Reconstructing Neutron Star Equation of State from Observational Data via Automatic Differentiation

Speaker: **Lingxiao Wang** (FIAS & GU)

work with: Shriya Soma, Shzuhe Shi, Horst Stoecker, **Kai Zhou**

arXiv:2201.01756

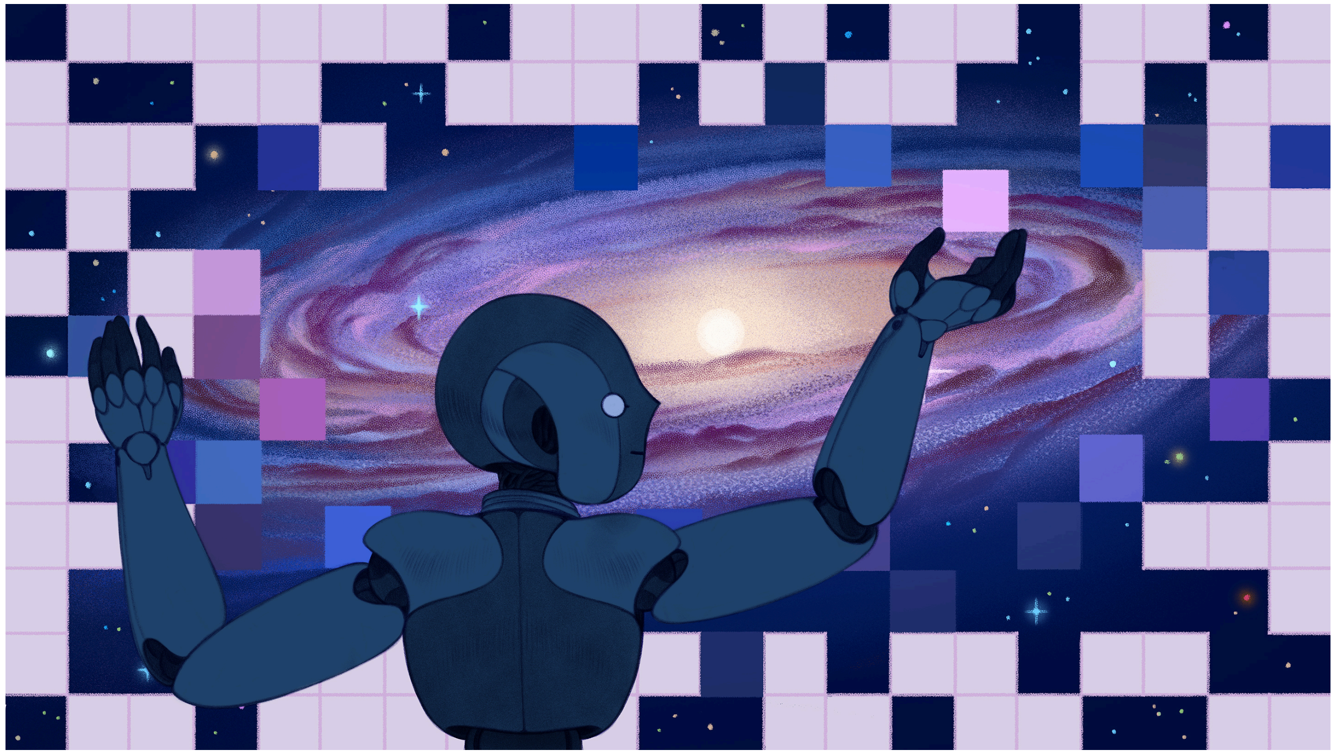
23 June, 2022 ECT* Workshop

FIAS Frankfurt Institute for Advanced Studies

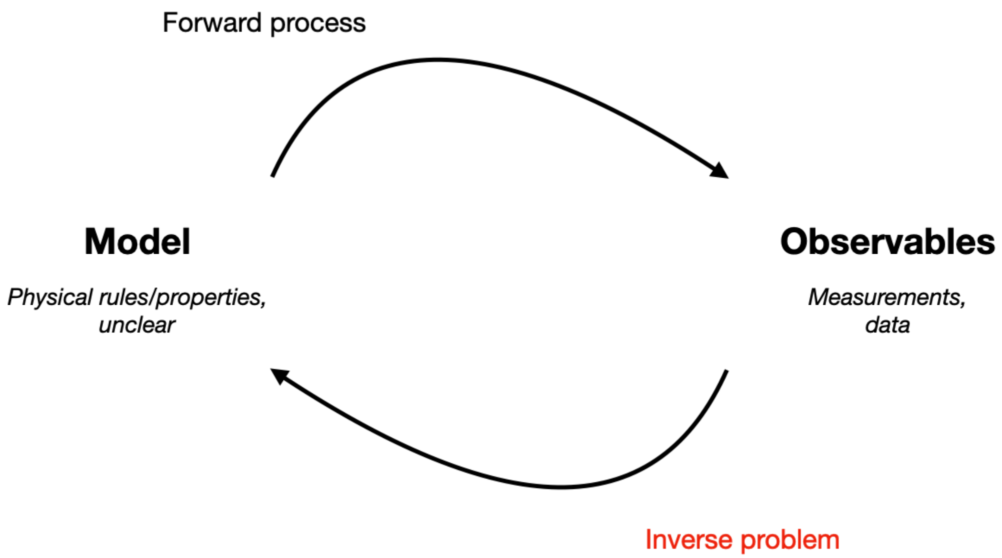
Goethe Universität Frankfurt am Main

Outline

- Machine Learning Physics
- Reconstructing EoS
  - Inverse problem
  - AD framework
  - Neural Network representations
  - Results
- Summary



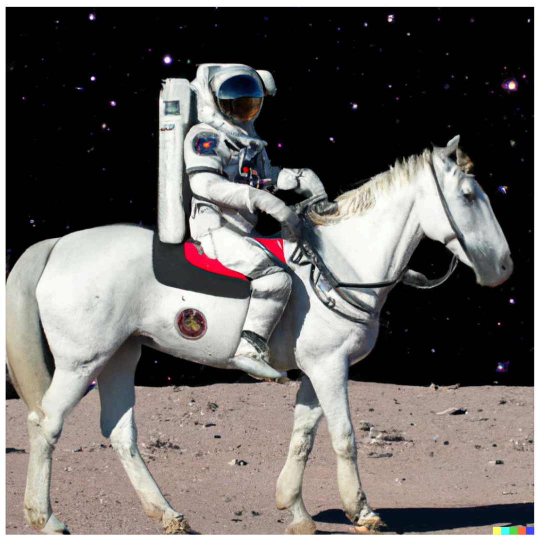
Rachel Suggs for Quanta Magazine

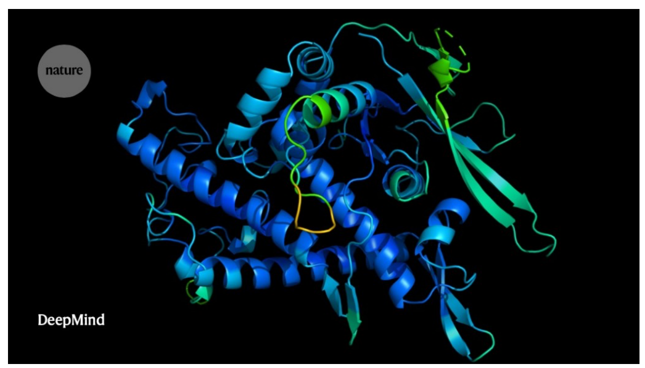


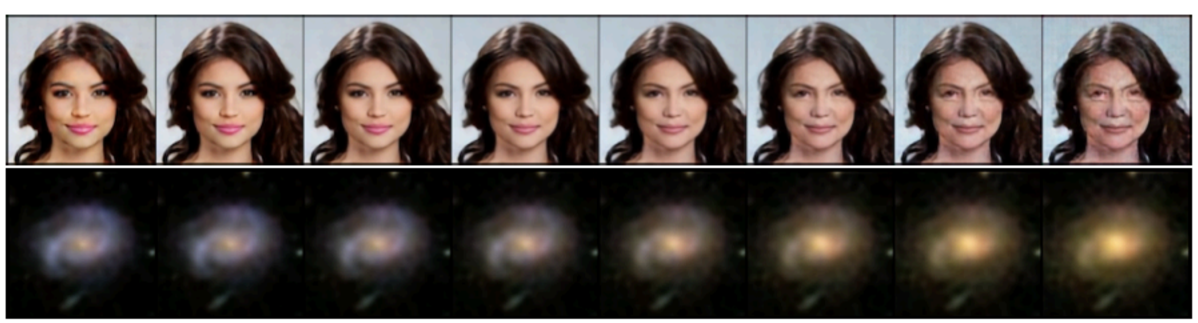
Machine Learning in

(Astro)Physics

DALL-E 2





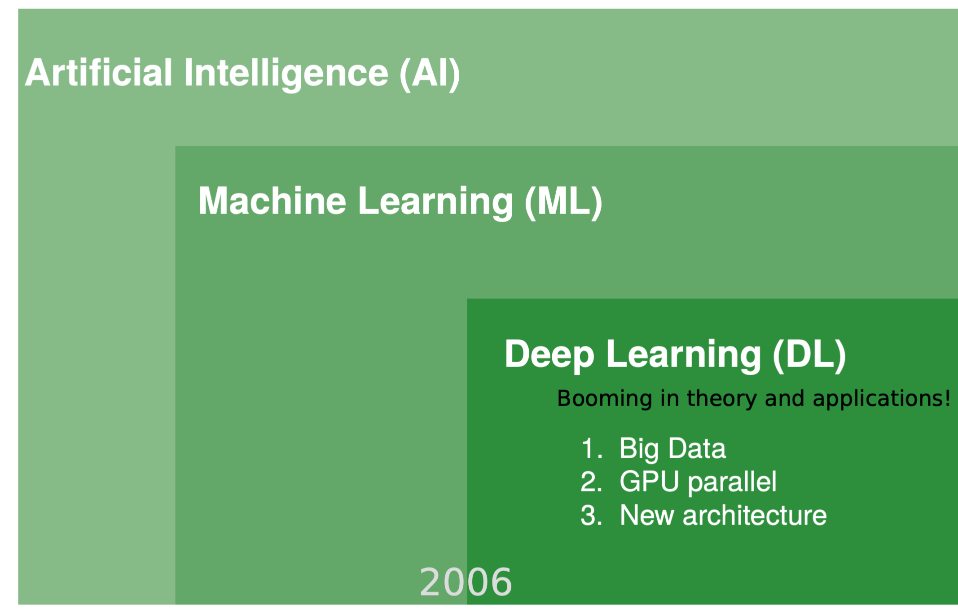


Astronomy & Astrophysics 616 (2018): L16

Big Data + Deep Models

↓ GPU

Successful Deep Learning!



Geoffrey Hinton

Data-driven ML

No explicit physical rules, but physical data

Machine Learning in

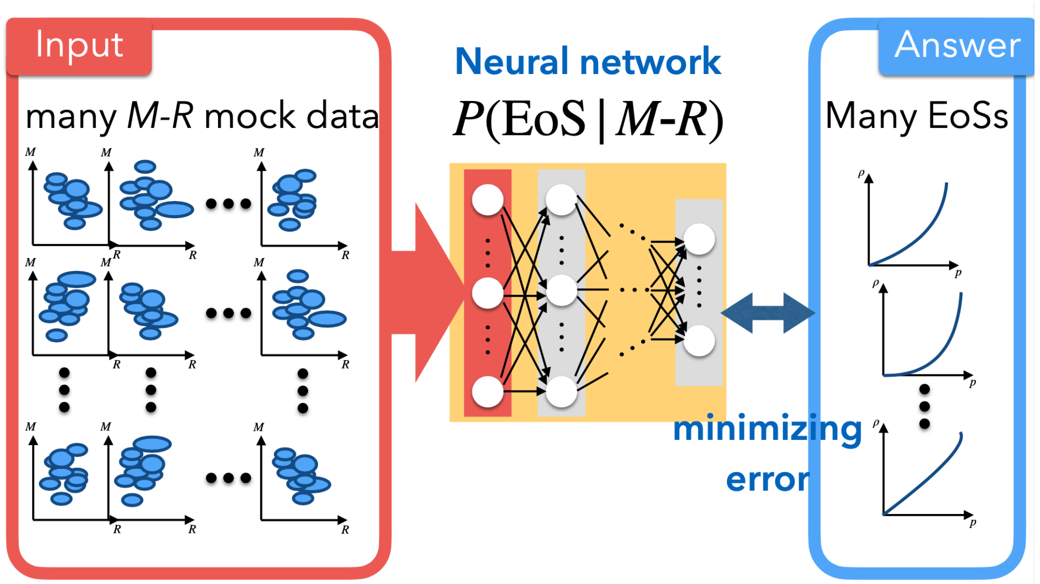
(Astro)Physics

Big Data + Deep Models

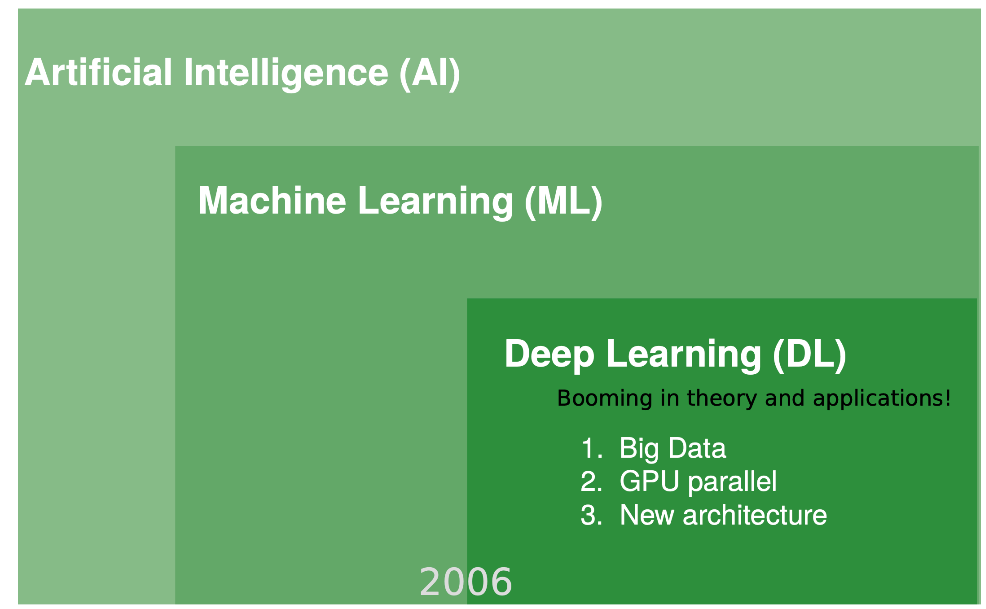
↓ GPU

Successful Deep Learning!

Astronomy & Astrophysics 616 (2018): L16



@Fujimoto&Fukushima



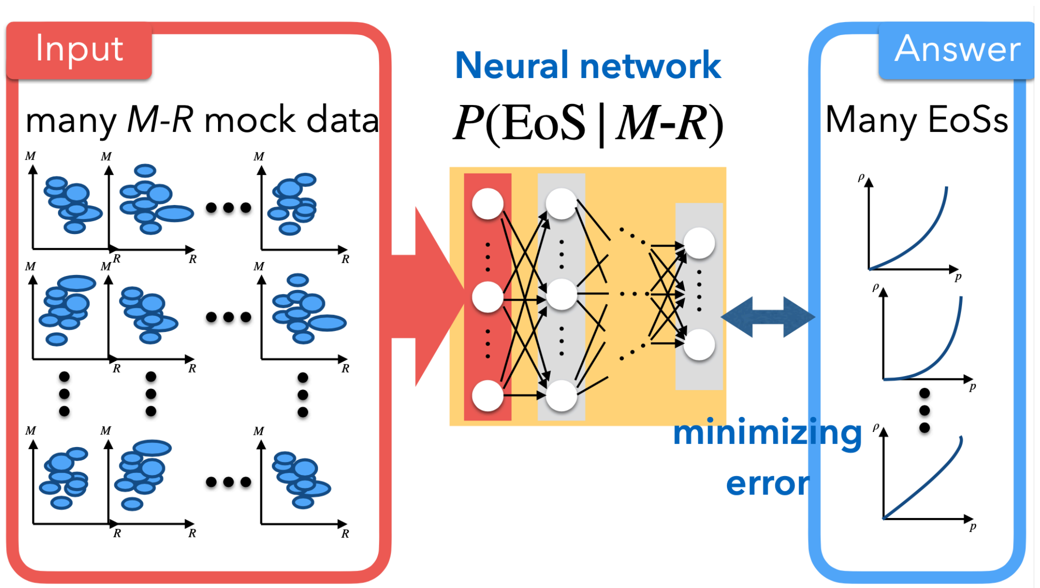
Geoffrey Hinton

Data-driven ML

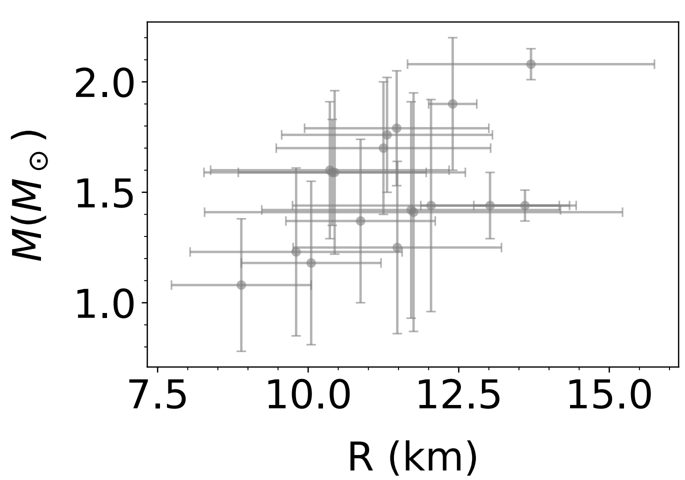
No explicit physical rules, but physical data

Machine Learning in

(Astro)Physics



@Fujimoto&Fukushima



Data-driven ML

No explicit physical rules, but physical data

What if data is few, noisy and unlabelled?

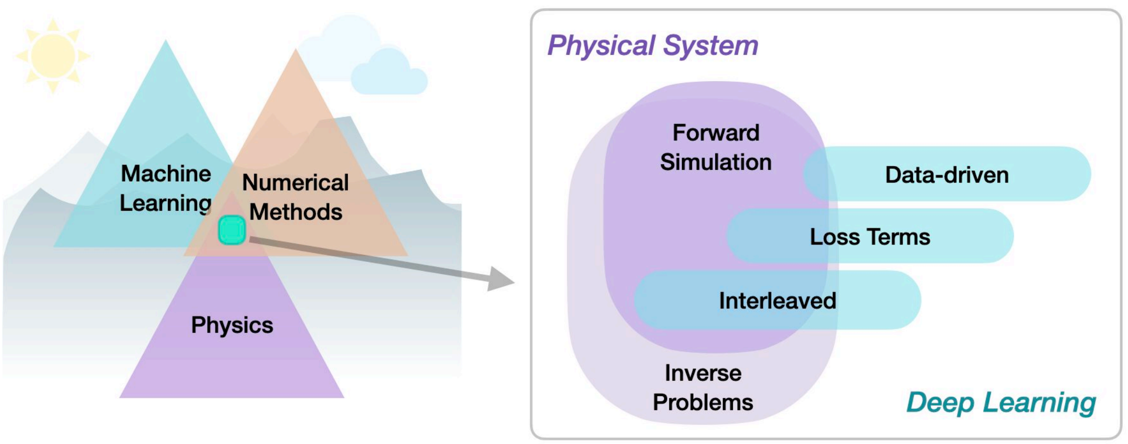
Physics-driven ML

Physical rules as differentiable modules in learning

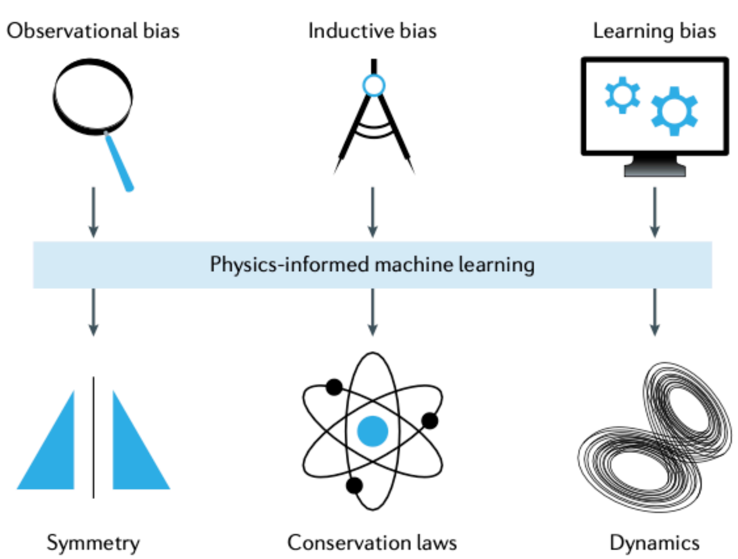
Demand for big-data

Physics in

Machine learning



Physics-based Deep Learning (PBDL)



Karniadakis GE, Kevrekidis IG, Lu L, Perdikaris P, Wang S, Yang L. *Physics-informed machine learning*. Nat Rev Phys 2021;3:422–40.

Demand for physics

Data-driven ML

No explicit physical rules, but physical data

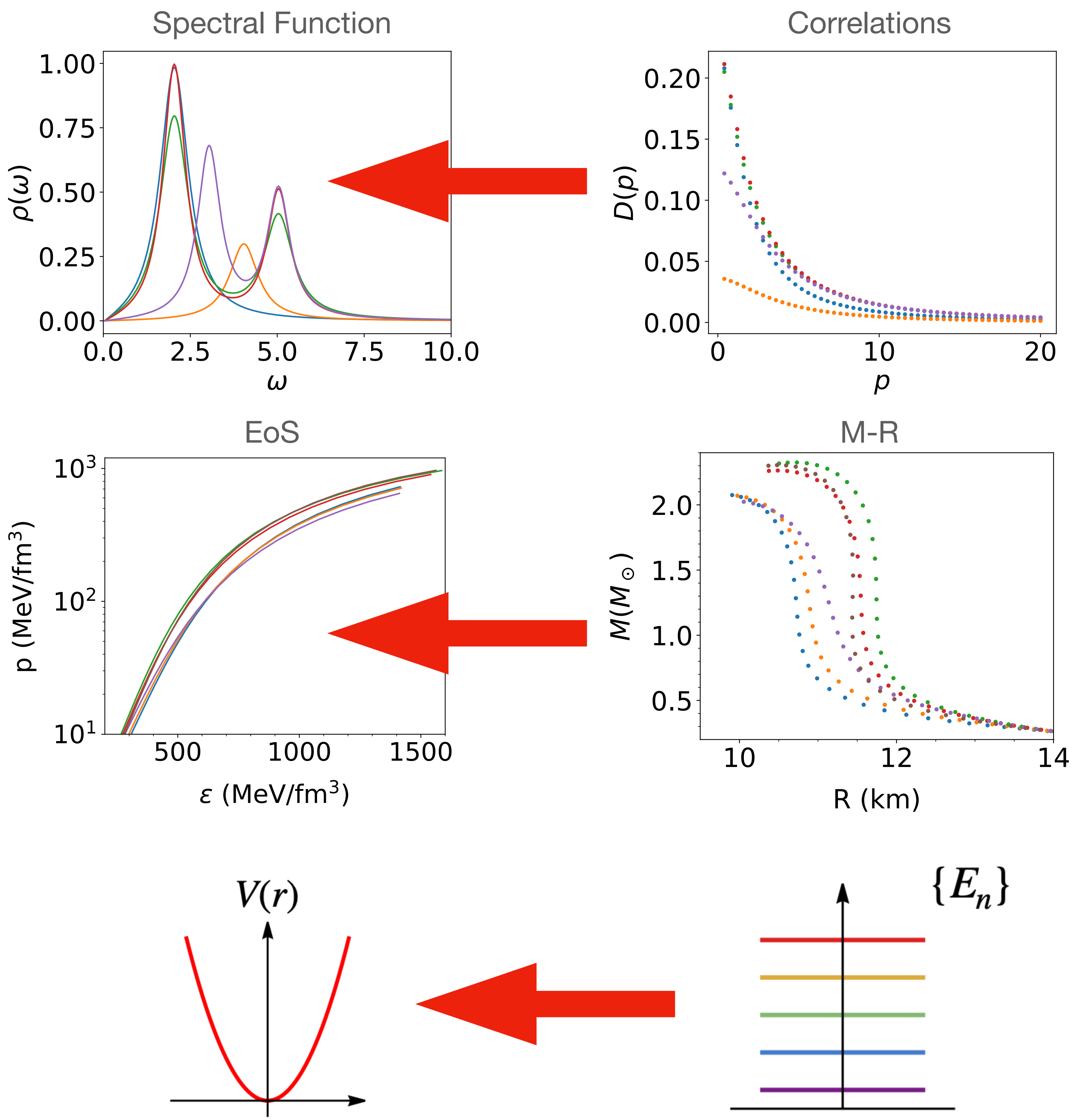
Physics-informed ML (Physical losses)

Physical rules as constraints in training

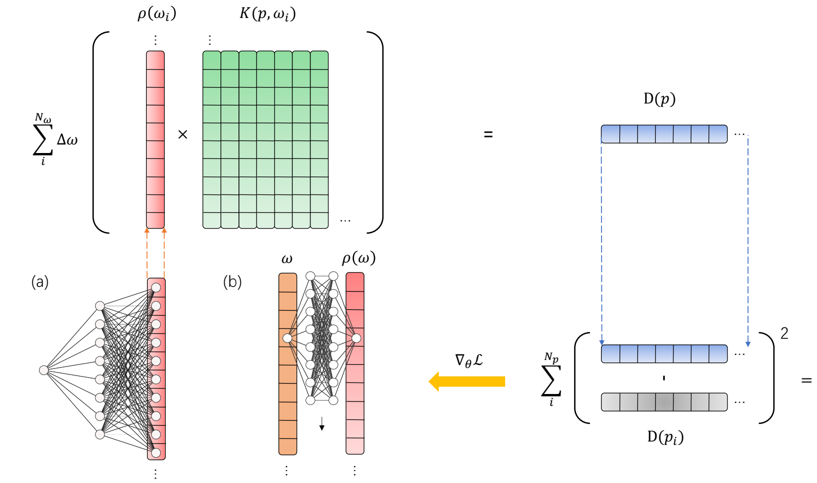
Physics-driven ML

Physical equations as differentiable modules in learning

Inverse Problems

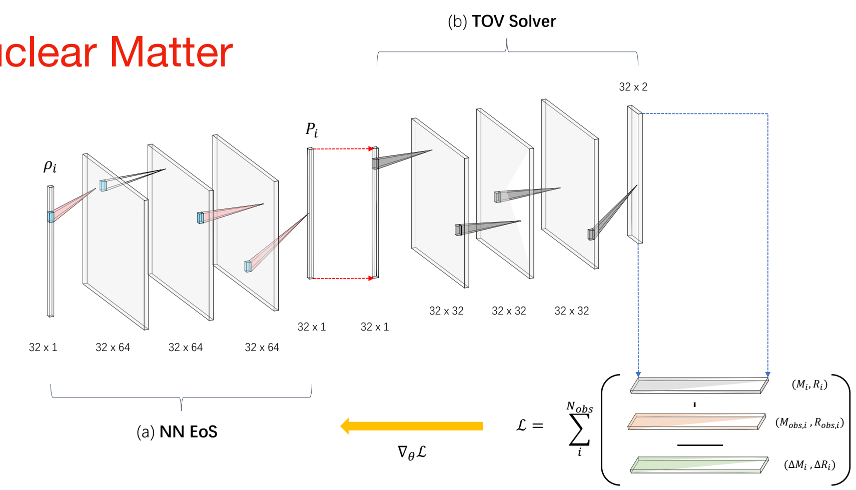


Lattice QCD



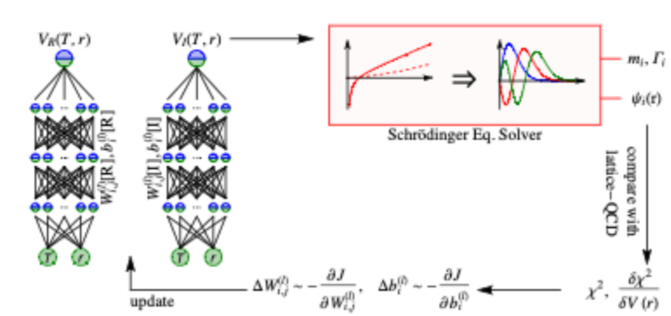
Lingxiao Wang, Shuzhe Shi and Kai Zhou,
arXiv: 2111.147760; 2201.02564; NeurIPS 2021 ML4PS

Nuclear Matter



Shriya Soma, Lingxiao Wang, *et al.* arXiv 2201.01756

Quantum Mechanics



Shuzhe Shi, *et al.* PhysRevD.105.014017

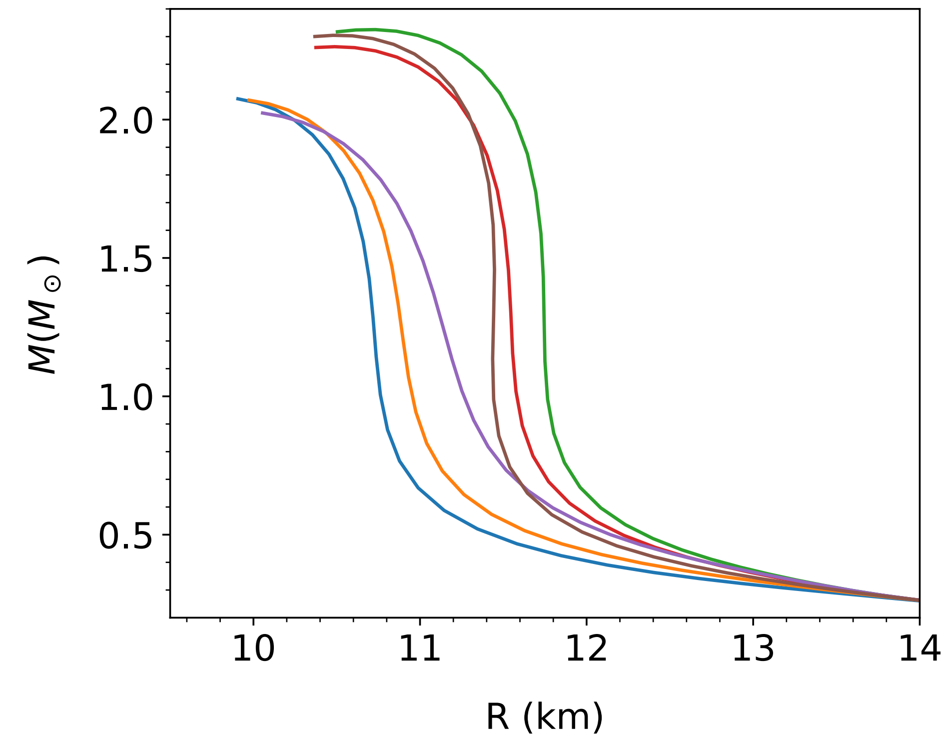
Inverse Problem

Learning EoS from M-R

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

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

Forward process

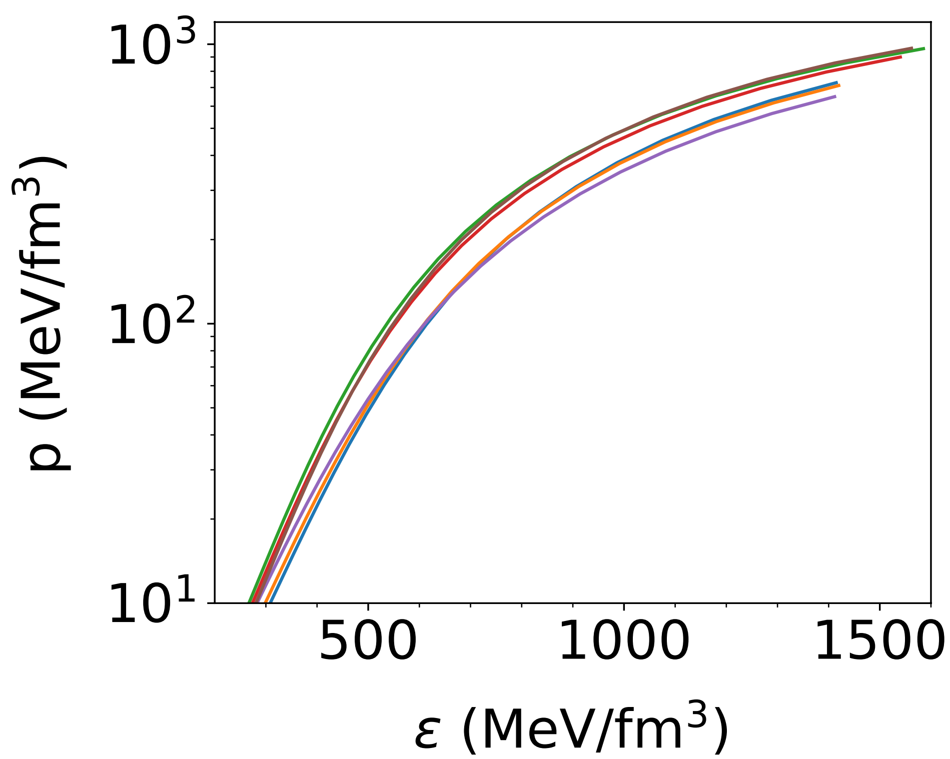


Model

Physical rules/properties, unclear

Observables

Measurements, data



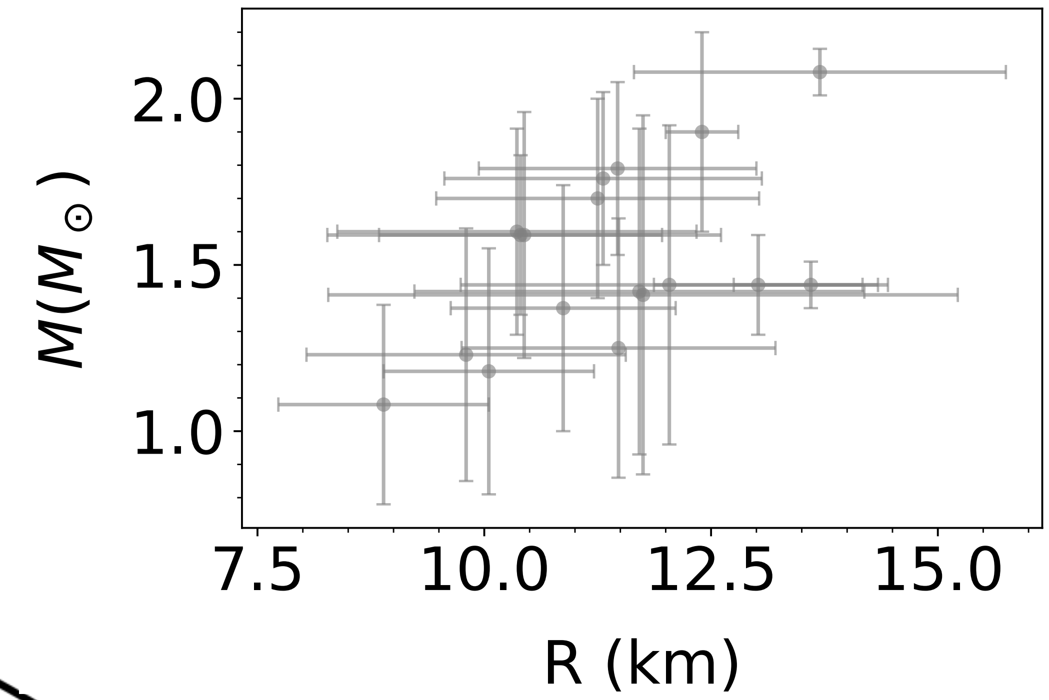
Inverse problem

If the whole M(R) is known,
it's well-solved problem.

L. Lindblom, A.J., 398, 569 (1992).

Inverse Problem

Learning EoS from M-R



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

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

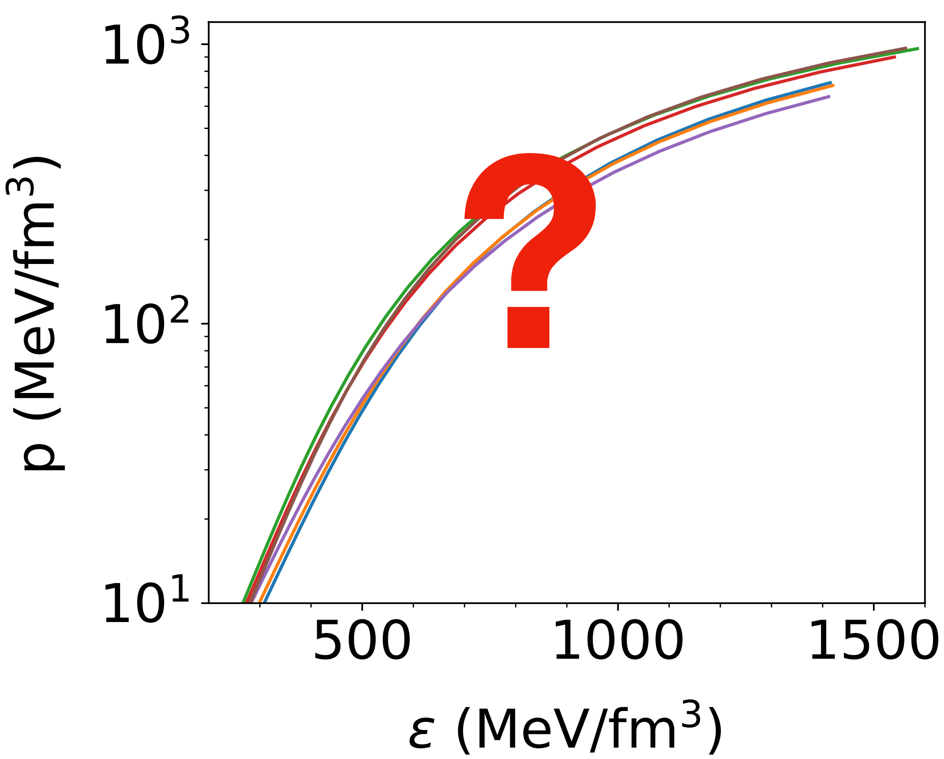
Forward process

Model

Physical rules/properties, unclear

Observables

Measurements, data



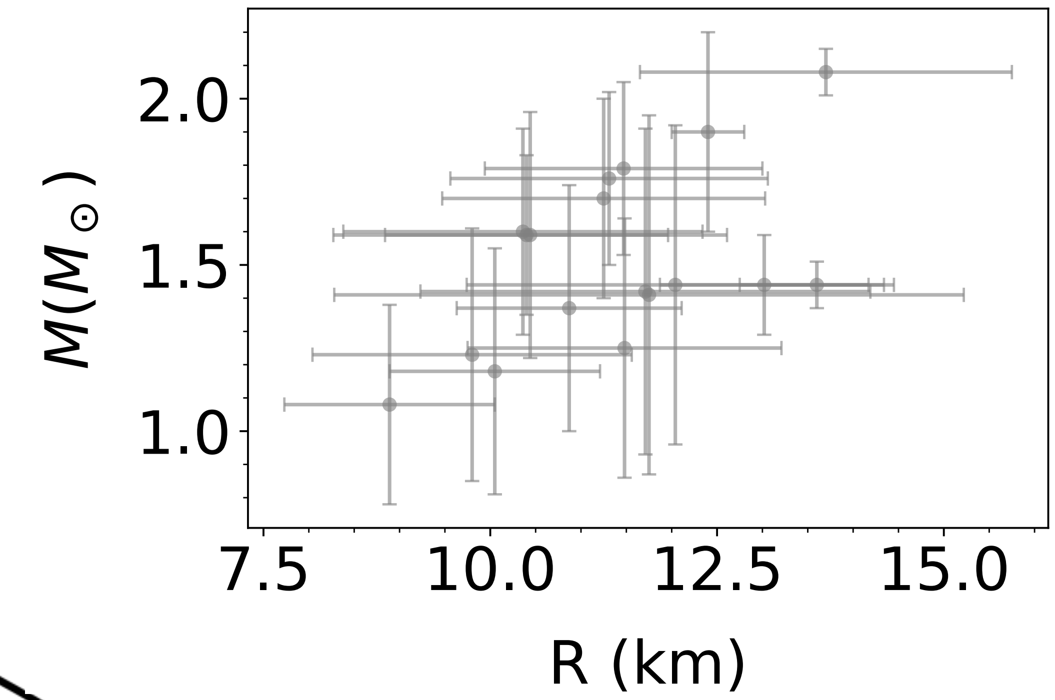
Parameterized EoS (e.g., Meta-modeling EoS) + Bayesian Inference + MCMC
many works in this workshop…

Machine Learning the Direct Inverse Mapping

[1] Y. Fujimoto, K. Fukushima, and K. Murase, Phys. Rev. D 98, 023019 (2018).
[2] P. Landry and R. Essick, Phys. Rev. D 99, 084049 (2019).
[3] Y. Fujimoto, K. Fukushima, and K. Murase, Phys. Rev. D 101, 054016 (2020).
[4] F. Morawski and M. Bejger, A&A 642, A78 (2020).
[5] P. Landry, R. Essick, and K. Chatziioannou, Phys. Rev. D 101, 123007 (2020).
[6] Y. Fujimoto, K. Fukushima, and K. Murase, JHEP 2021, 273 (2021).
[7] M. Ferreira and C. Providência, J. Cosmol. Astropart. Phys. 2021, 011 (2021).
…

Inverse Problem

Learning EoS from M-R



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

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

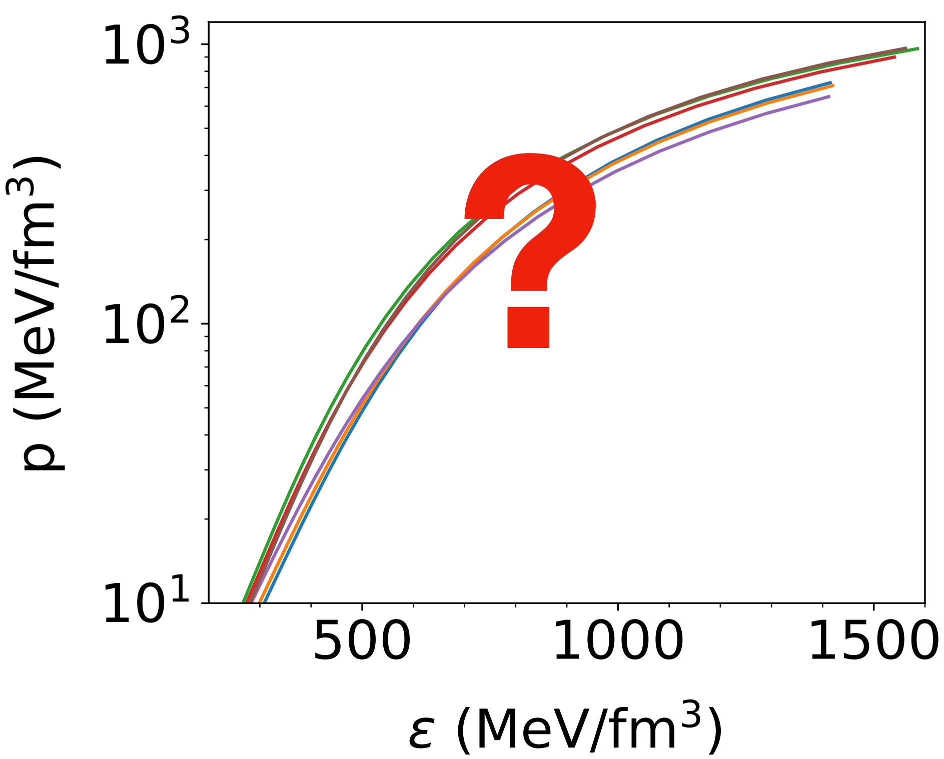
Forward process

Model

Physical rules/properties, unclear

Observables

Measurements, data



Unbiased EoS + Maximize Likelihood + High Effcient Optimization

Framework

AD

- **Automatic differentiation (AD)**
  - It refers to a general way of taking a program which **computes a value**, and **automatically constructing a procedure for computing derivatives of that value.**
  - Example

How we compute the derivatives of logistic least squares regression in a neural net,

ω weights, b bias, $\sigma(z)$ activation function
x input, y output, t target, $\mathscr{L}$ loss function.

Chain rule: $h'(x) = f'(g(x))g'(x).$

Computing the loss:

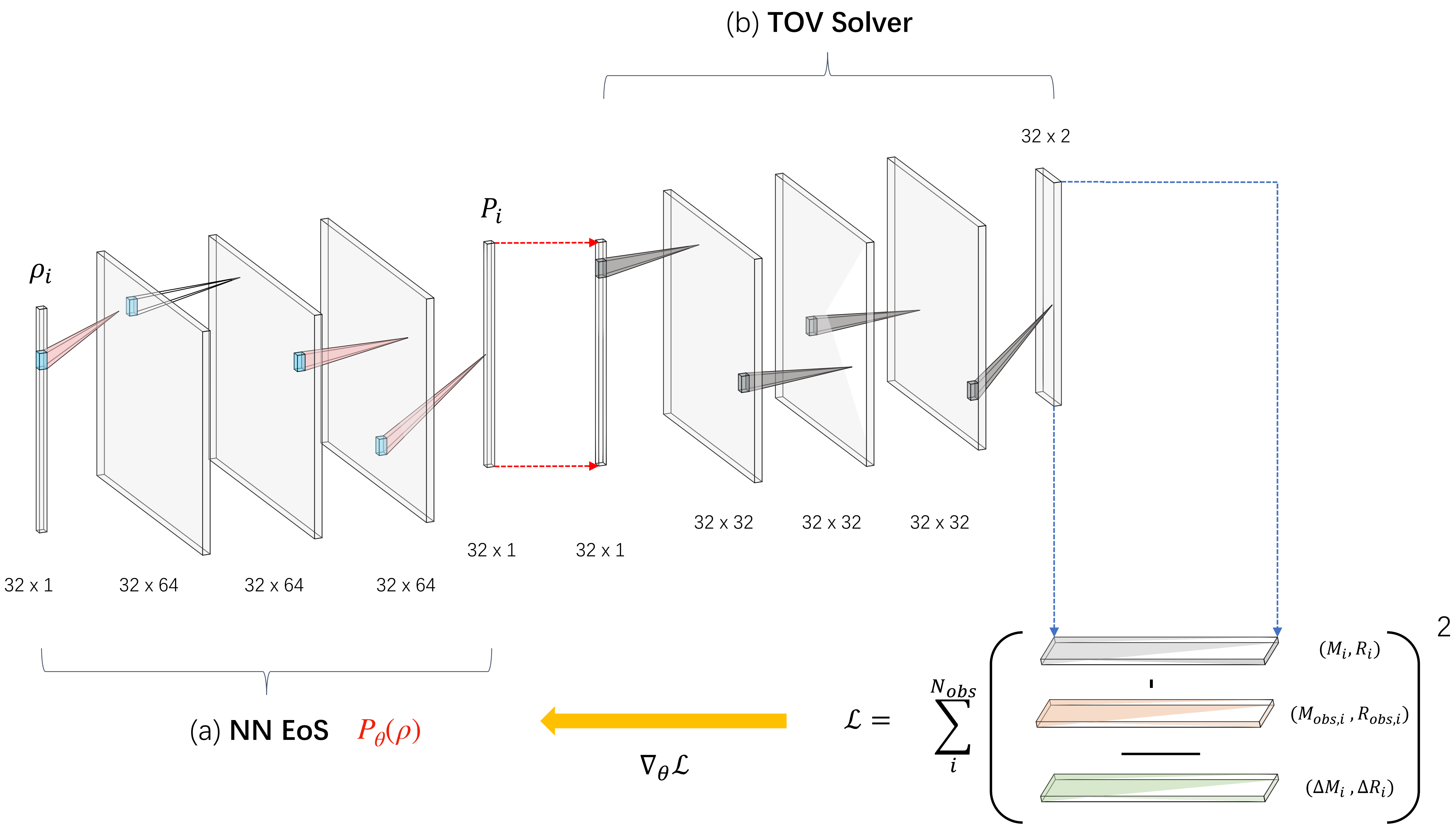
$$z = wx + b$$
$$y = \sigma(z)$$
$$\mathscr{L} = \frac{1}{2}(y - t)^2$$

Computing the derivatives:

$$\overline{\mathscr{L}} = 1$$
$$\bar{y} = y - t$$
$$\bar{z} = \bar{y}\sigma'(z)$$
$$\bar{w} = \bar{z}x$$
$$\bar{b} = \bar{z}$$

AD Framework

Reconstruct EoS

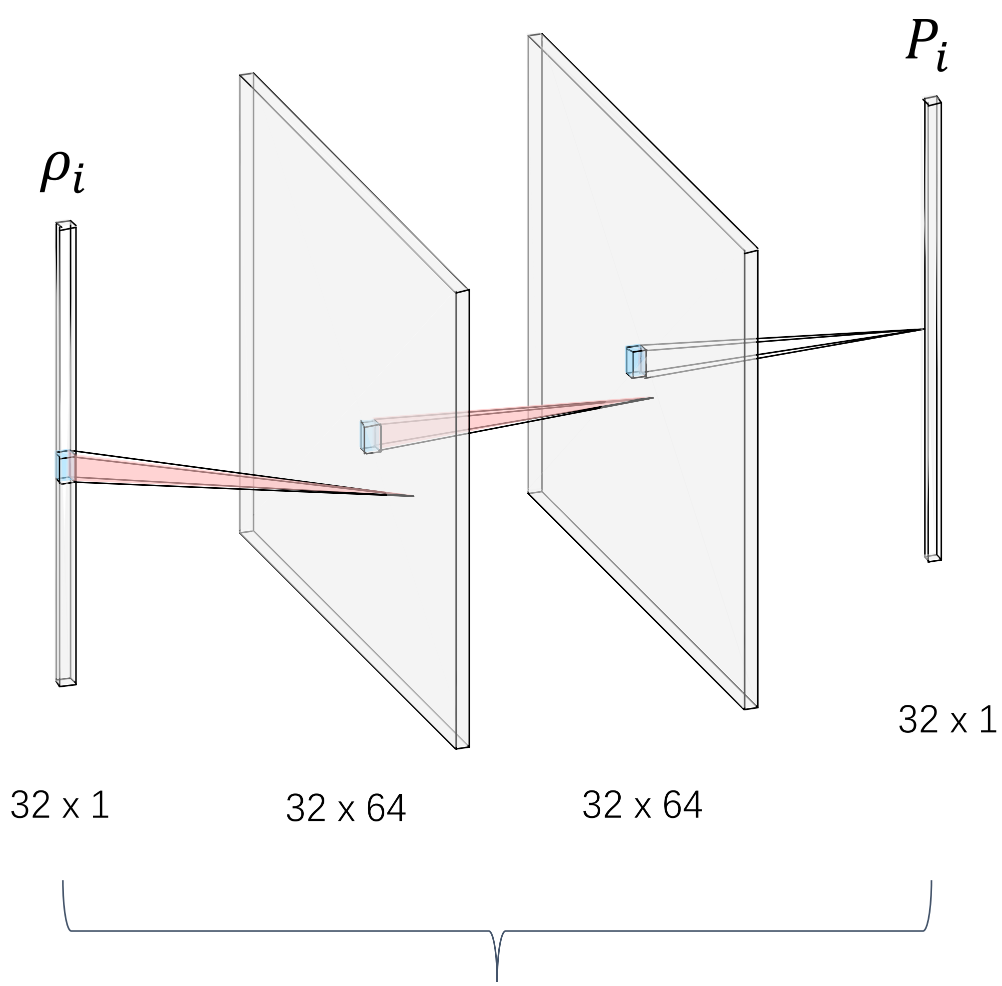


Framework

Neural Network EoS

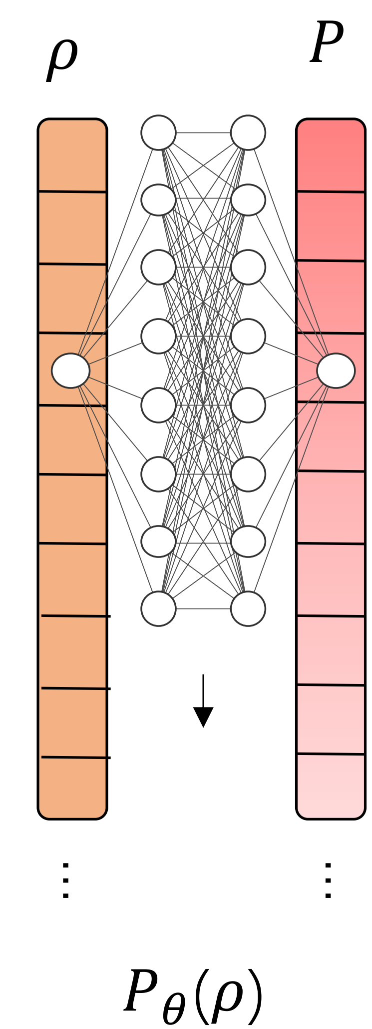
A **feed-forward network** with a single hidden layer containing a **finite number of neurons** can approximate arbitrary continuous functions.

Universal approximation theorem (1989,1991)



NN EoS

A Trainable Neural Network



$\{\theta\}$: weights and bias of the neural network

Size of $\{\theta\}$ = 4353

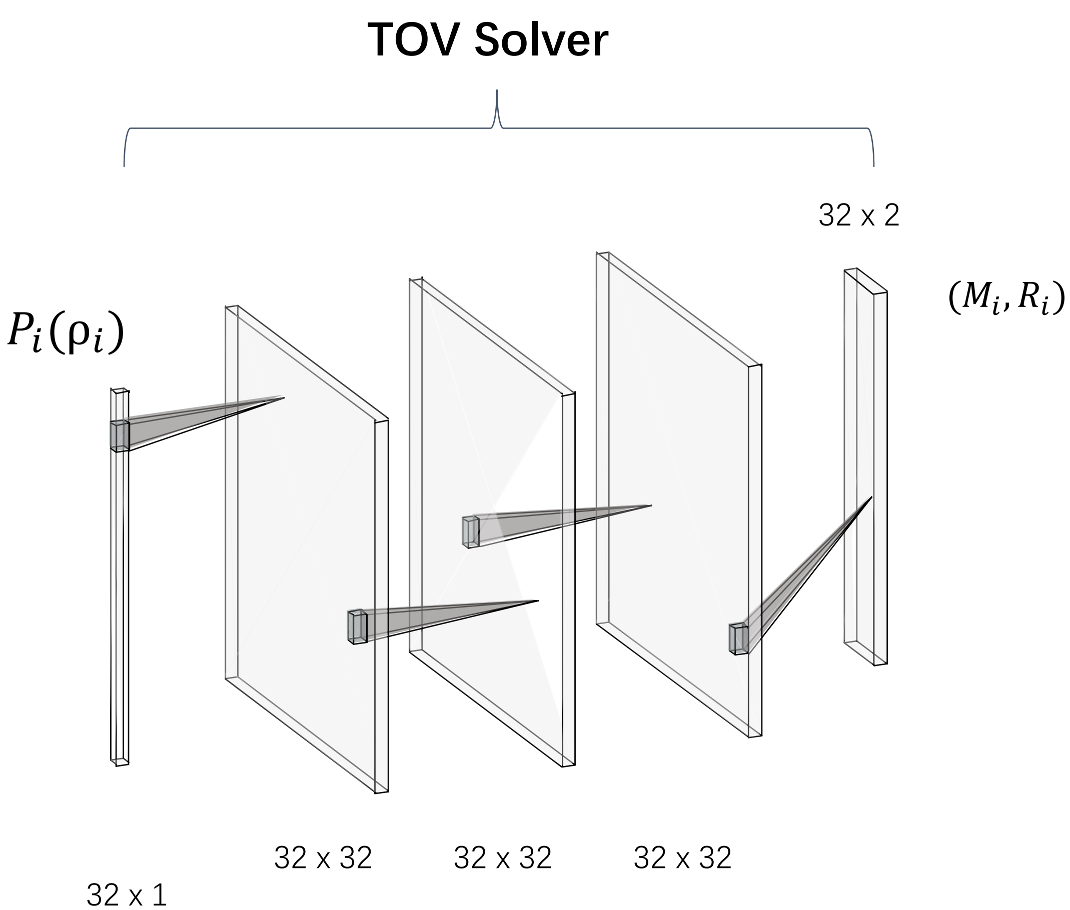
Framework

Neural Network TOV Solver

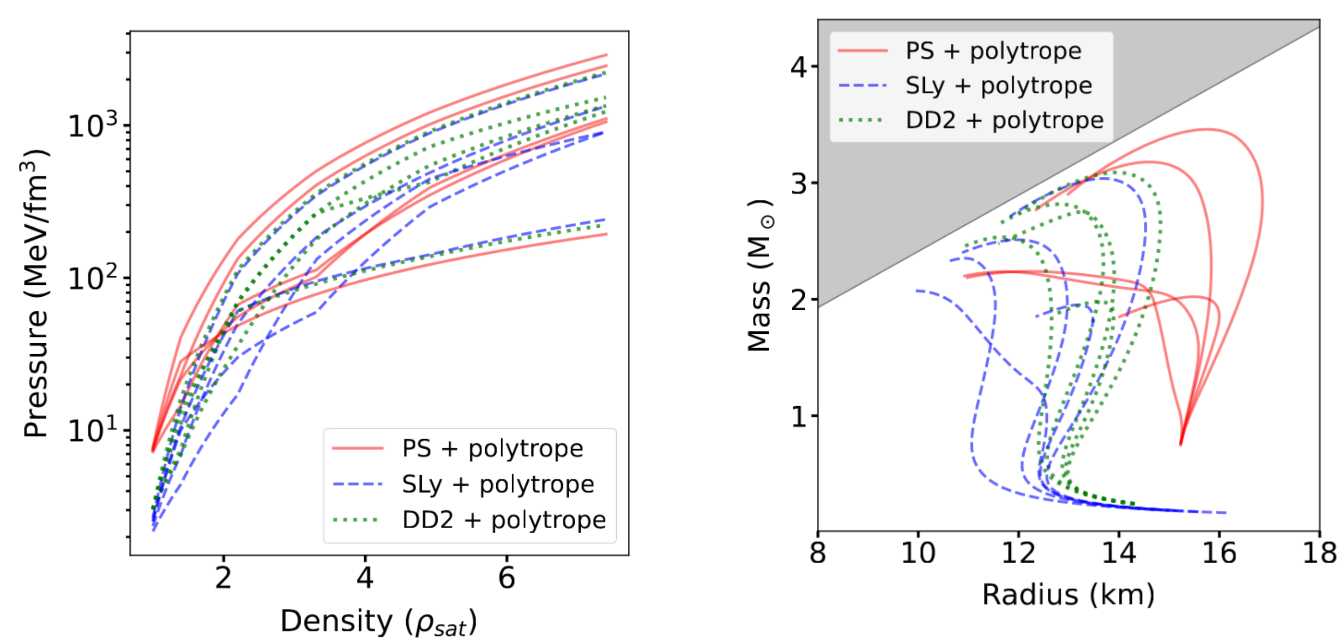
TOV

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

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



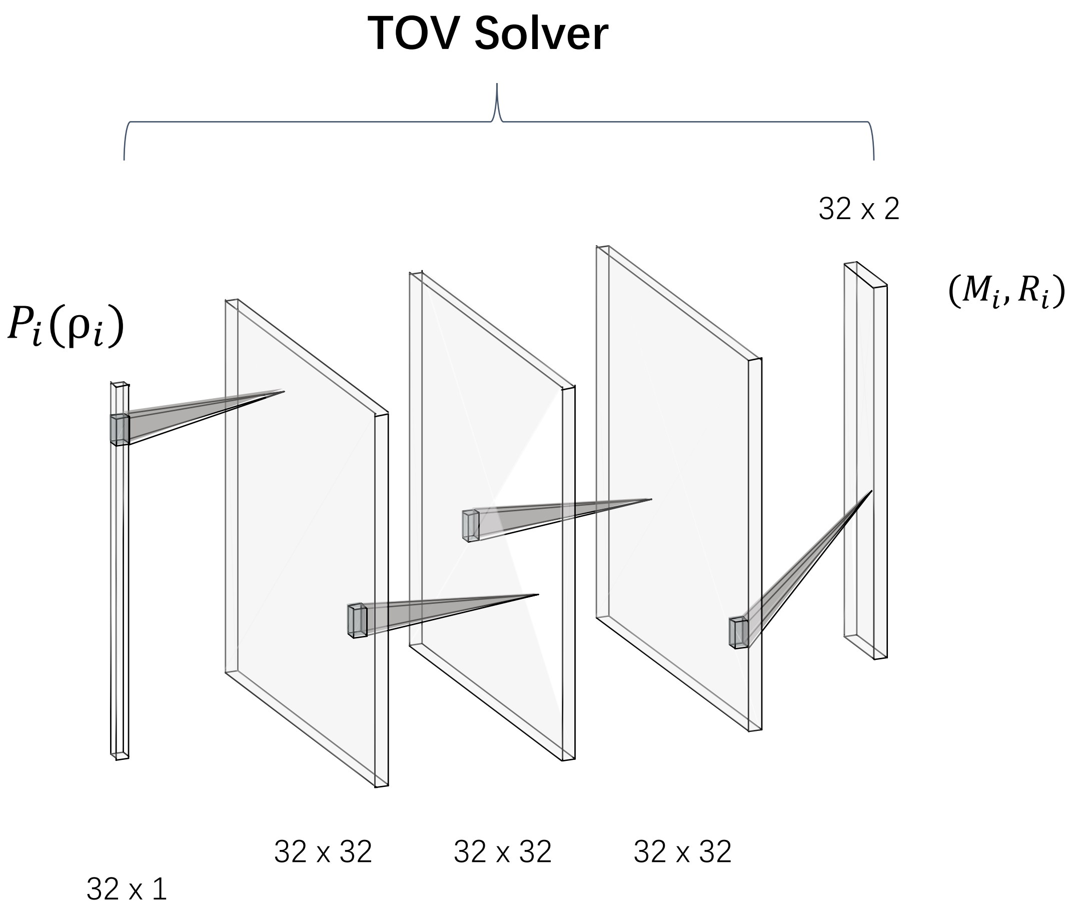
A well-trained Neural Network



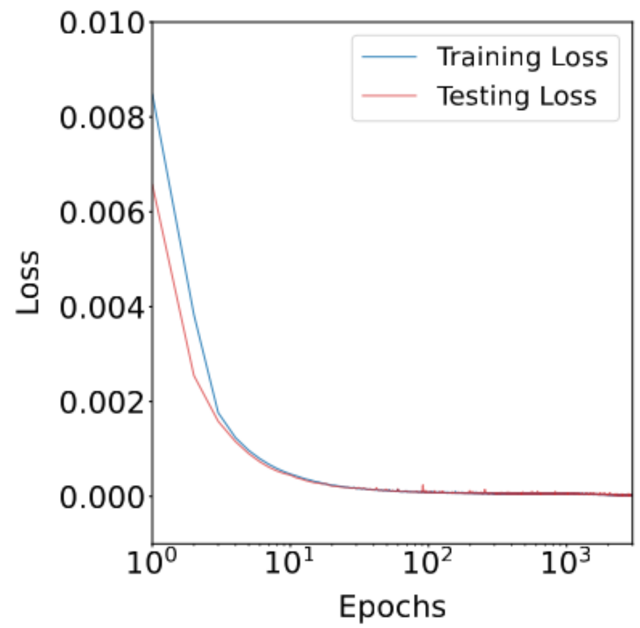
100,000 polytropic EoS functions for each low density model

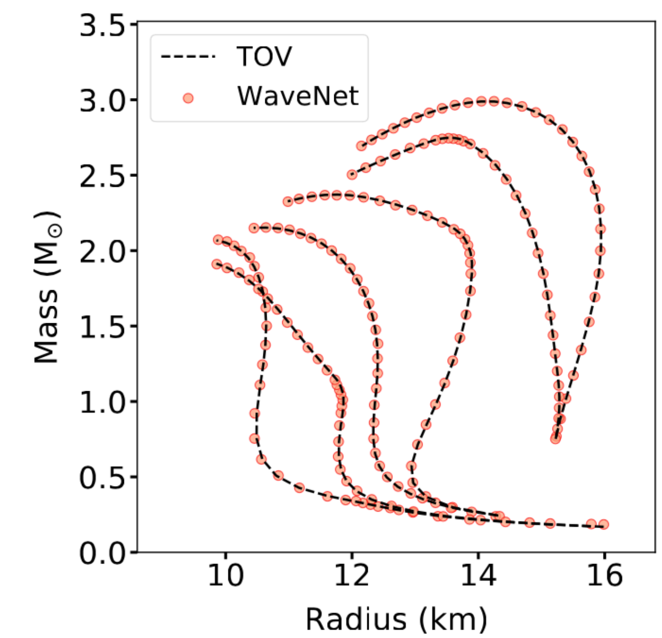
Framework

Neural Network TOV Solver



A well-trained Neural Network



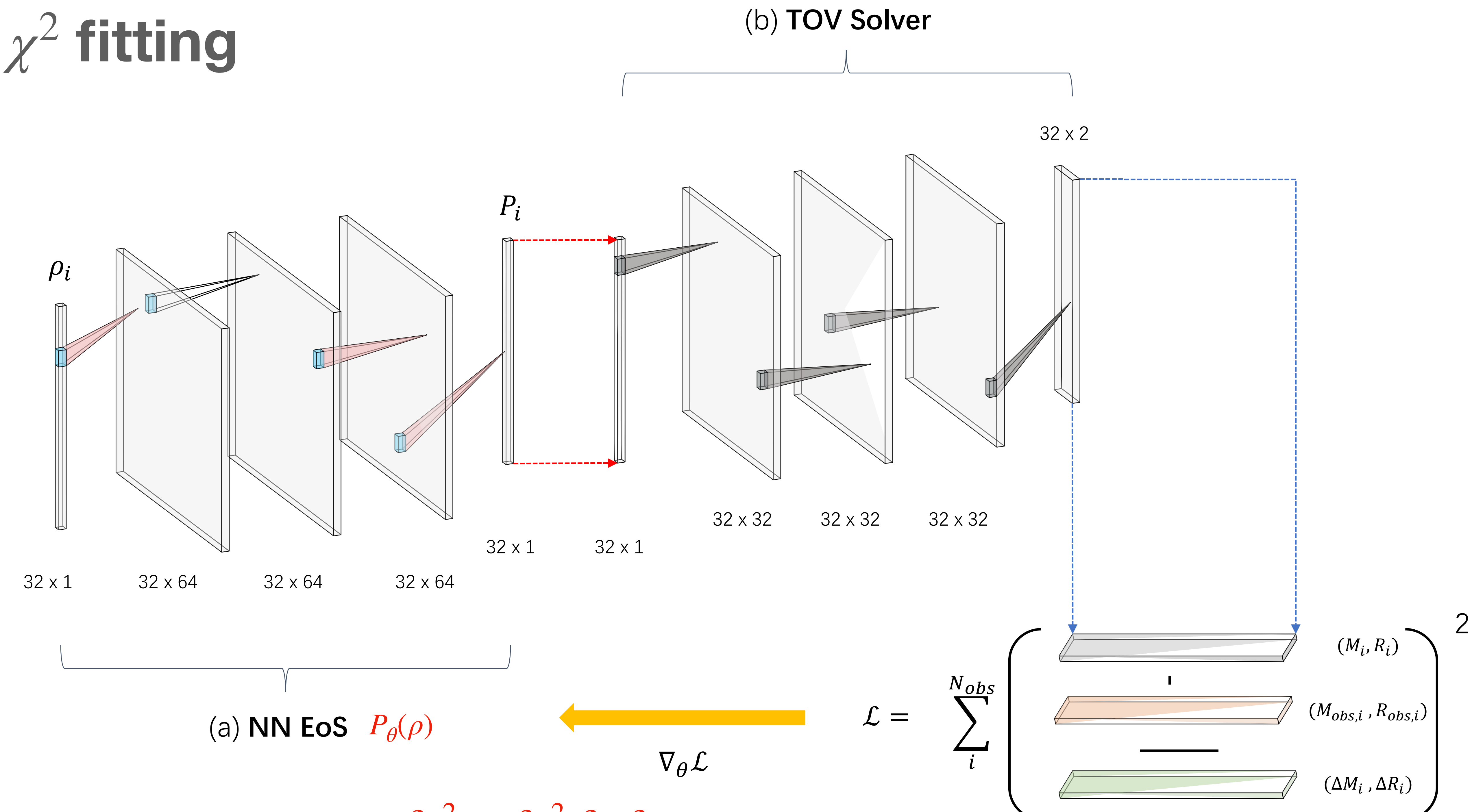


Comparison of the performance of different Neural Networks for solving the TOV equations.

N_ρ	NN	$\mathcal{R}^2$	MSE ($\times 10^{-5}$)	Parameters (#)	Epochs ($\times 10^3$)	Time ($\times 10^3 sec$)
128	CNN	0.9999	1.743	170,176	3.5	7.35
	FCN	0.9999	1.052	70,304	15	4.91
	LSTM	0.9998	0.741	347,416	3	32.5
	WaveNet	0.9998	3.003	296,706	3	64.7
32	CNN	0.9999	3.019	58,912	3.5	2.15
	FCN	0.9999	1.179	23,936	15	2.79
	LSTM	0.9999	0.814	74,904	3	4.03
	WaveNet	0.9999	3.047	18,882	3	10.7

Framework

χ^2 fitting



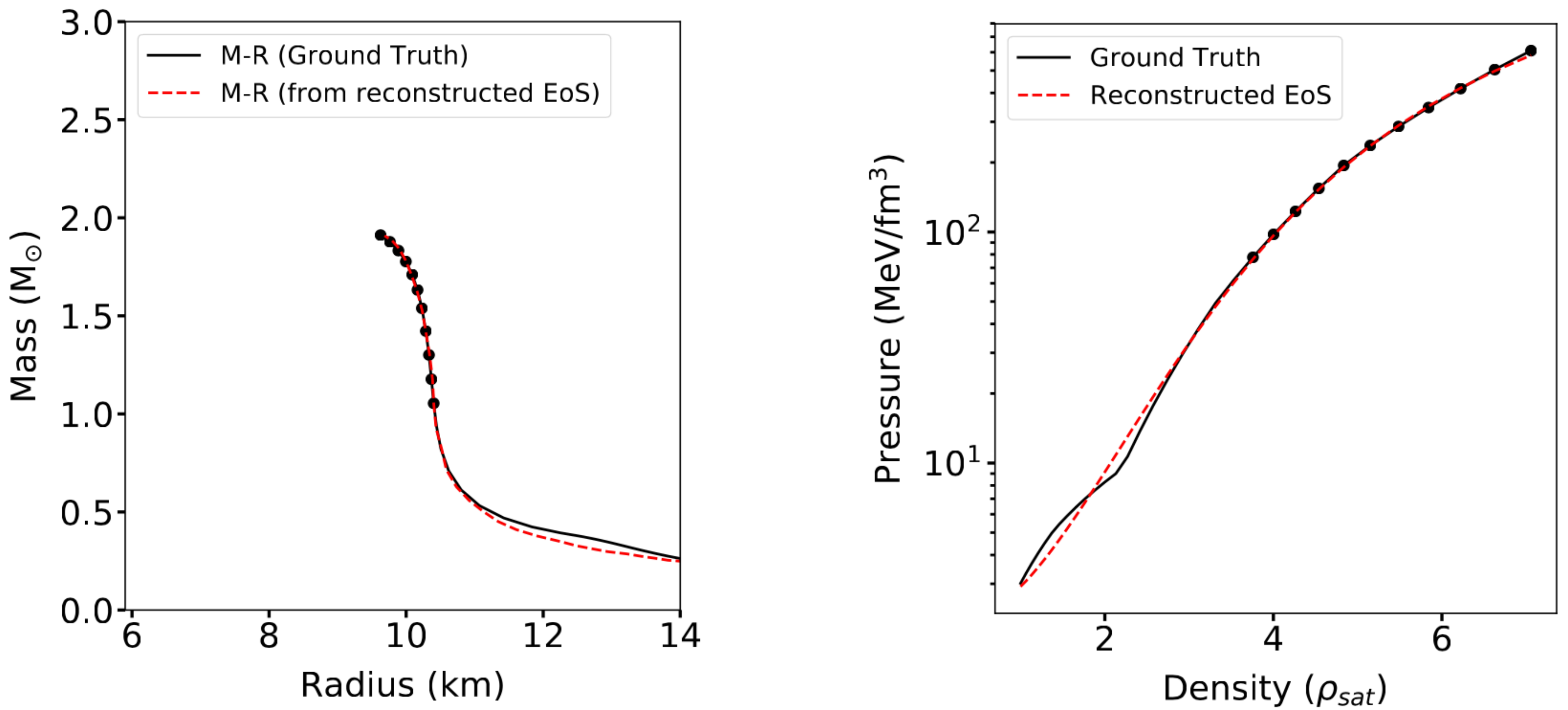
$$\frac{\delta \chi^2}{\delta \theta} = \frac{\delta \chi^2}{\delta \mathbf{z}} \frac{\delta \mathbf{z}}{\delta x} \frac{\delta x}{\delta \theta}$$

$\mathbf{z} = (M_i, R_i), x = P_i(\rho_i)$

$$L = \chi^2 = \sum_{i=1}^{N_{\rm obs}} \frac{\left(M_i - M_{{\rm obs},i}\right)^2}{\Delta M_i^2} + \frac{\left(R_i - R_{{\rm obs},i}\right)^2}{\Delta R_i^2}$$

Results

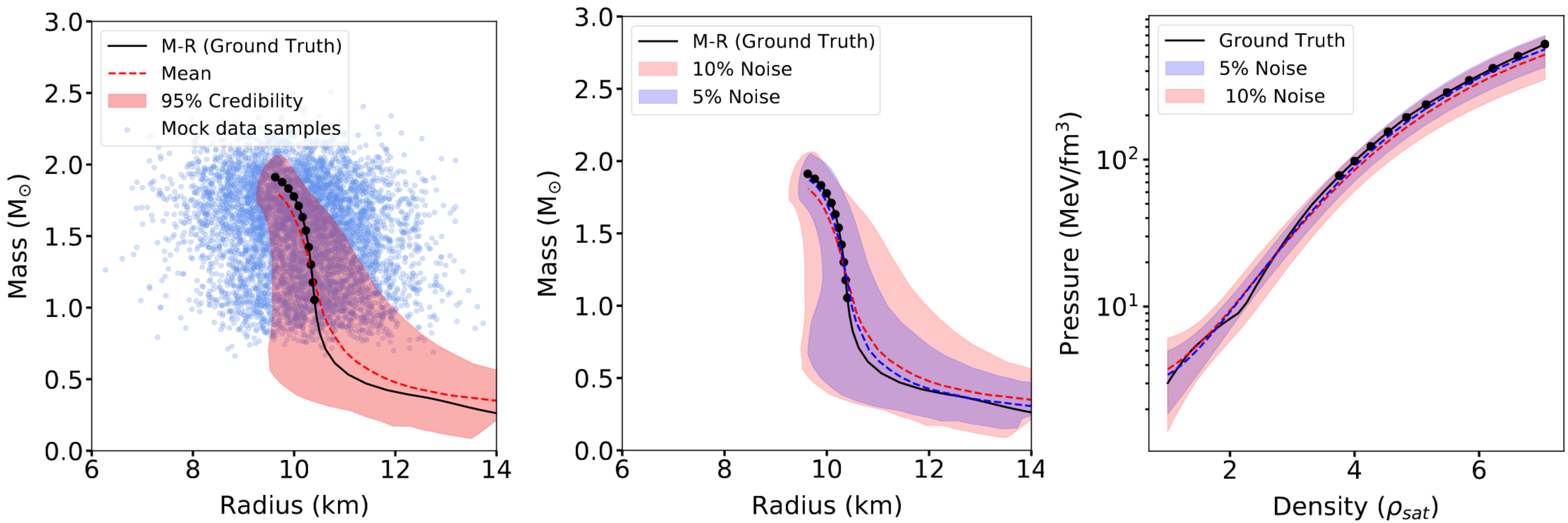
Test 1: mock data without noise



A reasonable agreement of the M-R curve from the reconstructed EoS (red dashed line) with the ground truth curve is depicted in the mass region $M > 1 M_\odot$,
with only 11 (M,R) pairs.

Results

Test 2: mock data with noise

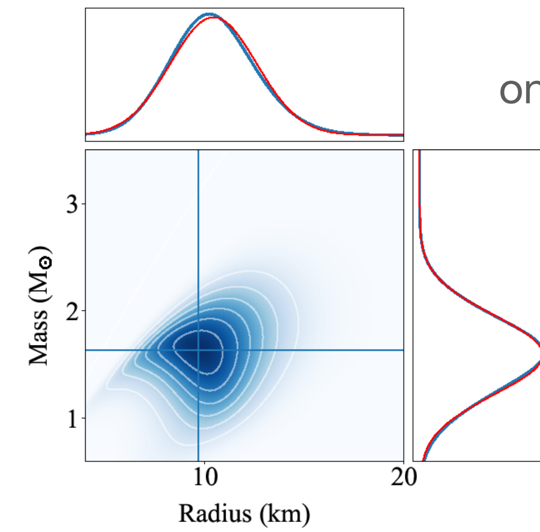


On mock data,
Noise(Mi) ~ $\mathcal{N}(0, 0.1M_i)$
Noise(Ri) ~ $\mathcal{N}(0, 0.1R_i)$

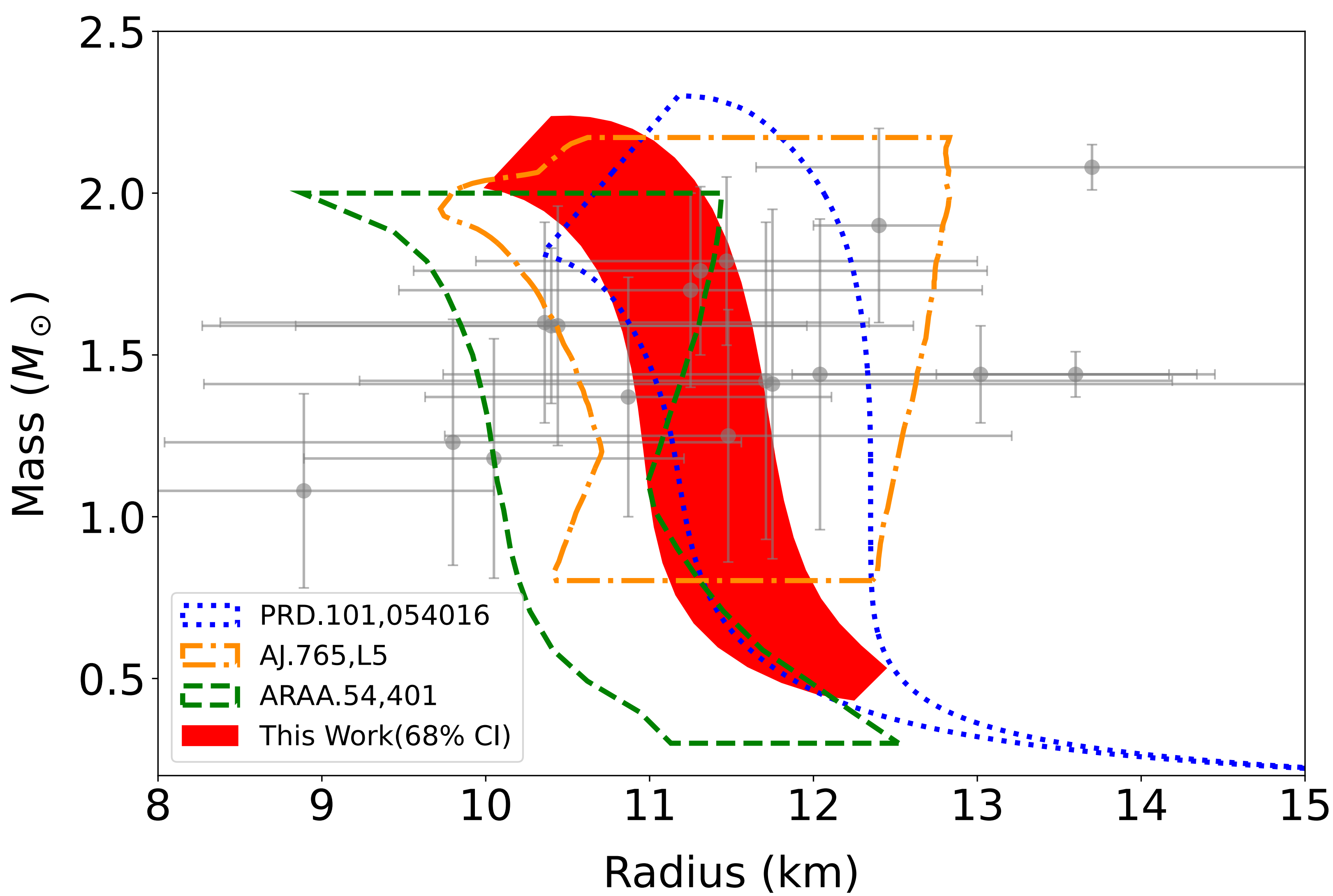
500 samples give us different reconstructed EOSs and M-R curves.
Gaussian fitting for each ρ_i or (M_i, R_i) to get confidence interval(CI).

Results

On real data: M-R



Gaussian Fitting
on the posterior table



Observable	Mass($M_\odot$)	Radius(km)
M13	1.42±0.49	11.71±2.48
M28	1.08±0.30	8.89±1.16
M30	1.44±0.48	12.04±2.30
NGC 6304	1.41±0.54	11.75±3.47
NGC 6397	1.25±0.39	11.48±1.73
ωCen	1.23±0.38	9.80±1.76
4U 1608-52	1.60±0.31	10.36±1.98
4U 1724-207	1.79±0.26	11.47±1.53
4U 1820-30	1.76±0.26	11.31±1.75
EXO 1745-248	1.59±0.24	10.40±1.56
KS 1731-260	1.59±0.37	10.44±2.17
SAX J1748.9-2021	1.70±0.30	11.25±1.78
X5	1.18±0.37	10.05±1.16
X7	1.37±0.37	10.87±1.24
4U 1702-429	1.90±0.30	12.40±0.40
PSR J0437–4715	1.44±0.07	13.60±0.85
PSR J0030+0451	1.44±0.15	13.02±1.15
PSR J0740+6620	2.08±0.07	13.70±2.05

18 (M_i, R_i) , batch size = 1000

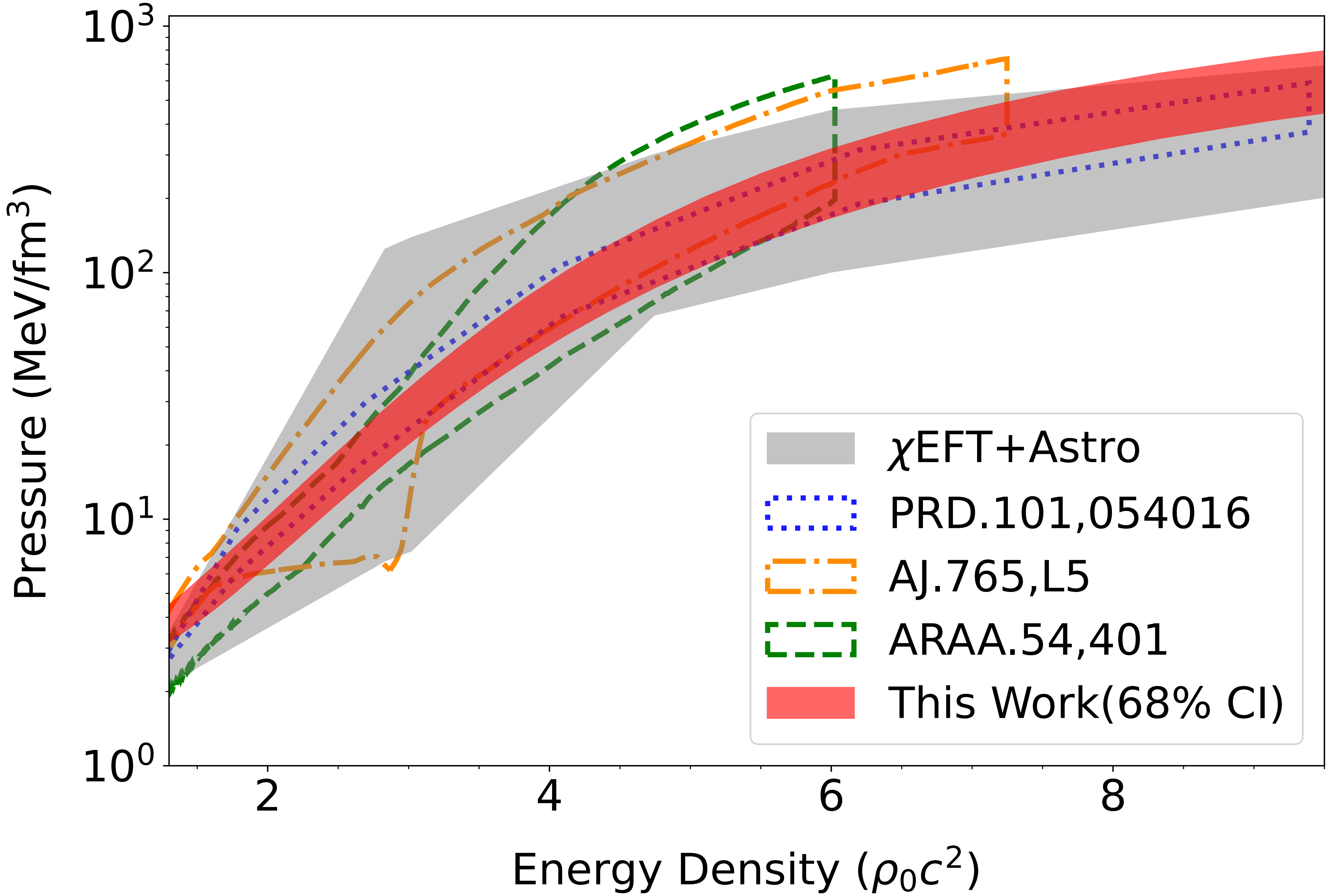
Constraints: causality, Maximum mass $\geq 1.9\, M_\odot$

Results

On real data: EoS

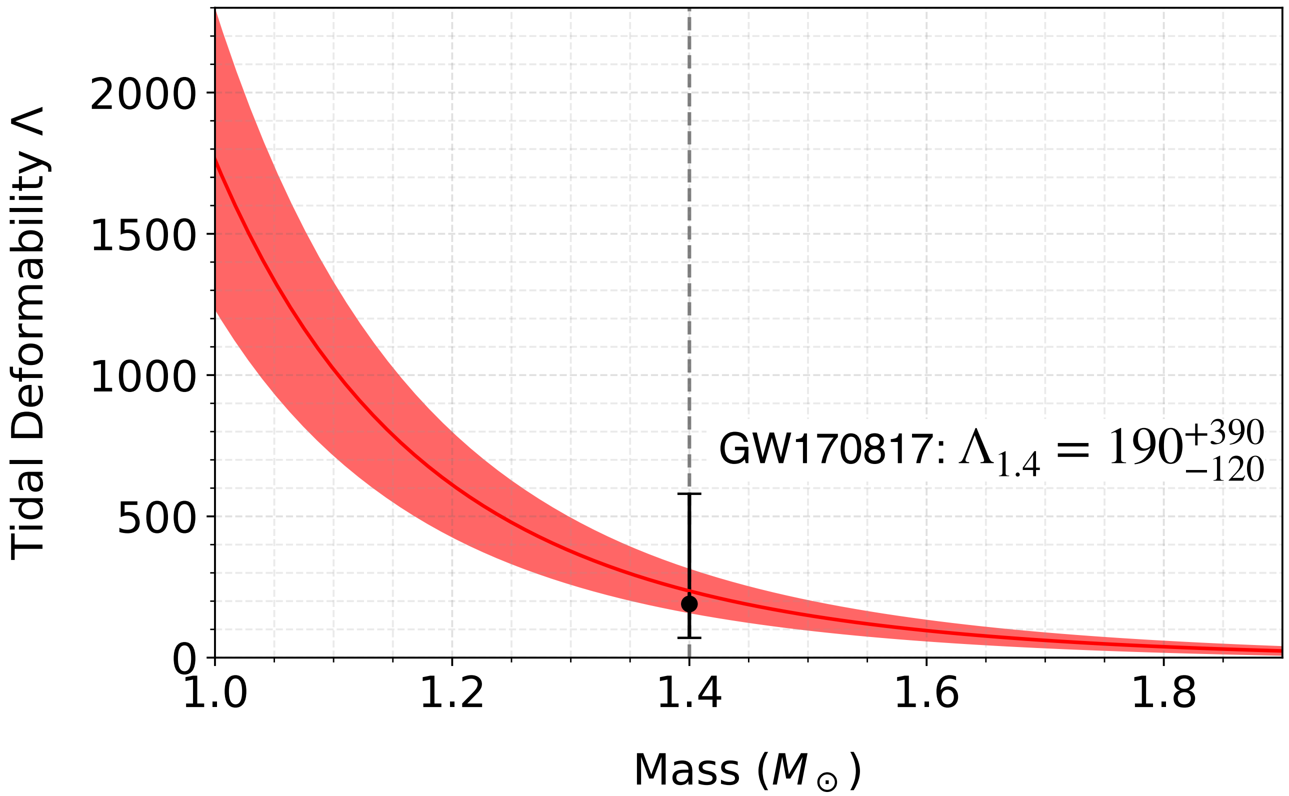
Blue dots: NN as direct inverse mapping

Yellow and Green dashed lines: Bayesian Approaches



Results

Others

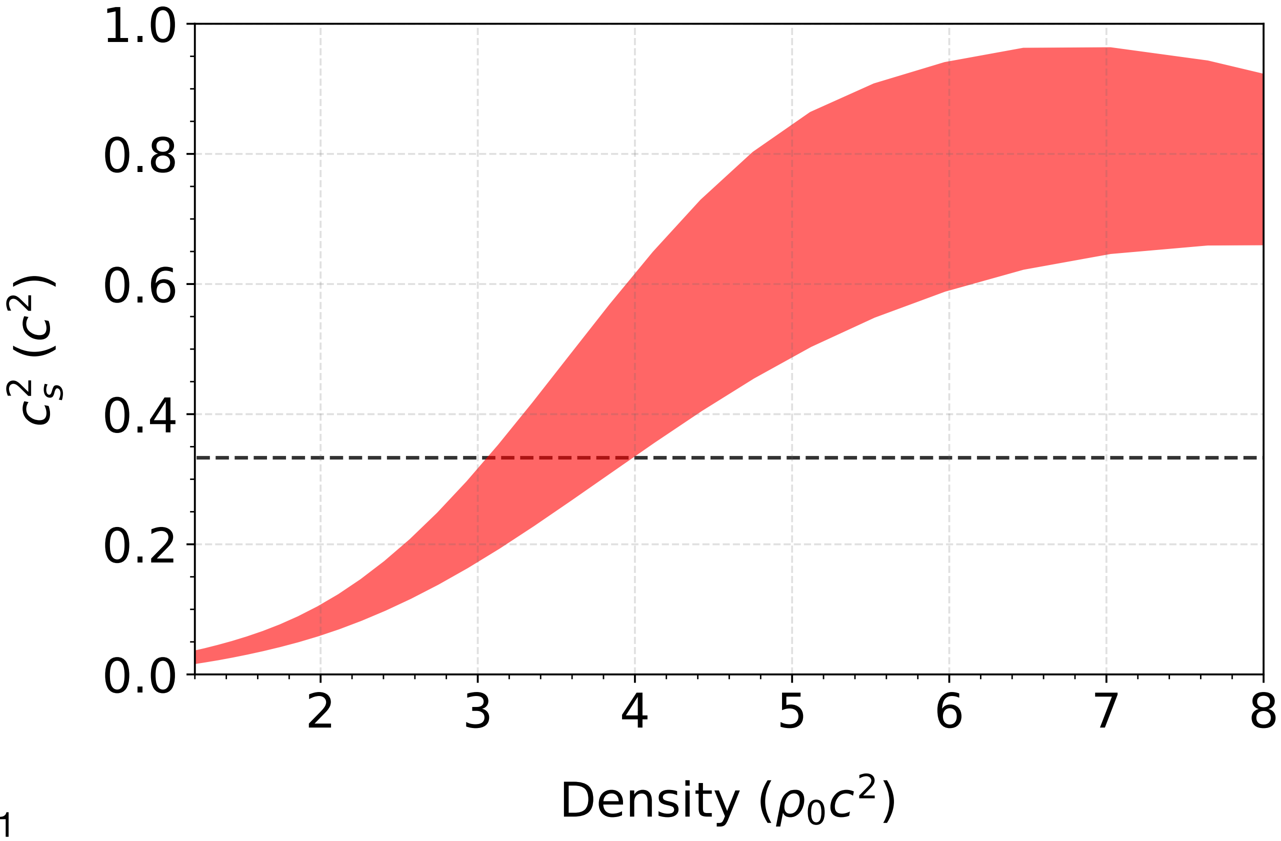


Consists with the GW observation

Phase Transitions?

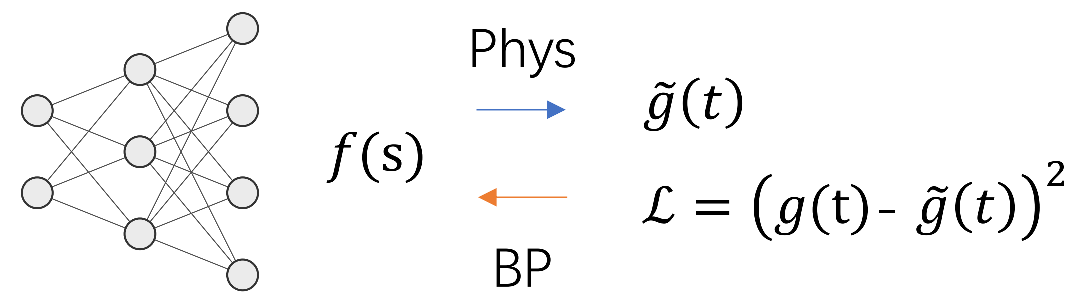
1st order PT/ cross-over is not ruled out.

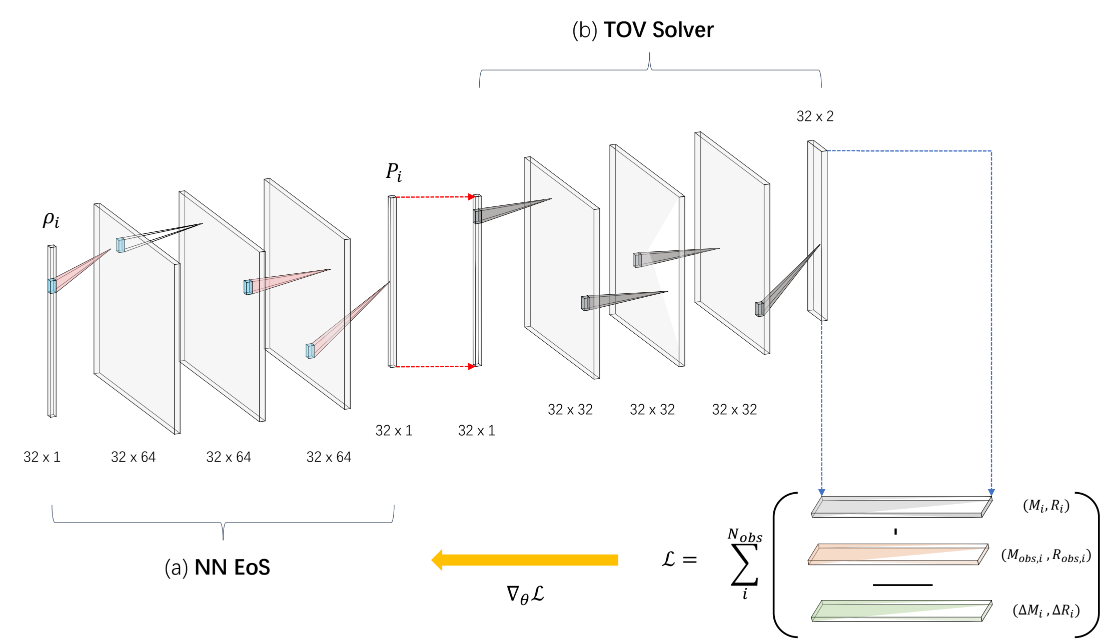
Need more accurate observations or new observables!

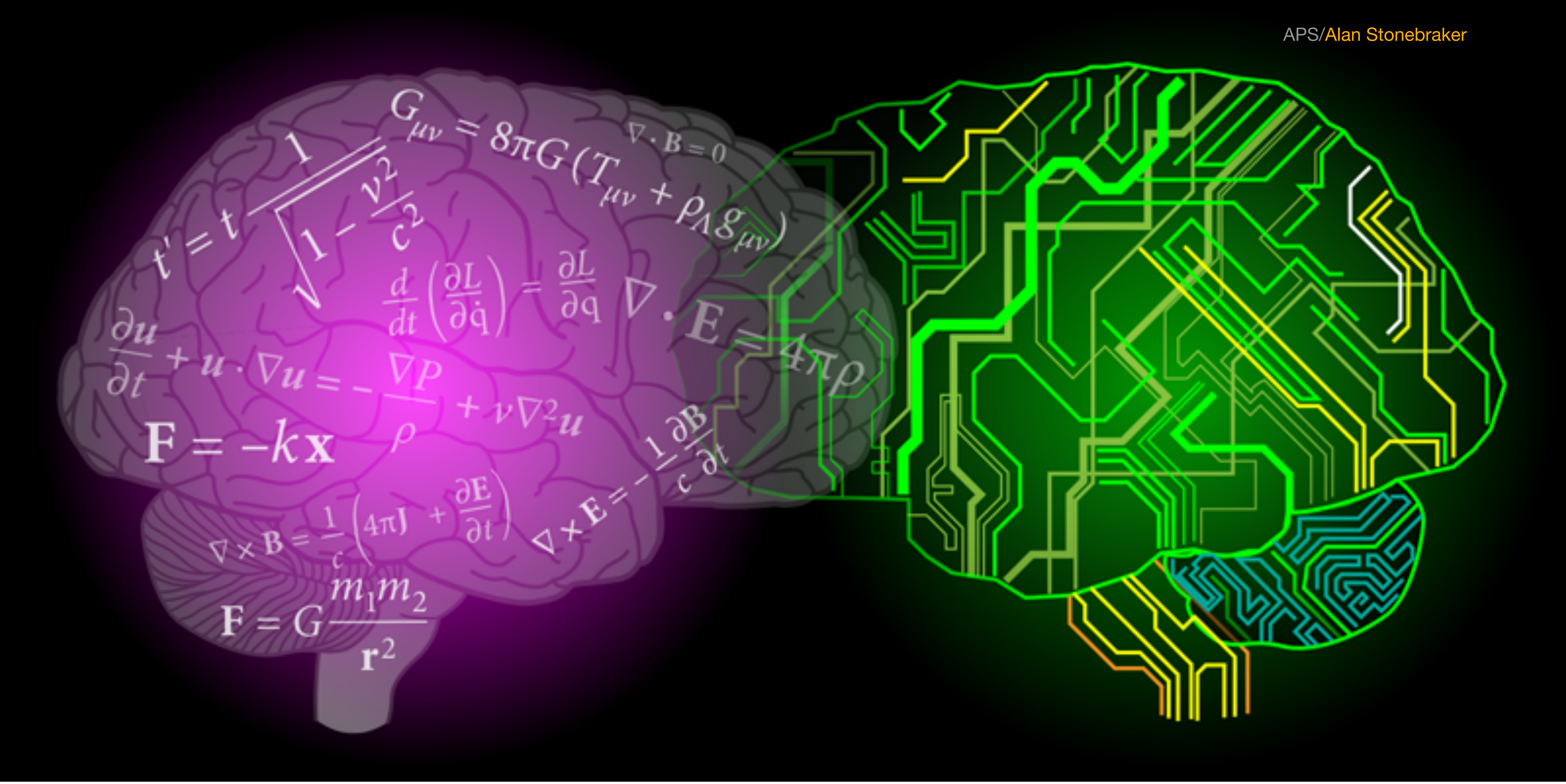


Summary
and Outlooks

- **Take-home messages**
  - AD can solve inverse problem using uncertain observations unsupervisedly
  - Neural network representations can help us to reconstruct EoS unbiasedly and can be trained easily
- **Future works**
  - Phase transitions
  - Multi-messager observations
  - Fully-physical AD
  - Open package…





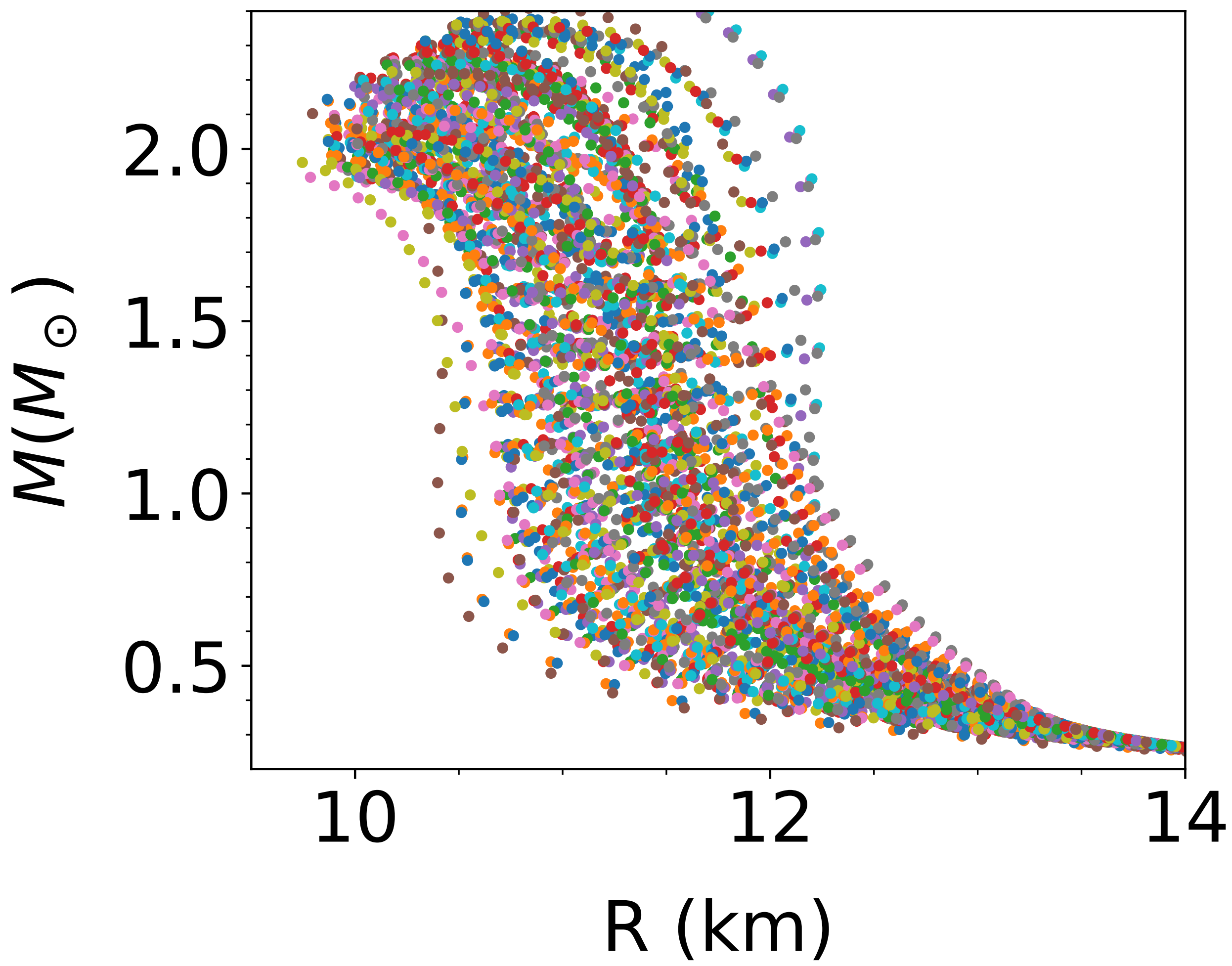


Future

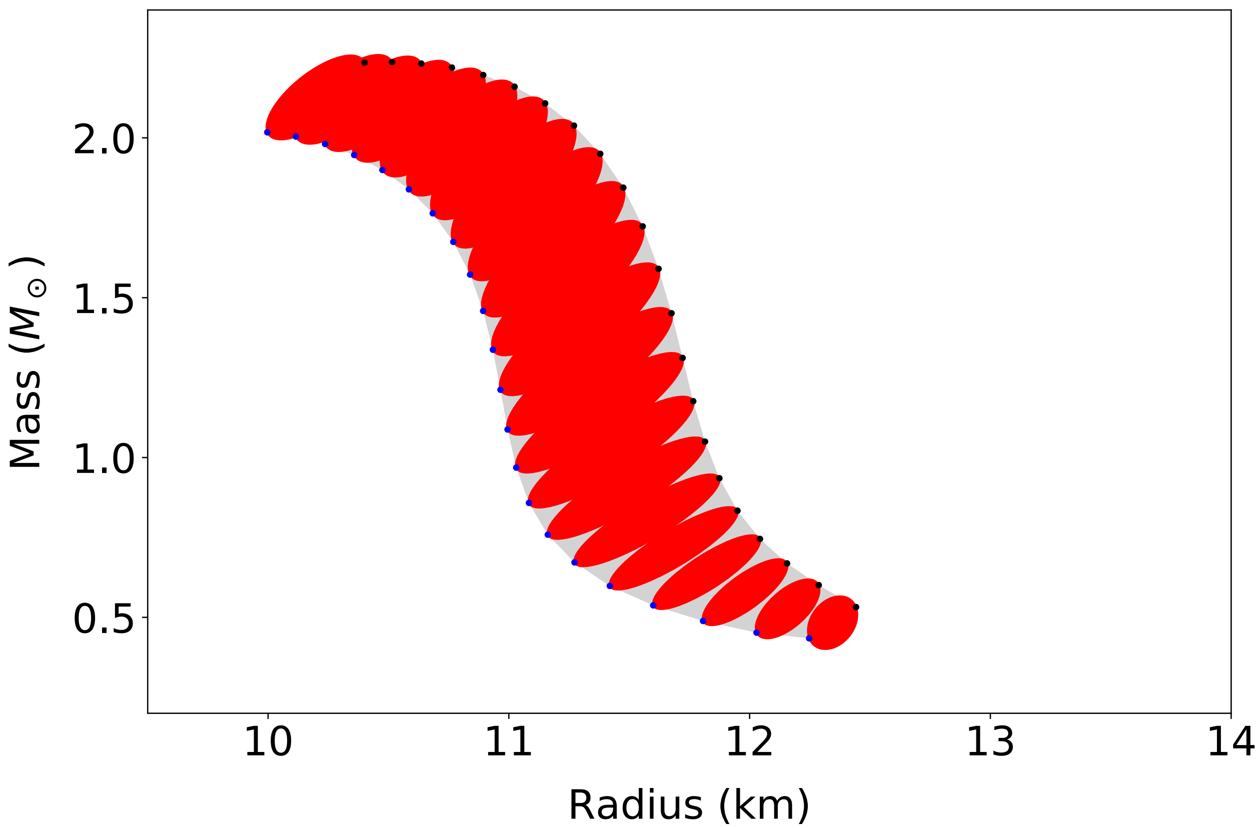
AD in Physics, opportunities and challenges

Backups

Calculate the uncertainty

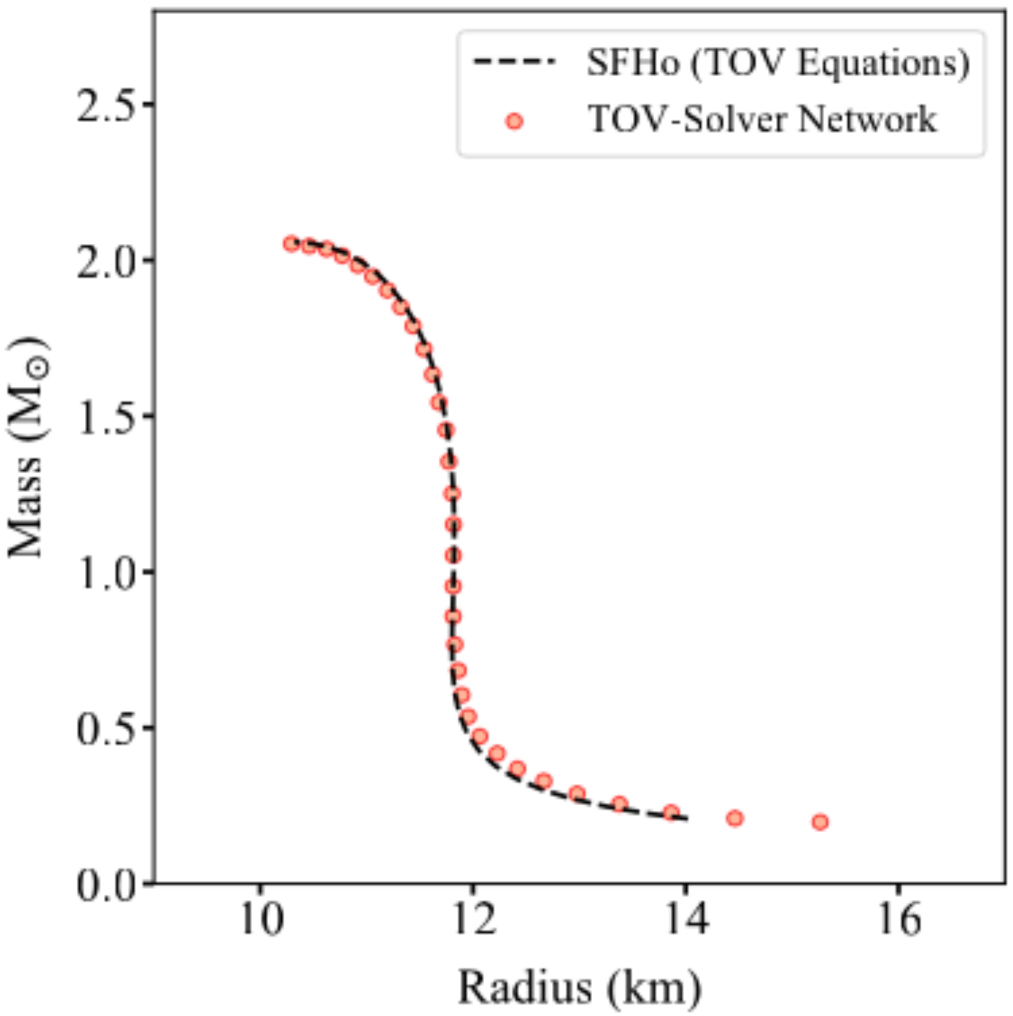


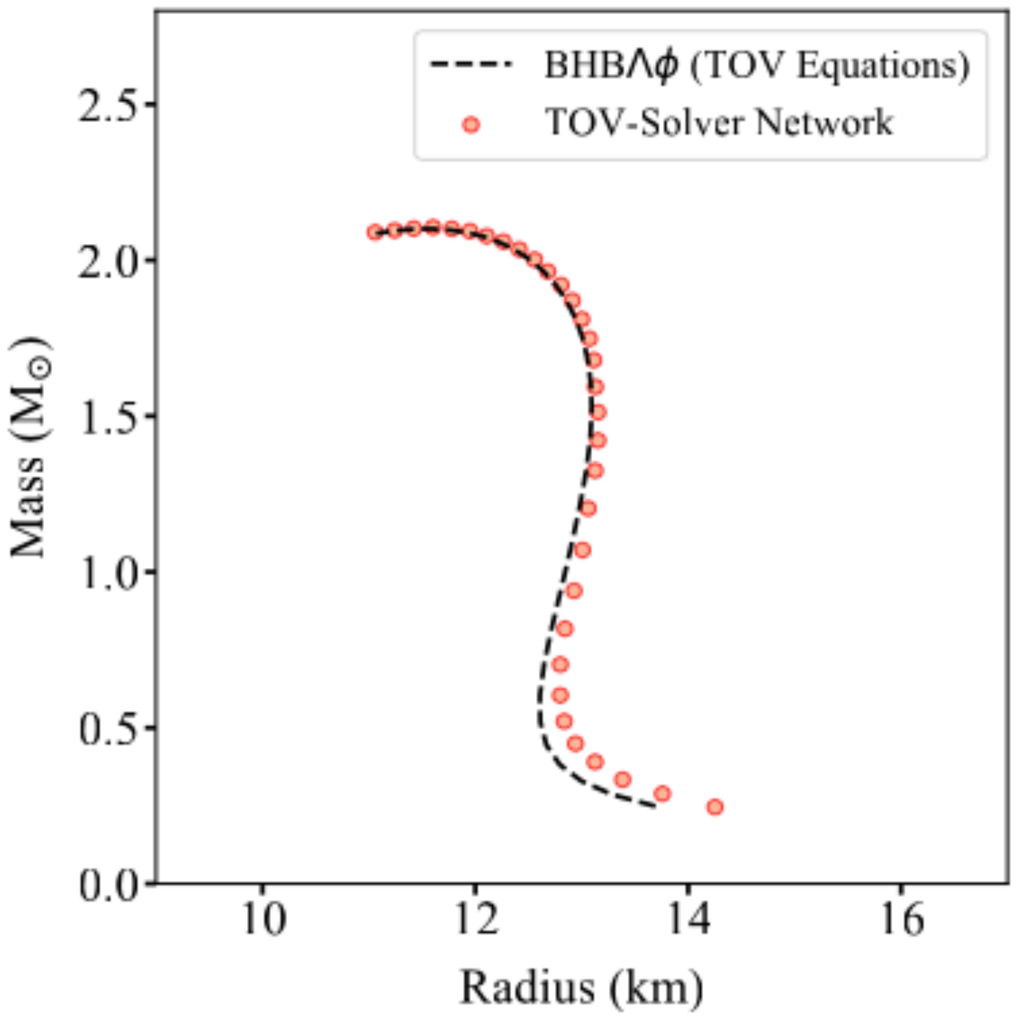
236 samples give us different batches of M-R pairs.
Gaussian fitting for each batch to get the uncertainty.



Backups

Closure test for TOV solver

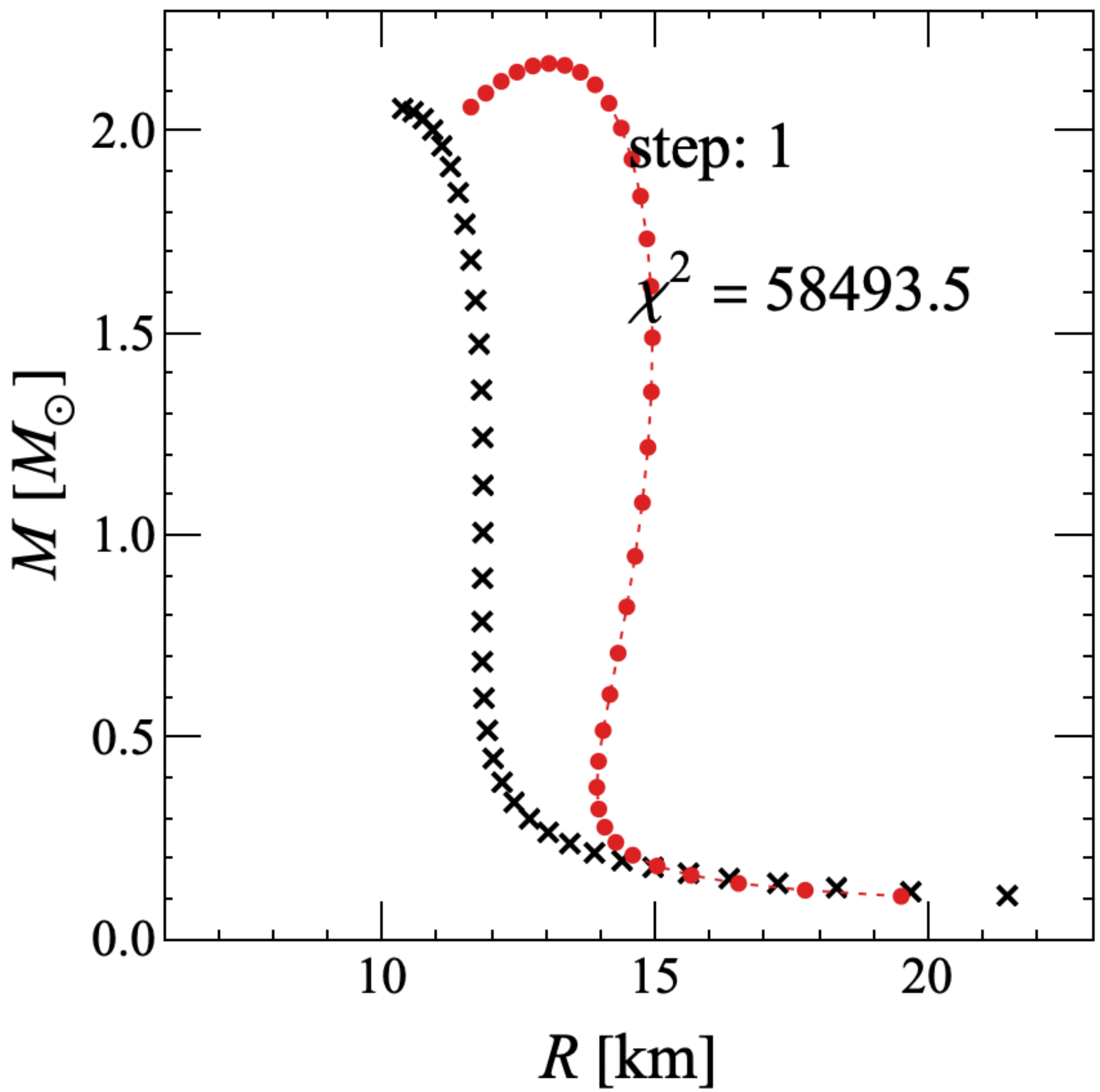




The NN TOV solver is not perfect.
We are replacing it with fully-physical differentiable modules.

Backups

Training process



Our interested area