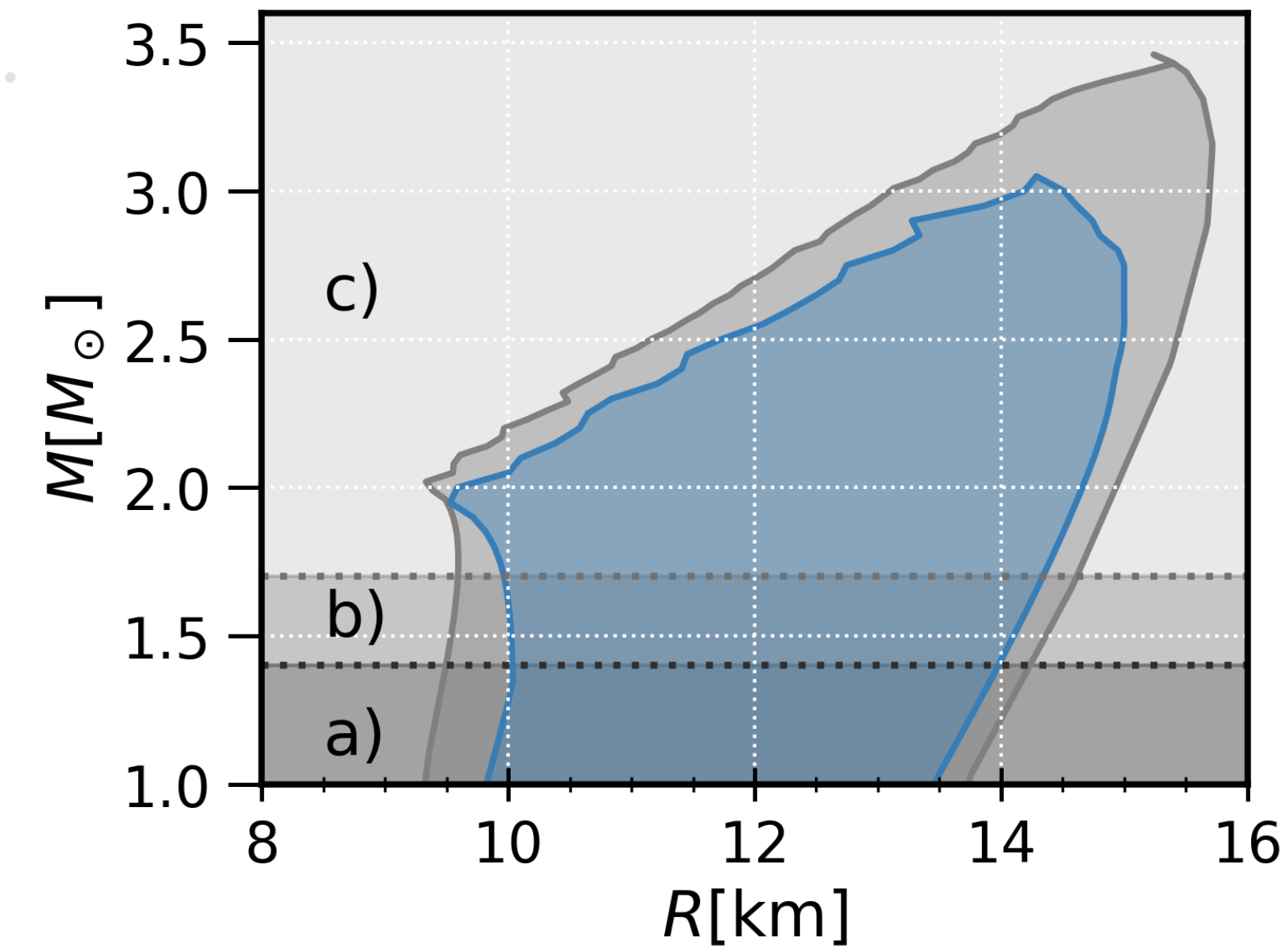
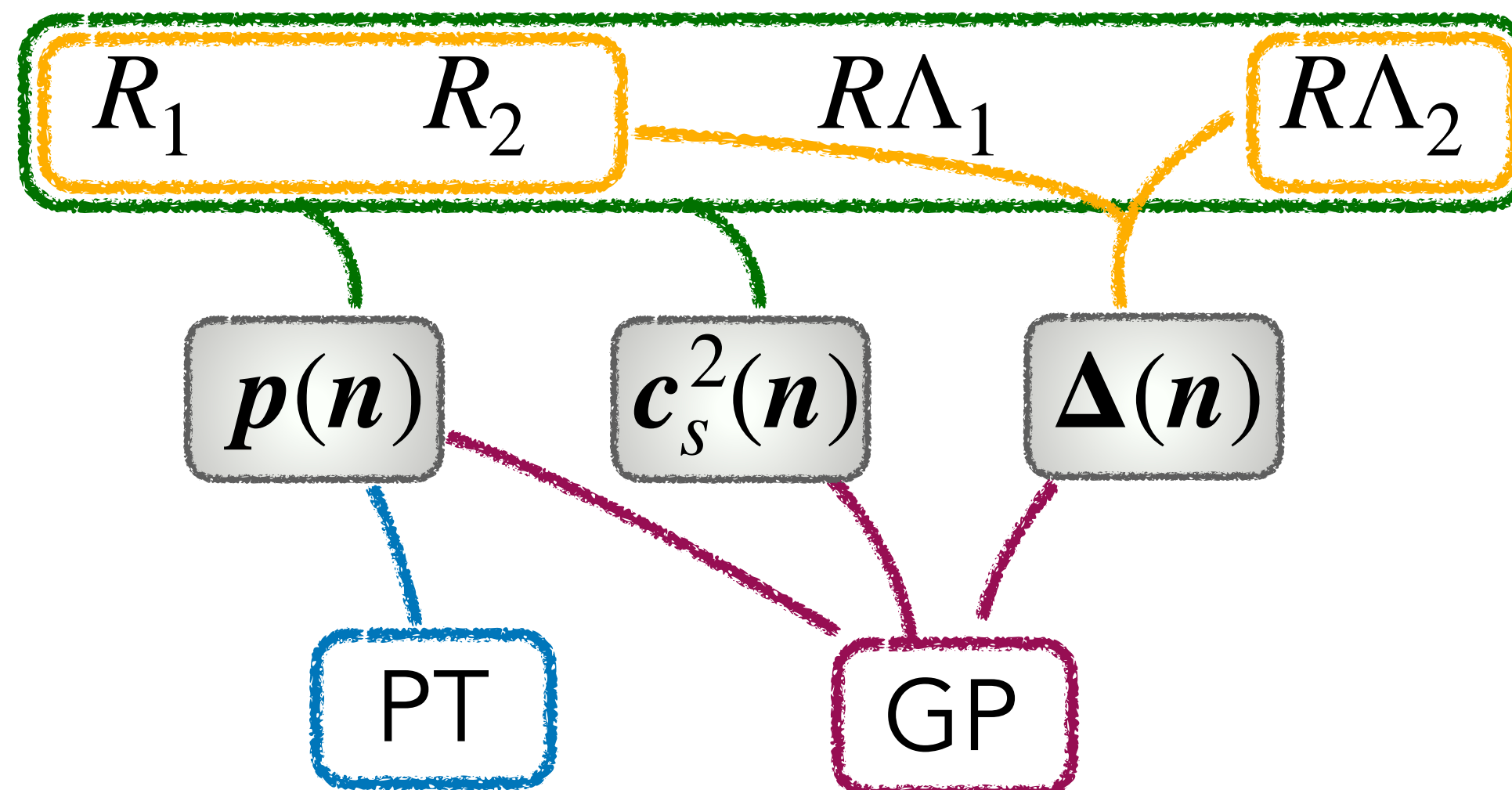
Dataset - Quantities we condition

The conditioned quantity :

$$\begin{cases} R_x = [M_1, \dots, M_{15}, R_1, \dots, R_{15}], x \in [1, 2] \\ R\Lambda_x = [M_1, \dots, M_{15}, R_1, \dots, R_{15}, M_1^*, \dots, M_{15}^*, \Lambda_1, \dots, \Lambda_{15}]. \end{cases}$$

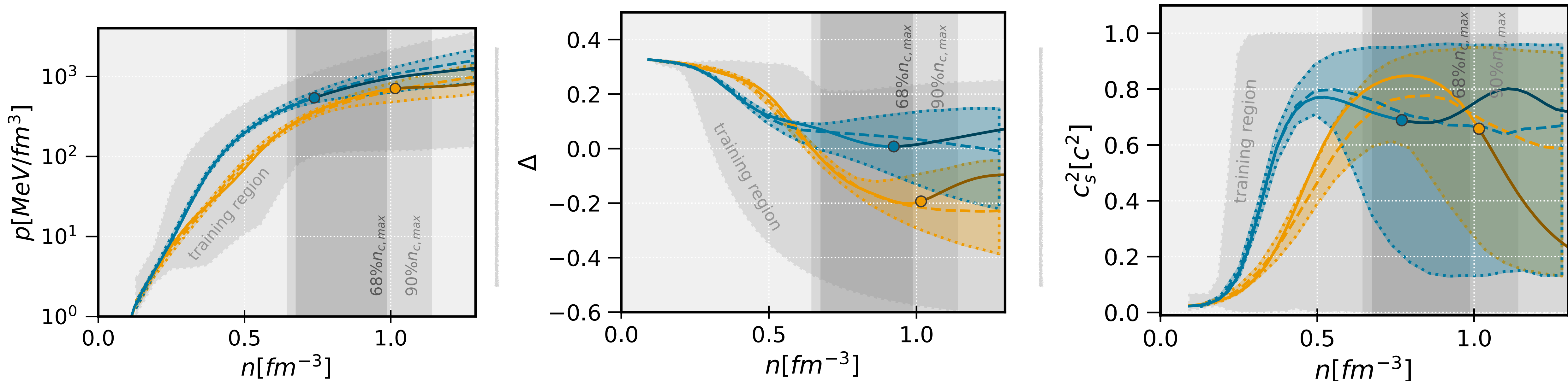
$x = 1$ without noise, $x = 2$ with gaussian noise.

The datasets:



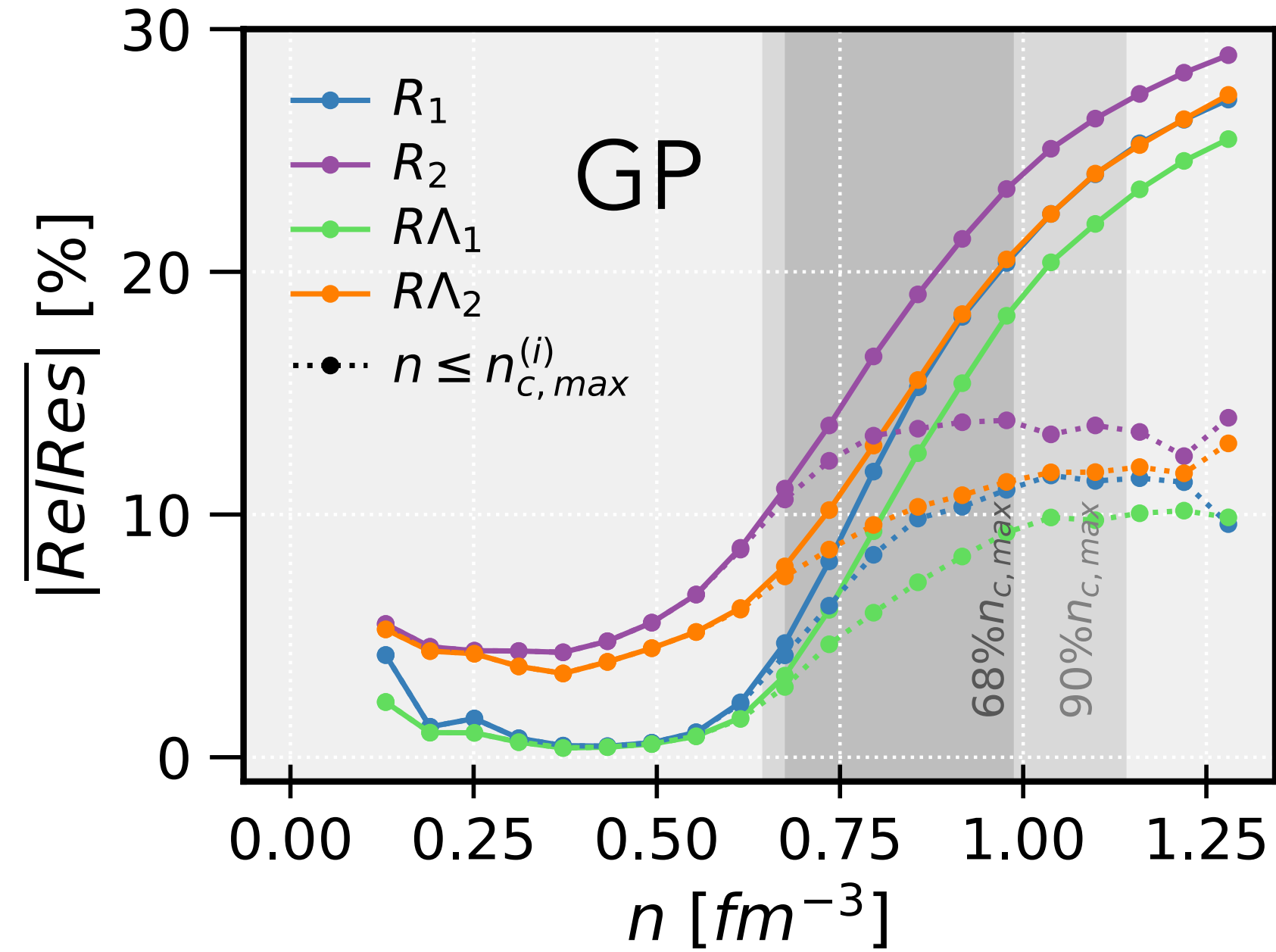
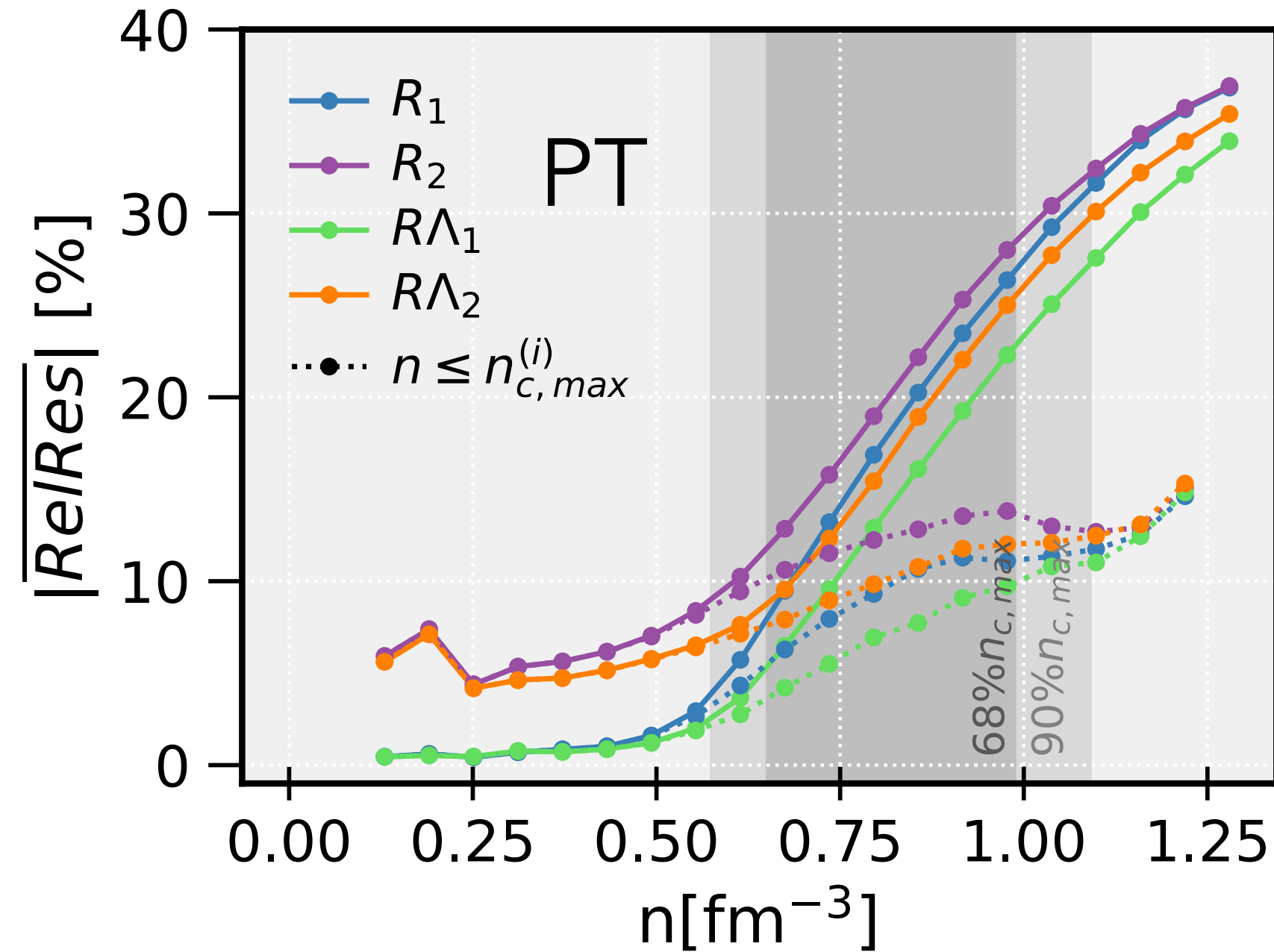
Key result: Accurate EoS reconstruction

- Prediction for 2 samples of the test set $R\Lambda_2$ —, with a 90 % CI  and median ,
- Increase in dispersion near maximum central density, represented by ,
- Predictions always inside the CI.



Just for GP dataset

Relative error for pressure



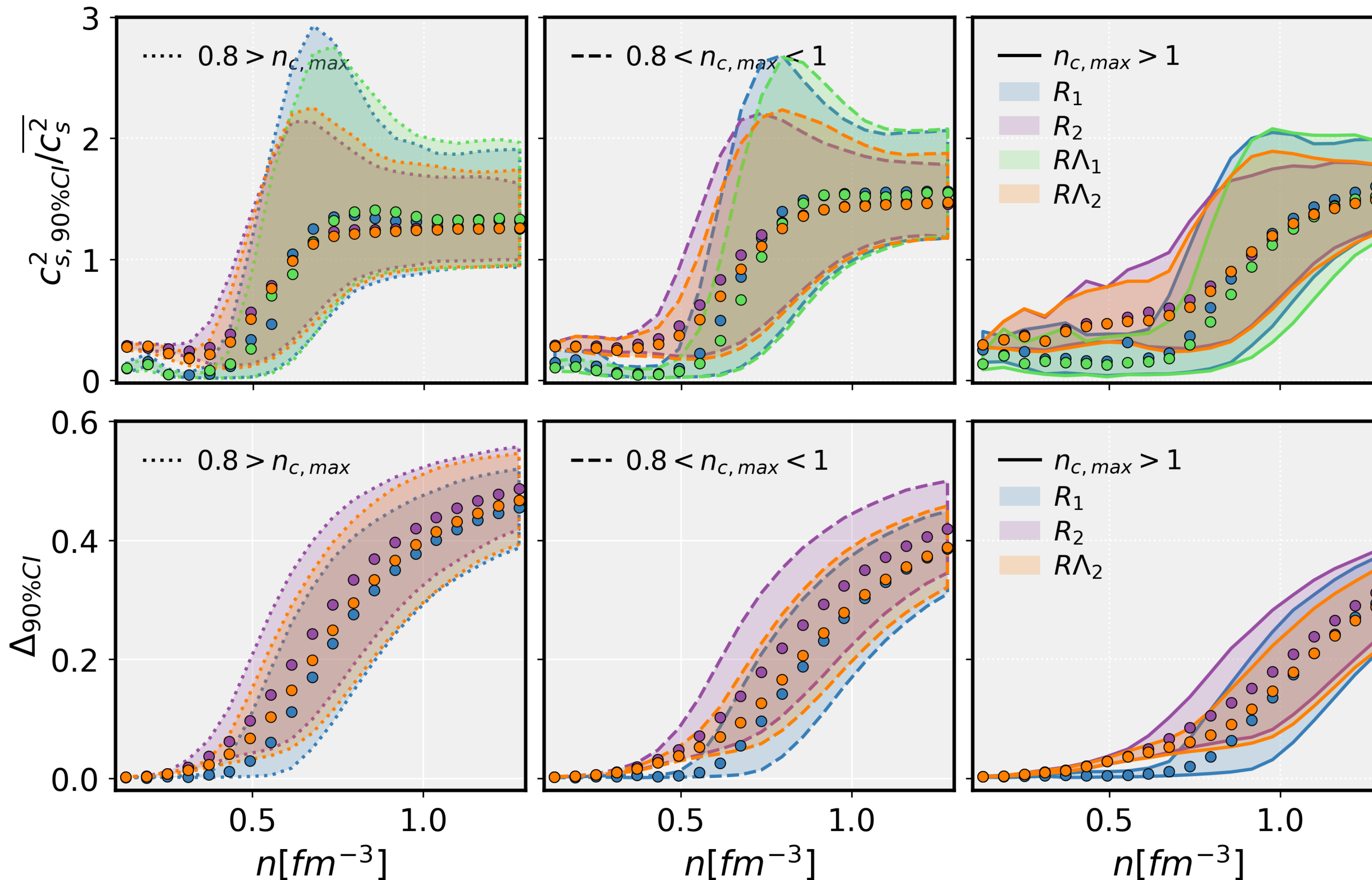
$$\text{RelRes}^{(i)}(n) = \text{Med}_l \left[\frac{X_p^{(i,l)}(n) - X_T^{(i)}(n)}{X_T^{(i)}(n)} \right] \times 100$$

$$\begin{cases} |\overline{\text{RelRes}}|_{\leq n_{c,max}} = \text{Med}_i |\text{RelRes}^{(i)}(n)| & \text{for } n \leq n_{c,max}^{(i)} \\ |\overline{\text{RelRes}}| = \text{Med}_i |\text{RelRes}^{(i)}(n)| \end{cases}$$

l = Posterior sample, i = EoS at density n

- Error increases with noise,
- Error decreases with tidal deformability,
- Error decreases when filtered for $n_{c,max}$.

Effect of maximum central density

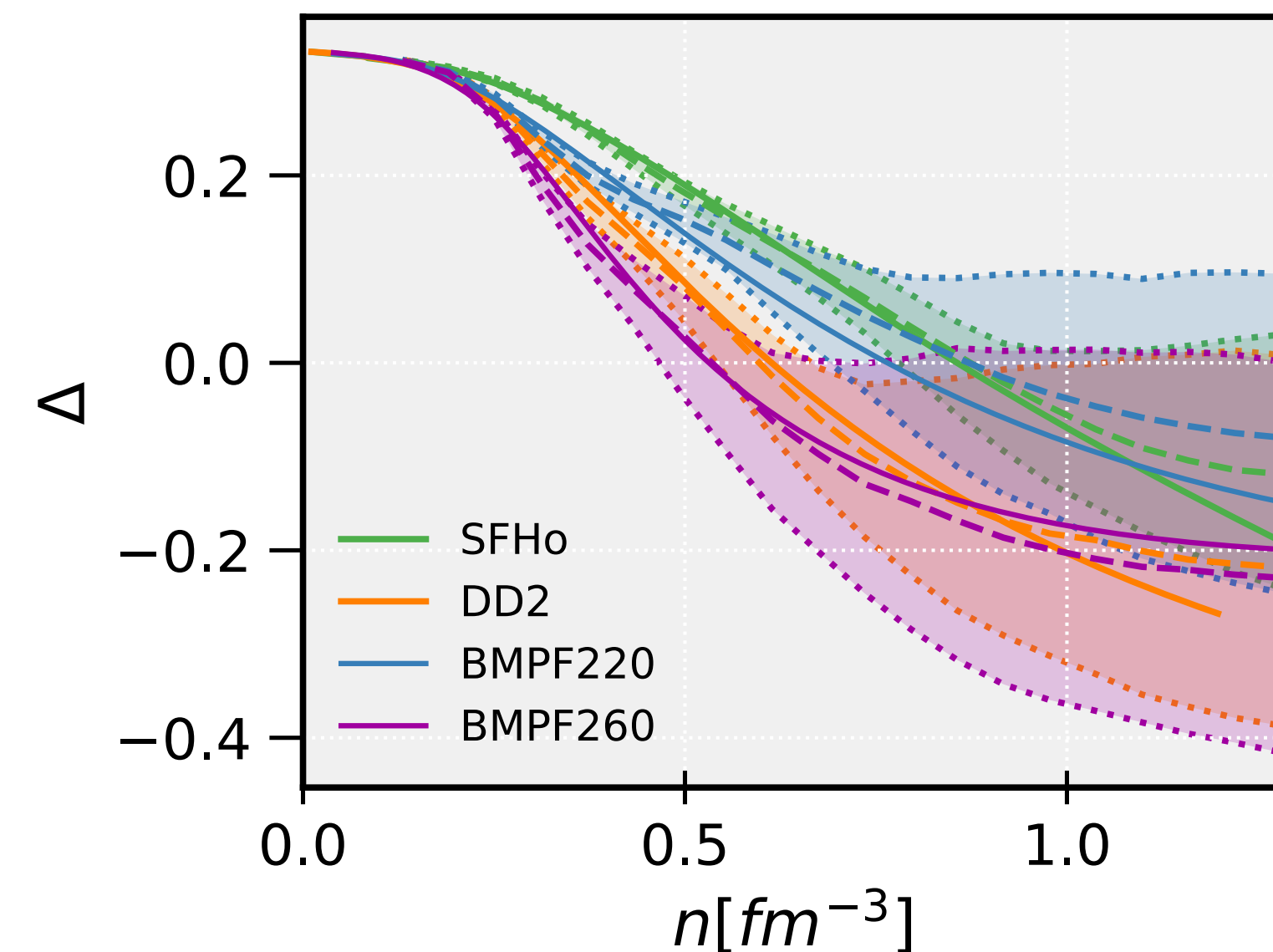
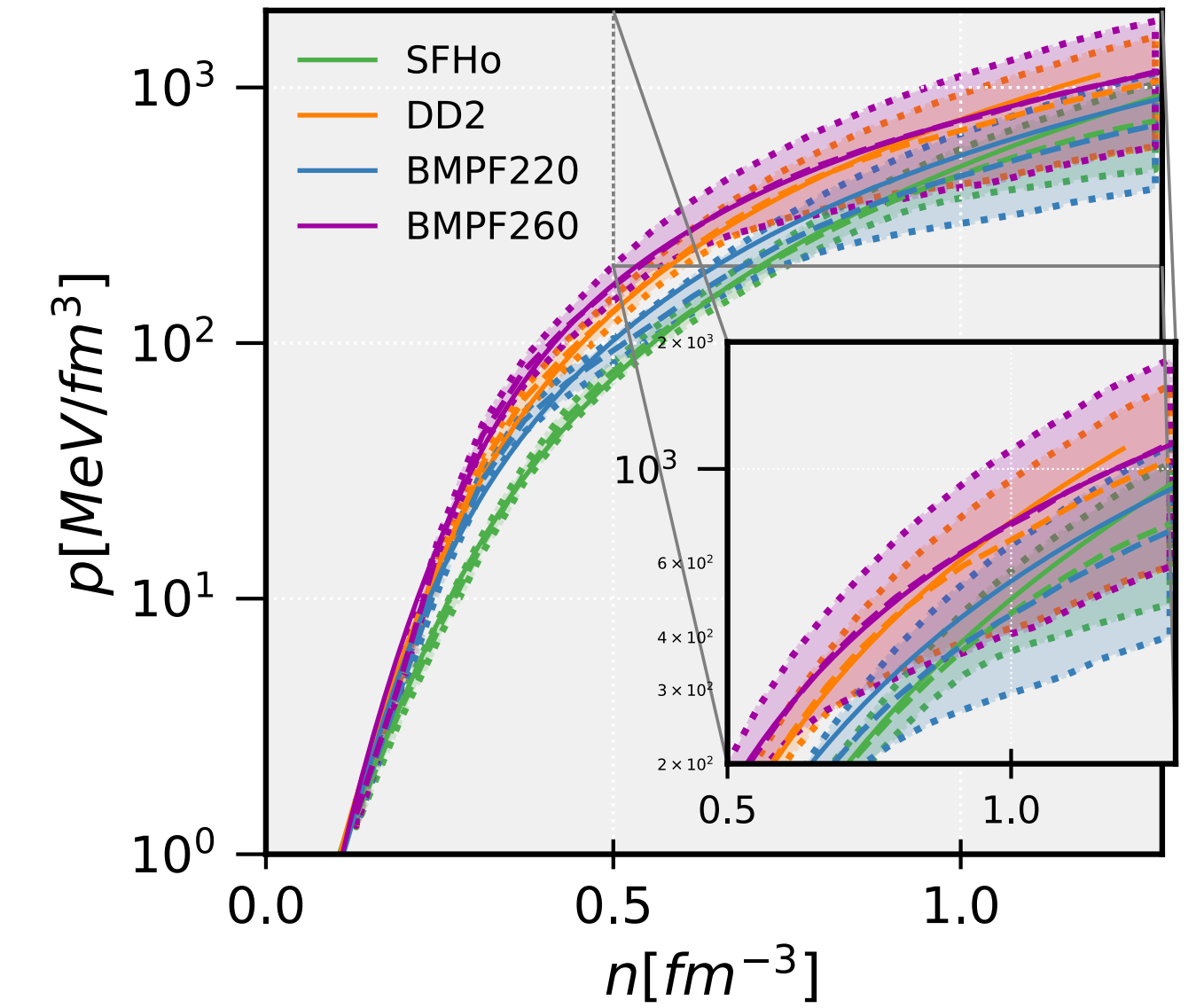
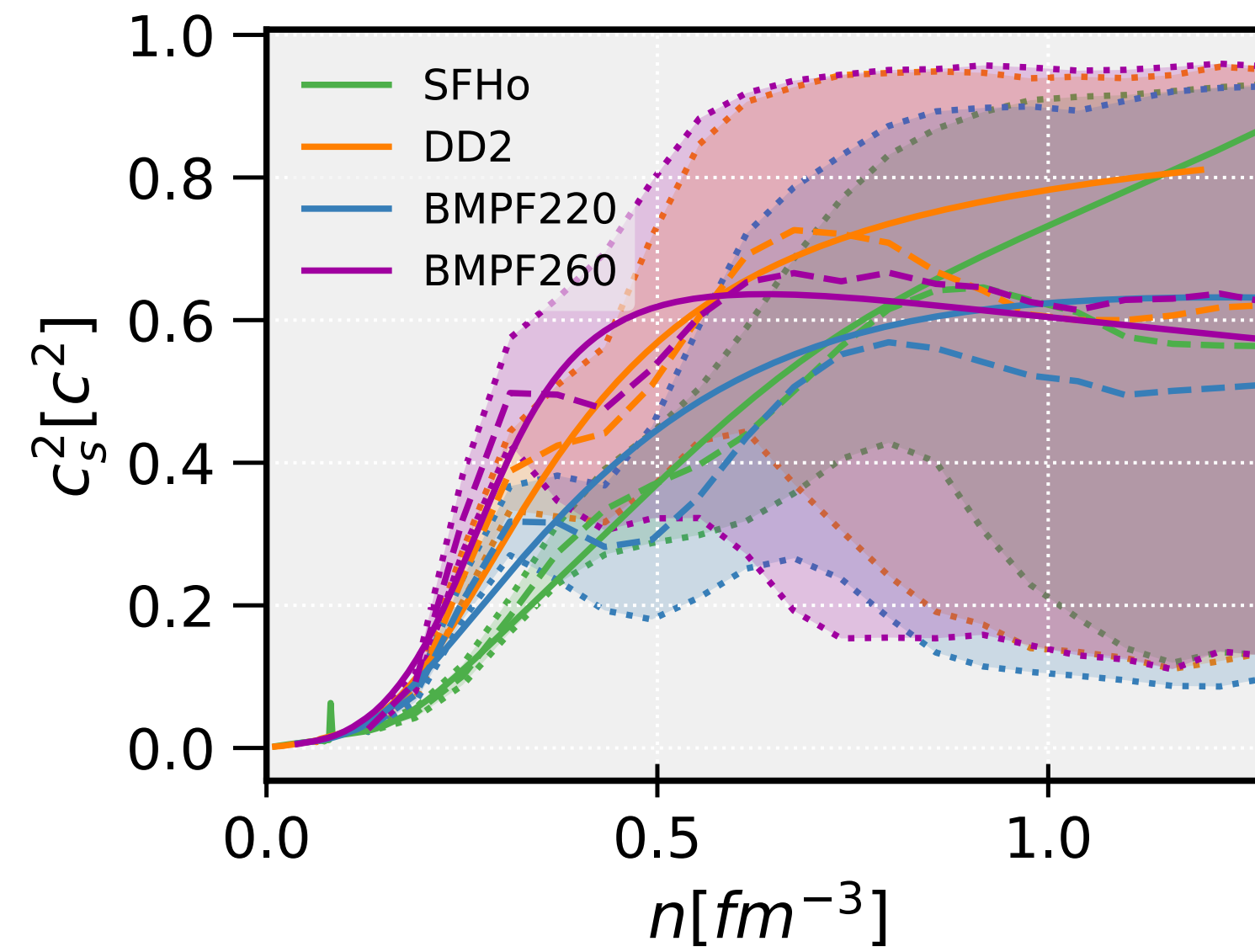
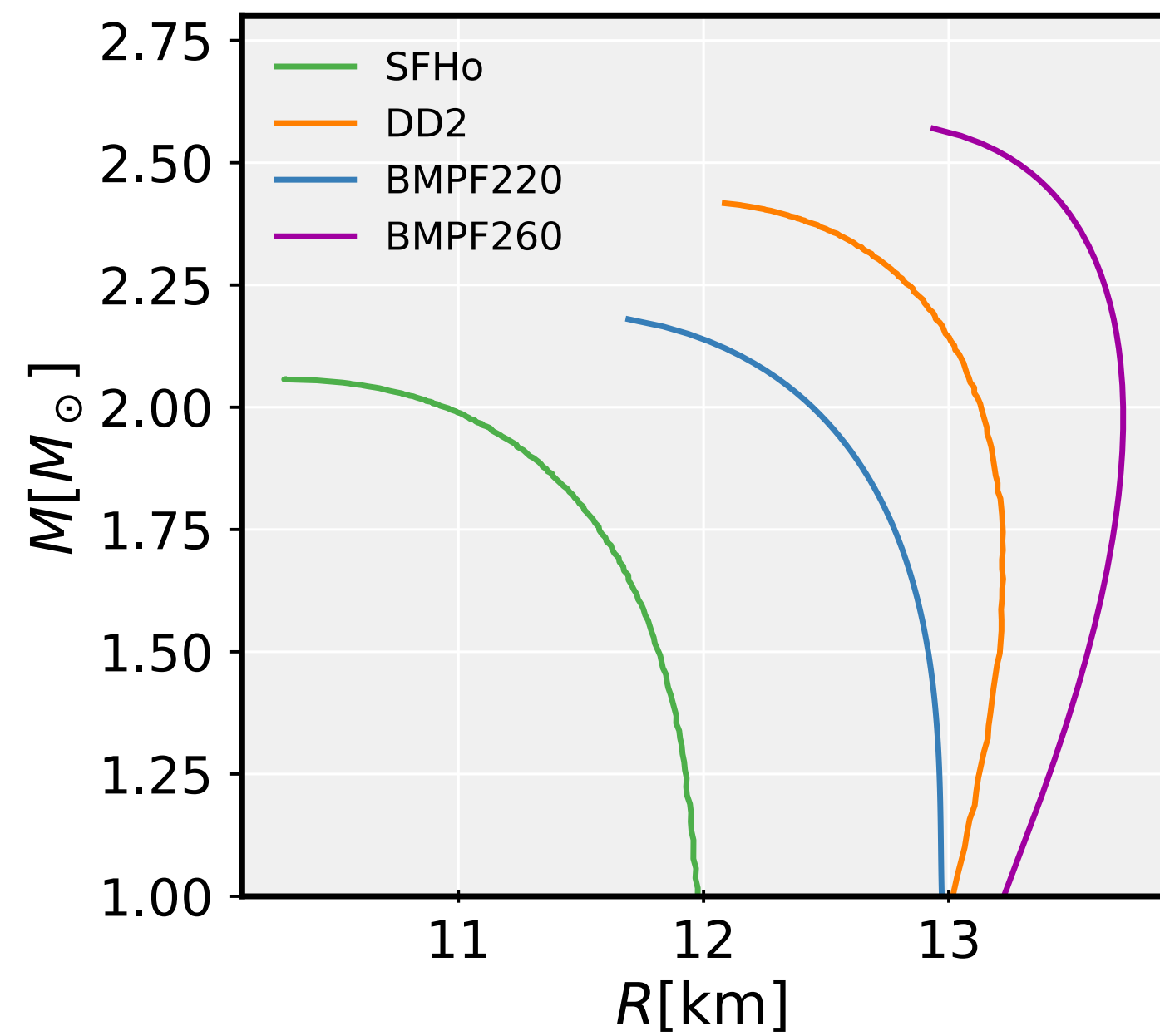


- ▶ Bands  are the 90% CI,
- ▶ Dots  are the mean.

- ▶ Predicted dispersion (CI) decrease with the increase of $n_{c,max}$.

Another dataset inference test

- For the R_2 dataset we tested 4 EoS,
- Very different models.



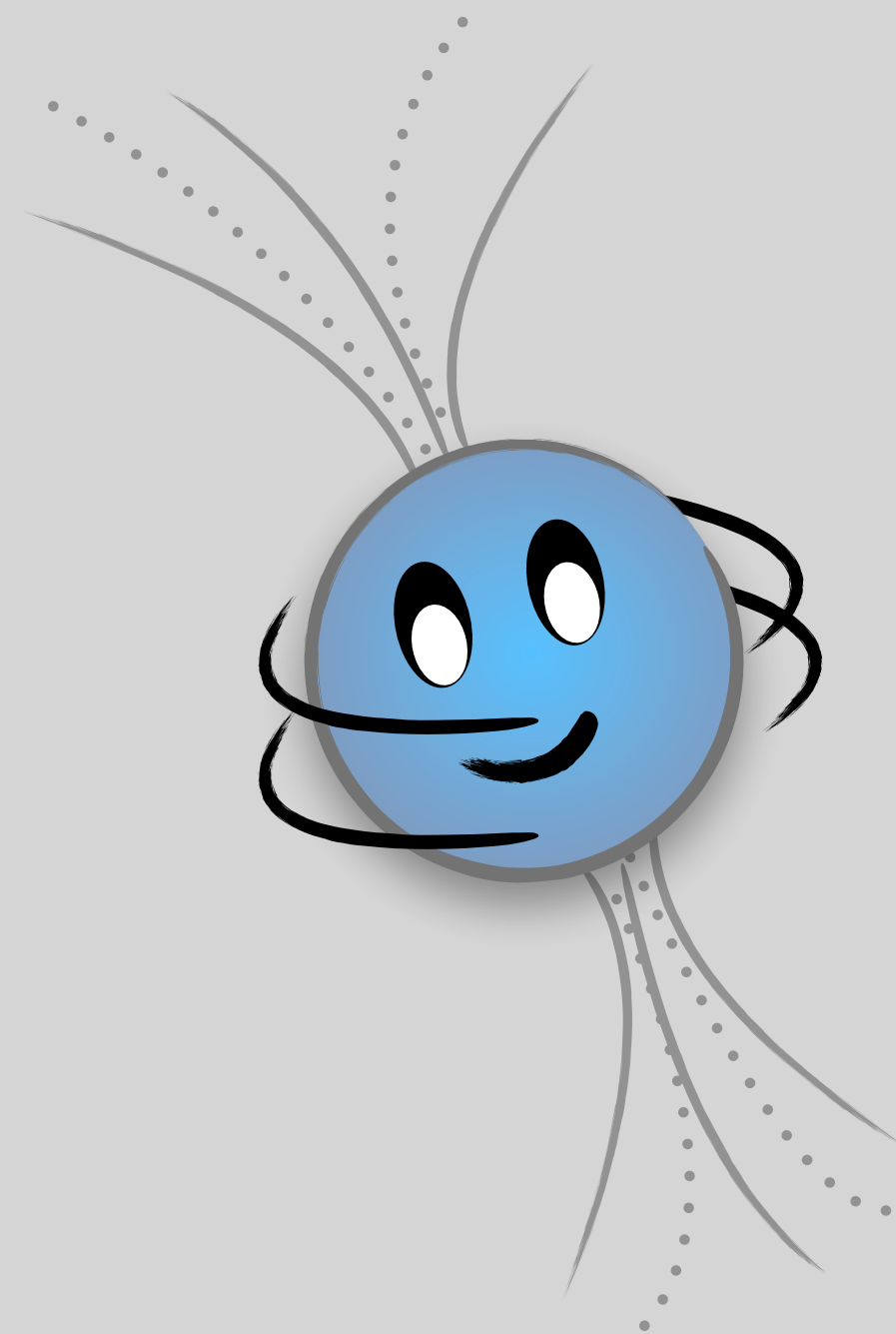
Just for GP dataset

Summary

- ▶ **Validation of neural posterior estimation:** Demonstrated that CNF can successfully recover neutron star EoS,
 - ▶ **Crucial role of tidal deformability:** Including Λ alongside mass–radius data improves predictions,
 - ▶ **Sensitivity to maximum central density:** The model naturally learns correlations between predictive uncertainty and the $n_{c,max}$,
 - ▶ **Uncertainty quantification:** NPE provides well-calibrated posteriors.
- ▶ **The outlook:** This method is well-suited for the upcoming “golden age” of multimessenger data, offering a promising tool to constrain dense matter physics.



arXiv:
2507.23506



**Thank you for
your attention**

More information available at:
val.mar.dinis@gmail.com

| Backup

Dataset - Mock dataset

$$\mathbf{M}^0 = [M_1^a, \dots, M_{n_o}^a, M_1^b, \dots, M_{n_o}^b, M_1^c, \dots, M_{n_o}^c]$$

$$\mathbf{R}^0 = [R(M_1^a), \dots, R(M_{n_o}^a), R(M_1^b), \dots, R(M_{n_o}^b), R(M_1^c), \dots, R(M_{n_o}^c)]$$

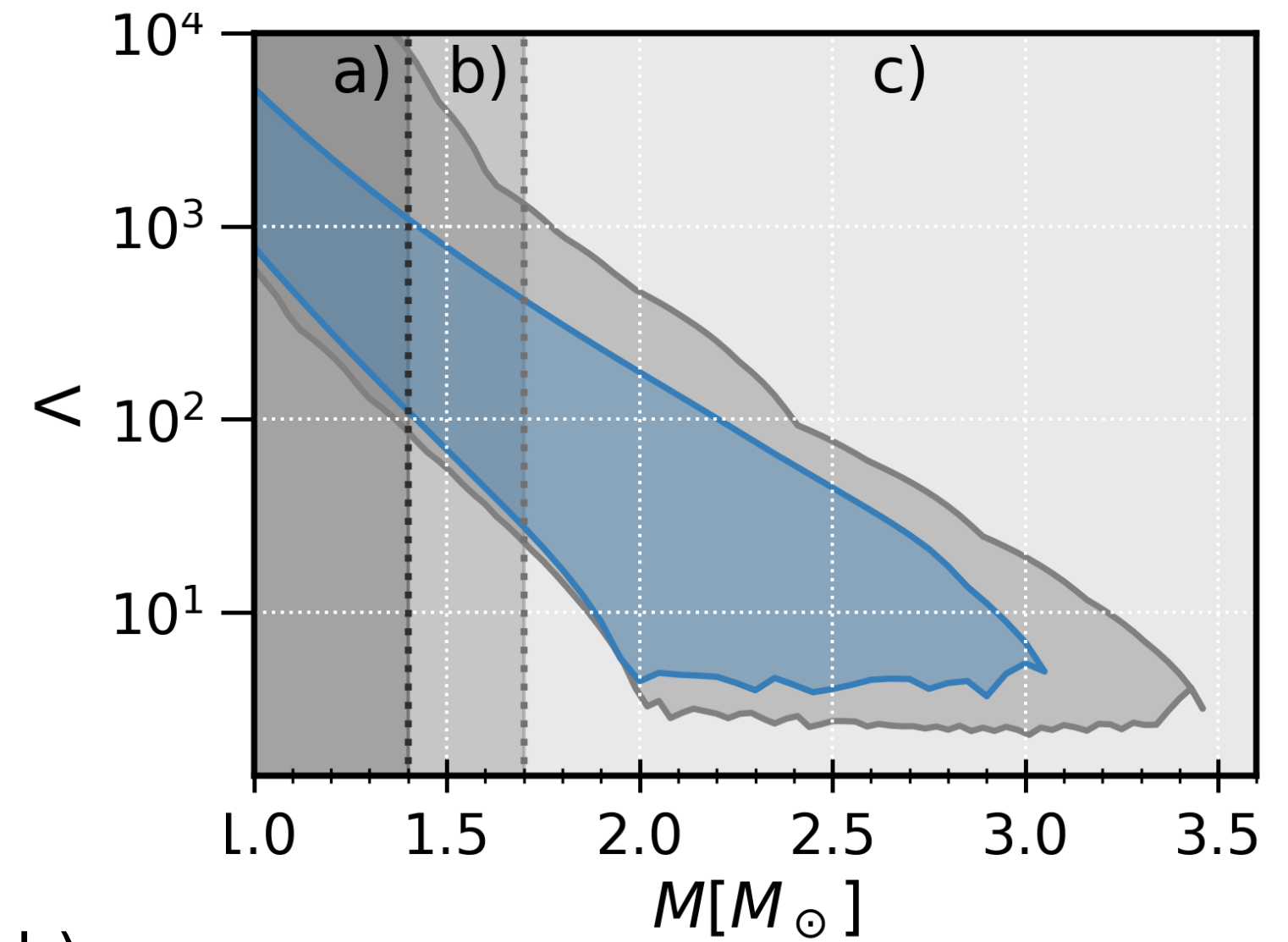
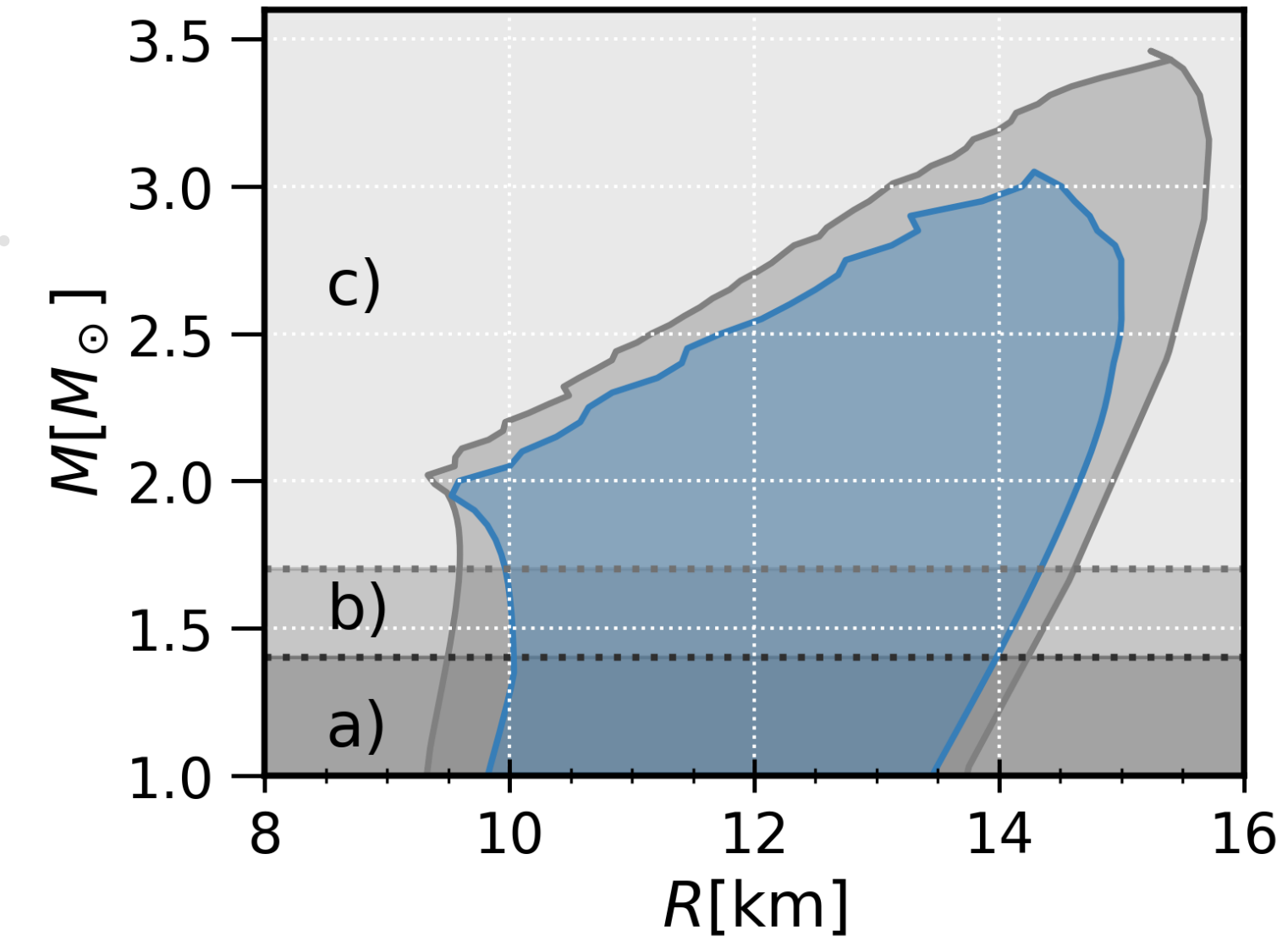
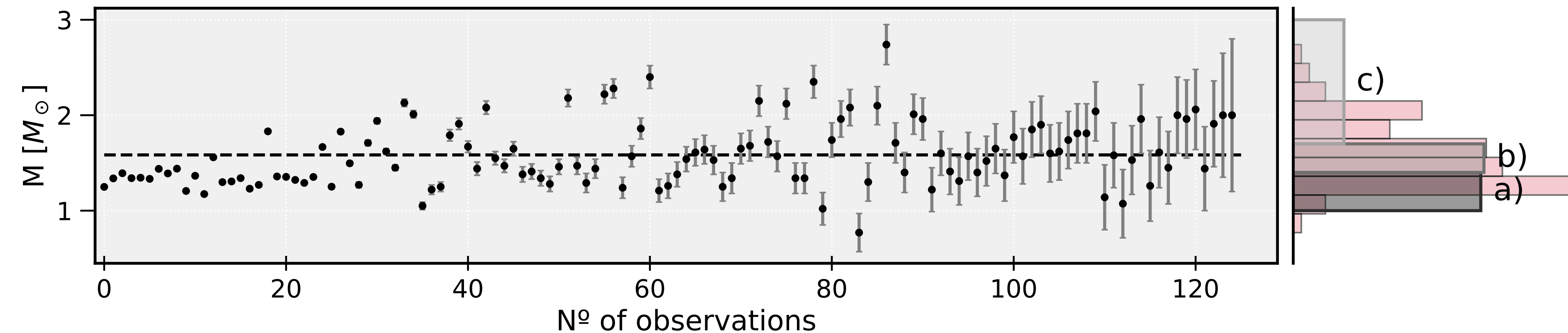
$$\mathbf{M}^* = [M_1^{*a}, \dots, M_{n_o}^{*a}, M_1^{*b}, \dots, M_{n_o}^{*b}, M_1^{*c}, \dots, M_{n_o}^{*c}]$$

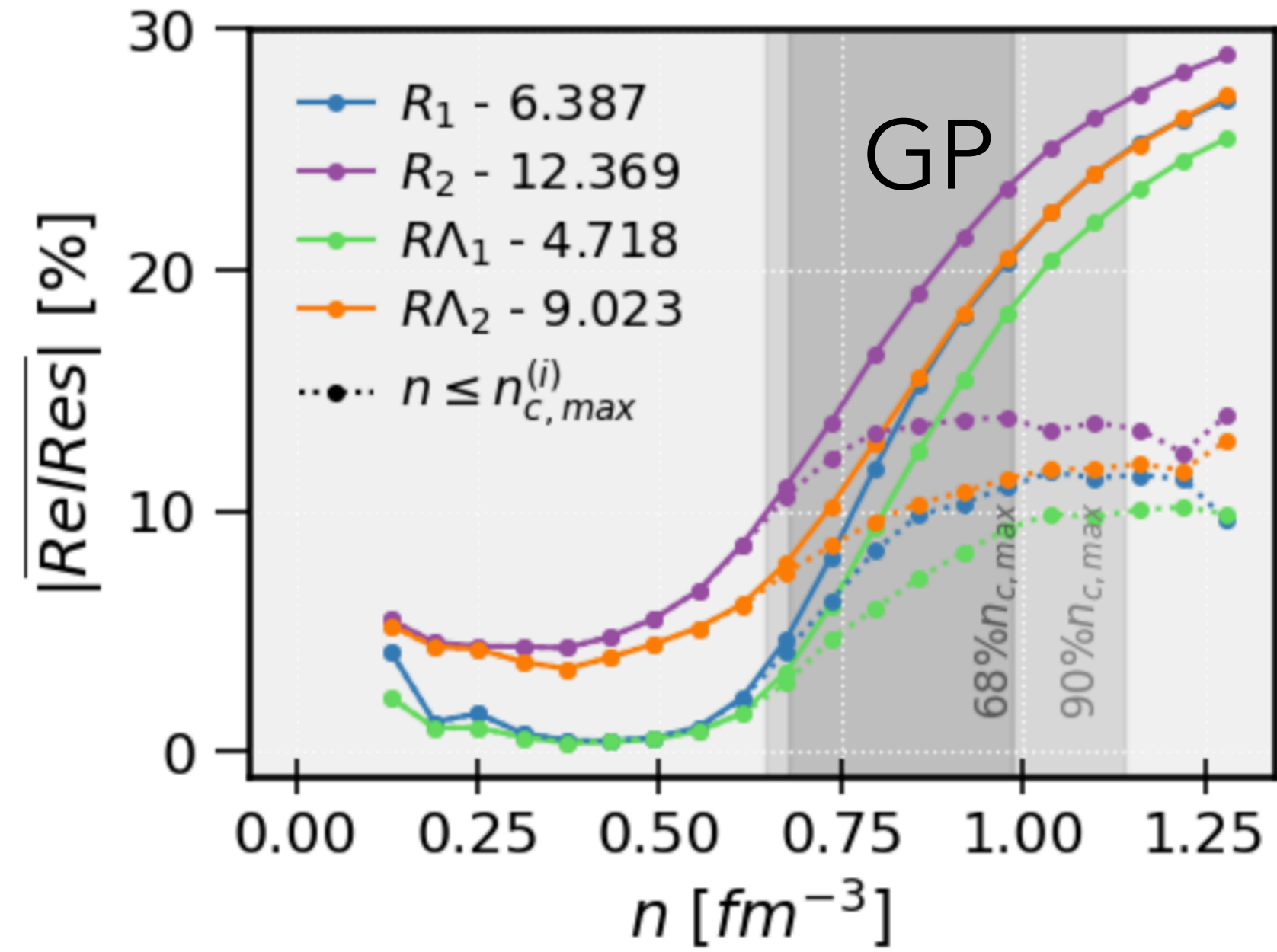
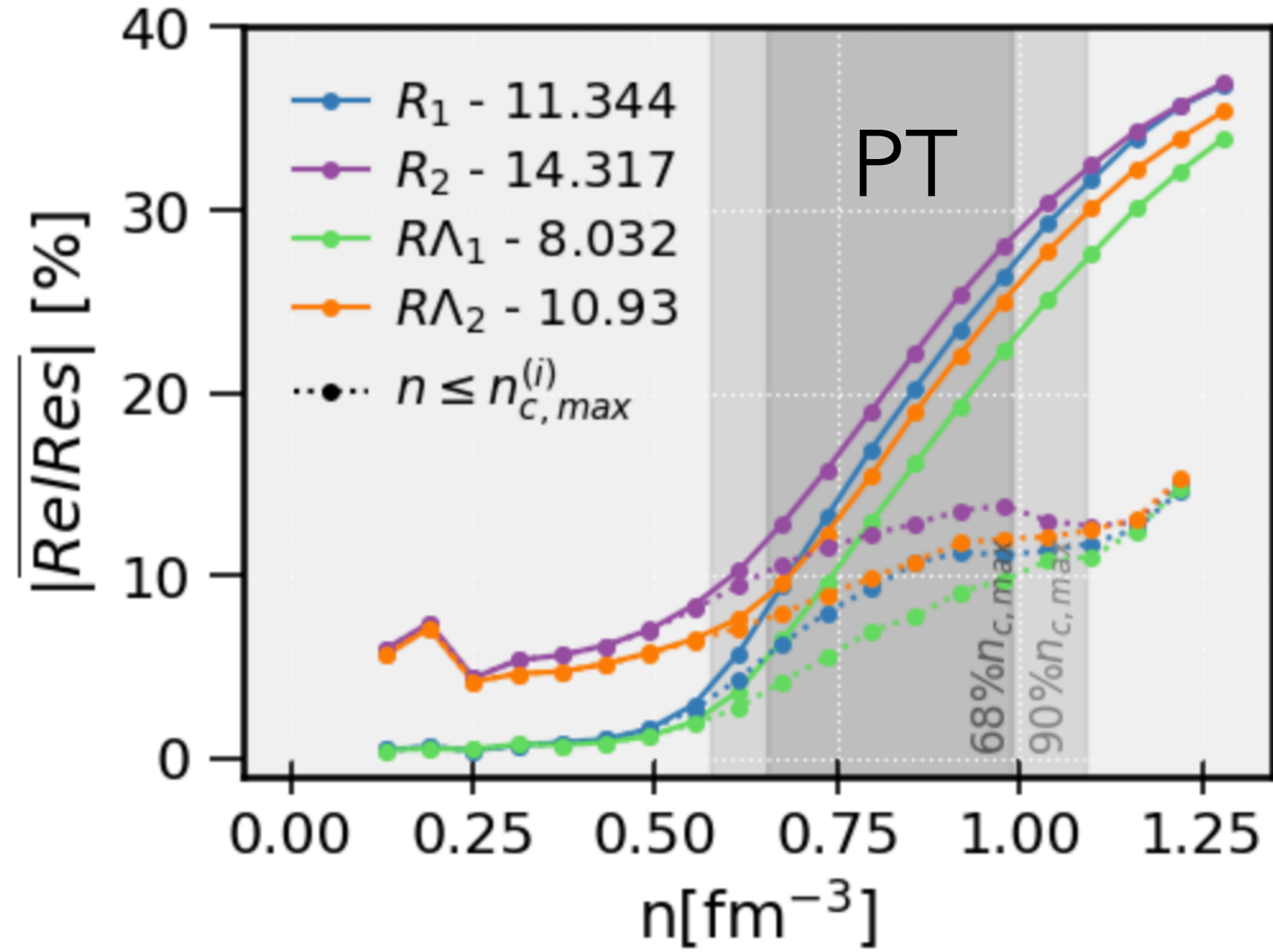
$$\mathbf{\Lambda}^0 = [\Lambda(M_1^{*a}), \dots, \Lambda(M_{n_o}^{*a}), \Lambda(M_1^{*b}), \dots, \Lambda(M_{n_o}^{*b}), \Lambda(M_1^{*c}), \dots, \Lambda(M_{n_o}^{*c})].$$

$$\mathbf{M} \sim \mathcal{N}(\mathbf{M}^0, \sigma_{\mathbf{M}}^2), \quad \mathbf{R} \sim \mathcal{N}(\mathbf{R}^0, \sigma_{\mathbf{R}}^2)$$

$$\sigma_{\mathbf{M}} \sim \mathcal{U}[0, \sigma_M], \quad \sigma_{\mathbf{R}} \sim \mathcal{U}[0, \sigma_R],$$

$$\mathbf{\Lambda} \sim \mathcal{N}(\mathbf{\Lambda}^0, \sigma_{\mathbf{\Lambda}}^2(\mathbf{M}^*)), \quad \sigma_{\mathbf{\Lambda}}(\mathbf{M}^*) \sim \mathcal{U}[0, \sigma_{\mathbf{\Lambda}}(M^*)].$$





$$\eta = \frac{\sum_n^{20} |\overline{\text{RelRes}}|_{\leq n_{c,\max}}}{\sum_n^{20} |\overline{\text{RelRes}}|},$$

(0.55, 0.67, 0.51, 0.63) respectively for $(R_1, R_2, R\Lambda_1, R\Lambda_2)$ sets, while for the PT dataset we find (0.41, 0.53, 0.41, 0.53).

Likelihood-based or simulation-based?

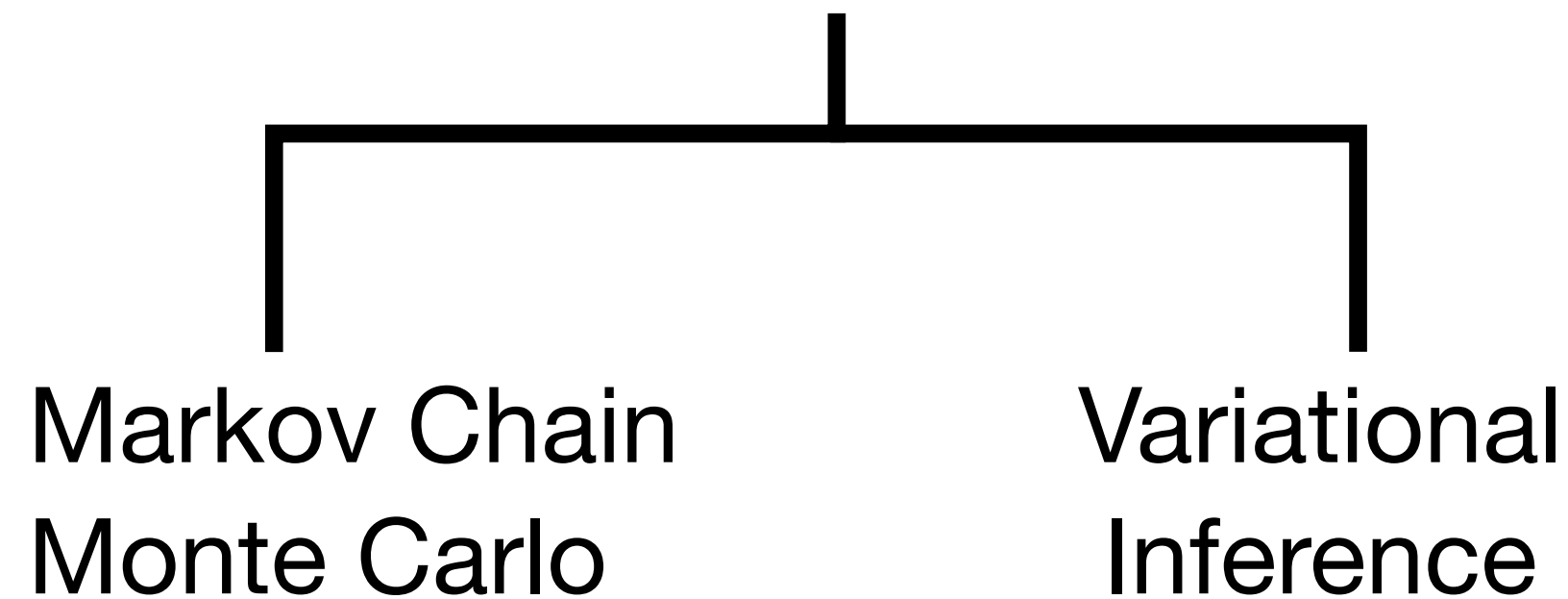
.....

Bayes' theorem

$$\text{Posterior } p(\theta | d) = \frac{\text{Likelihood } p(d | \theta) \text{ Prior } p(\theta)}{\text{Evidence } p(d)}$$

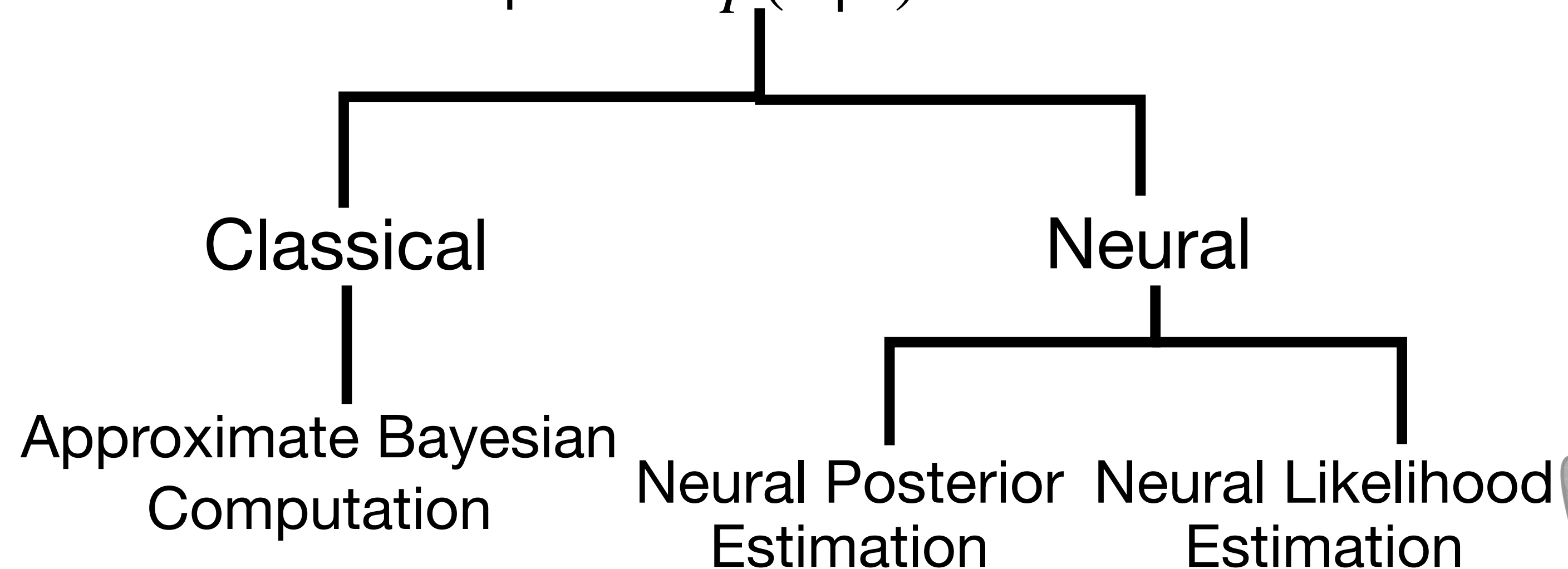
Likelihood-based Inference

Evaluate $p(d | \theta)$



Simulation-based Inference

Sample $d \sim p(d | \theta)$

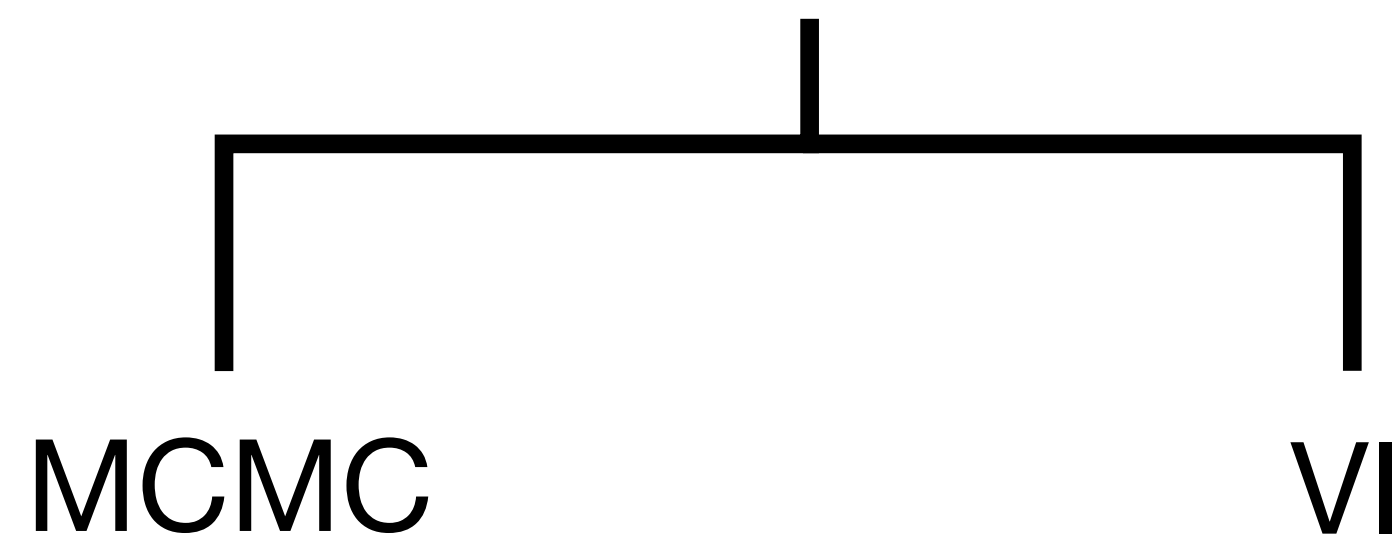


Likelihood-based or simulation-based?

Bayes' theorem $p(\theta | d) = \frac{p(d | \theta)p(\theta)}{p(d)}$

Likelihood-based Inference

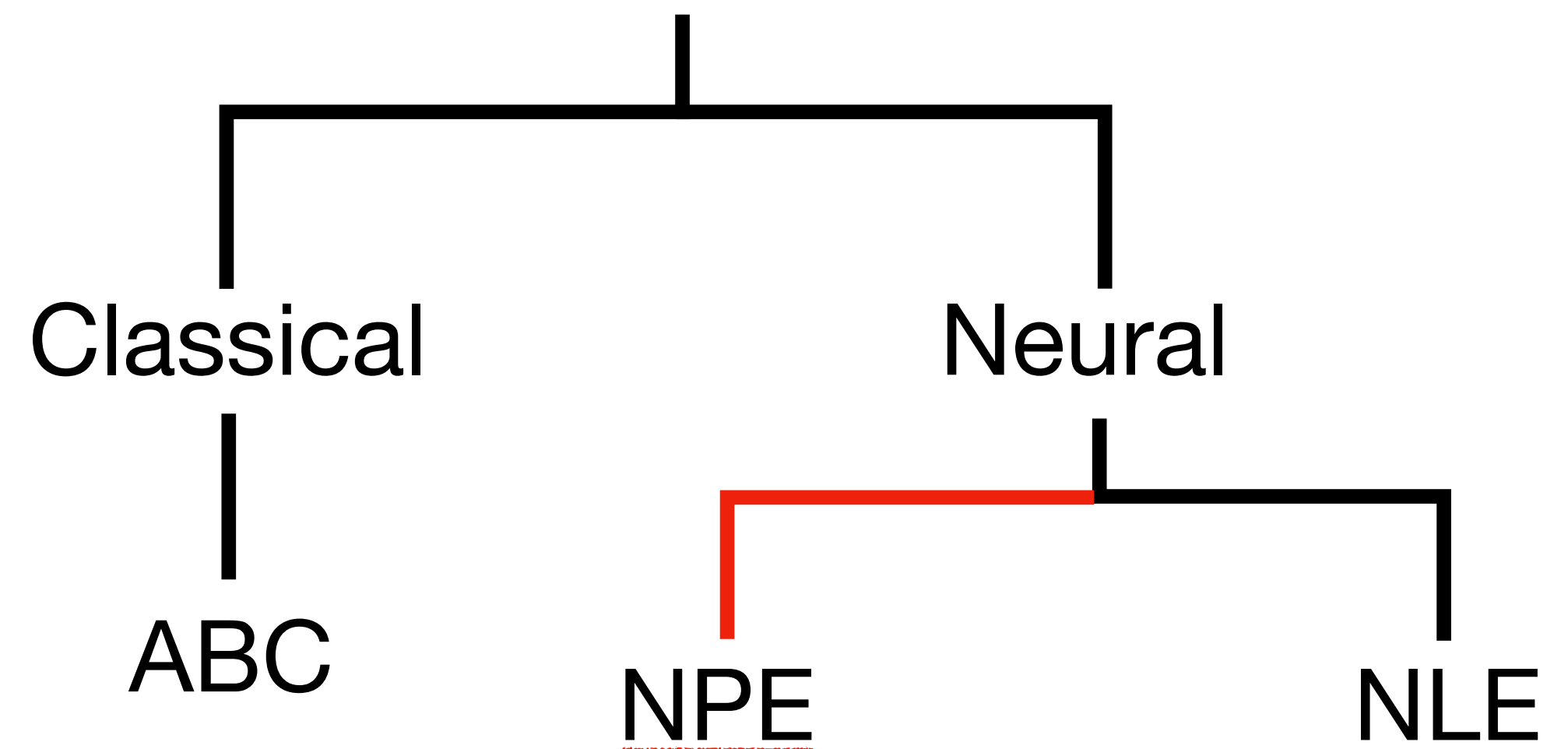
Evaluate $p(d | \theta)$



Requires a likelihood
Slow inference

Simulation-based Inference

Sample $d \sim p(d | \theta)$



Doesn't require a likelihood
Slow training
Fast Inference

Neural posterior estimation

Learning an amortised posterior estimation $q_{\phi}(\theta | d)$

Forward

Trains with the goal:

$$q_{\phi}(\theta | d) \approx p(\theta | d) \text{ w.r.t the weights } \phi$$

Which translates to:

$$\begin{aligned} \phi^* &= \underset{\phi}{\operatorname{argmin}} \mathbb{E}_{d \sim p(d)} D_{KL}(p(\theta | d) | q_{\phi}(\theta | d)) \\ &= \underset{\phi}{\operatorname{argmin}} - \mathbb{E}_{\theta \sim p(\theta)} \mathbb{E}_{d \sim p(d | \theta)} [\log q_{\phi}(\theta | d)] \end{aligned}$$

$$D_{KL}(p(x) | q(x)) = \int_{-\infty}^{\infty} p(x) \ln \frac{p(x)}{q(x)} dx = \begin{cases} 0 & \text{if } p(x) = q(x) \\]0, \infty[& \text{if } p(x) \neq q(x) \end{cases}$$

