FREmu

Power spectrum emulator for f(R) gravity

Jiachen Bai - Department of Astronomy, Beijing Normal University

F(R) gravity & Large-scale structure of the Universe

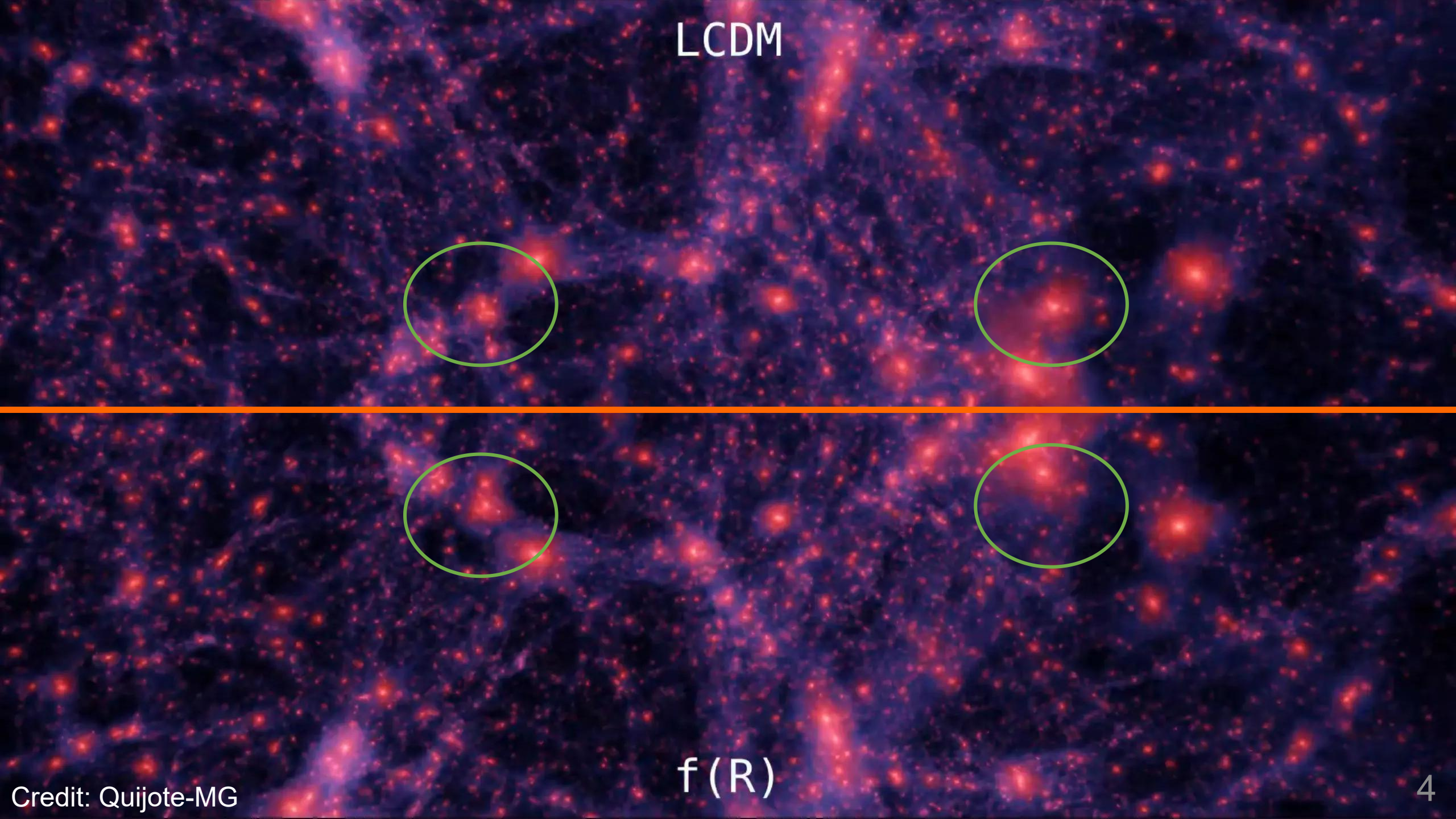
$$S = \int \mathrm{d}^4x\sqrt{-g}\left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_\mathrm{m}\right]$$

An additional term as the deviation from GR

F(R) gravity & Large-scale structure of the Universe

GR: $$\nabla^2\Phi = \frac{16\pi G}{3}\delta\rho$$

F(R): $$\nabla^2\Phi = \frac{16\pi G}{3}\delta\rho - \frac{1}{6}\delta R$$



Credit: Quijote-MG

F(R) gravity & Large-scale structure of the Universe

$$P(\mathbf{k}) \equiv V\langle|\delta_{\mathbf{k}}|^2\rangle = \int d^3r\, e^{-i\mathbf{k}\cdot\mathbf{r}}\xi(\mathbf{r})$$

We can now predict linear P(k) precisely,

but non-linear effects cannot be ignored.

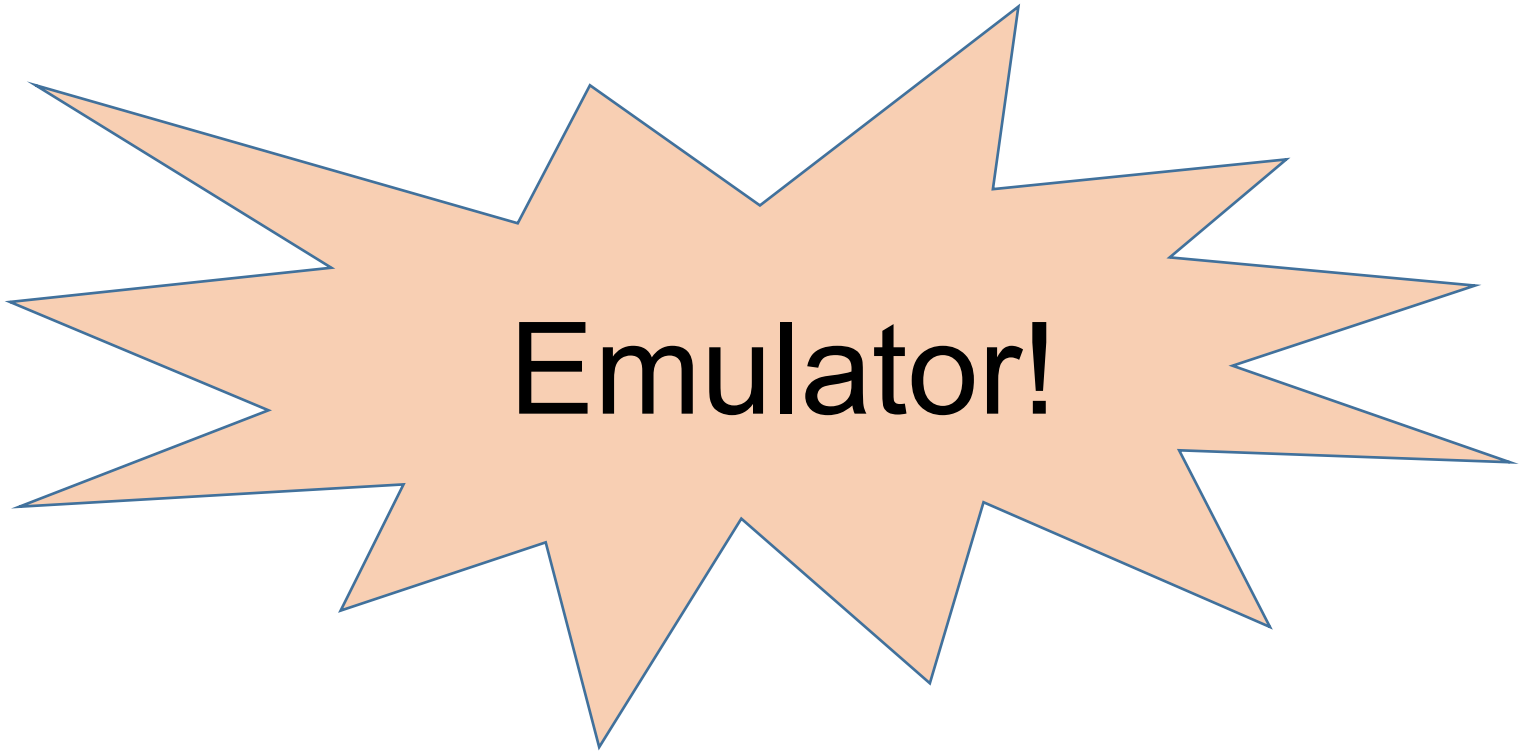
Simulation & Emulation

N-body simulations => Non-linear P(k)

MONEY & TIME (Months)

MCMC? IMPOSSIBLE!!!

Simulation & Emulation



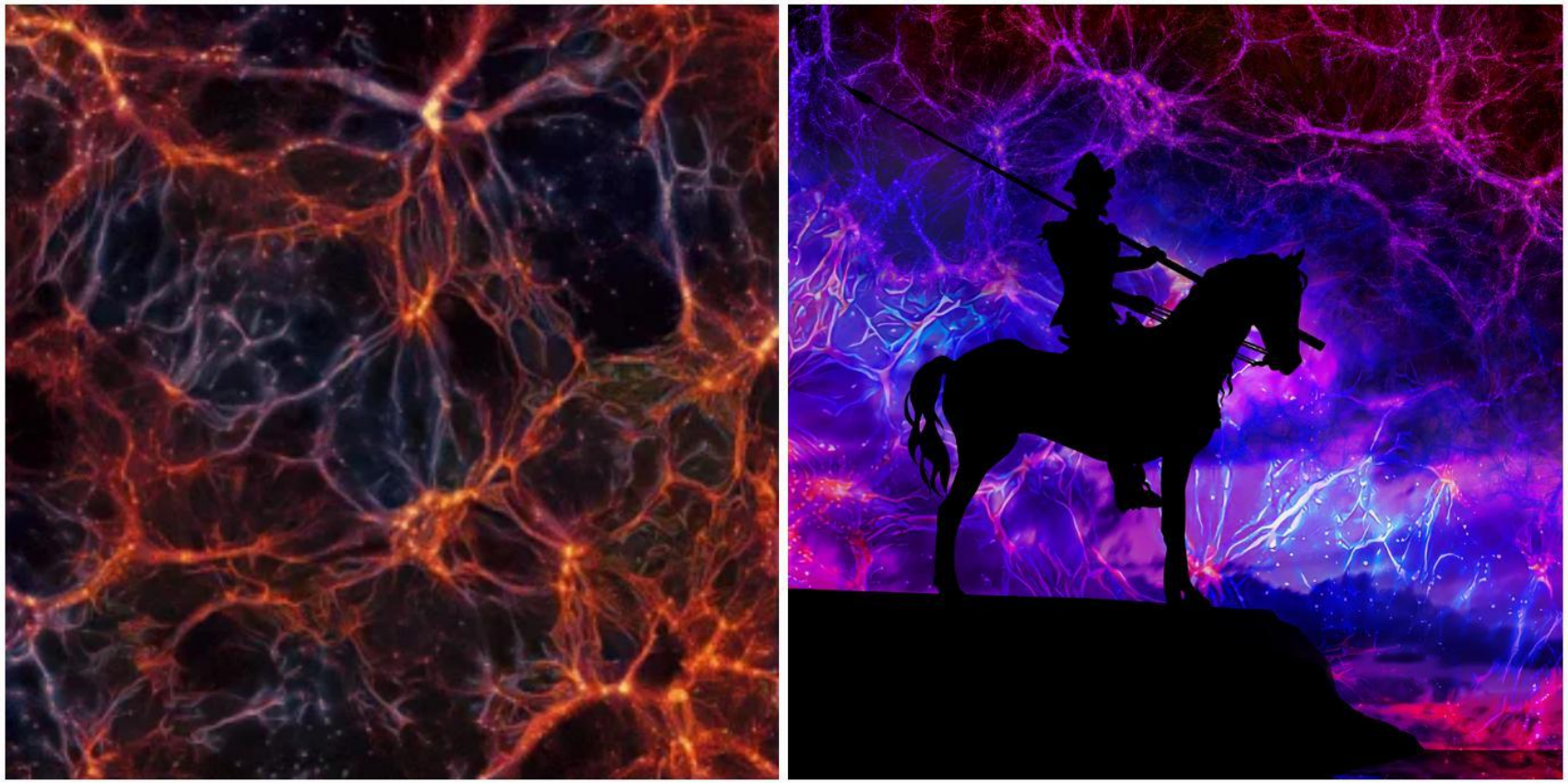
Directly predict

P(k) in only

few seconds!

Build an emulator - Data

Quijote simulations



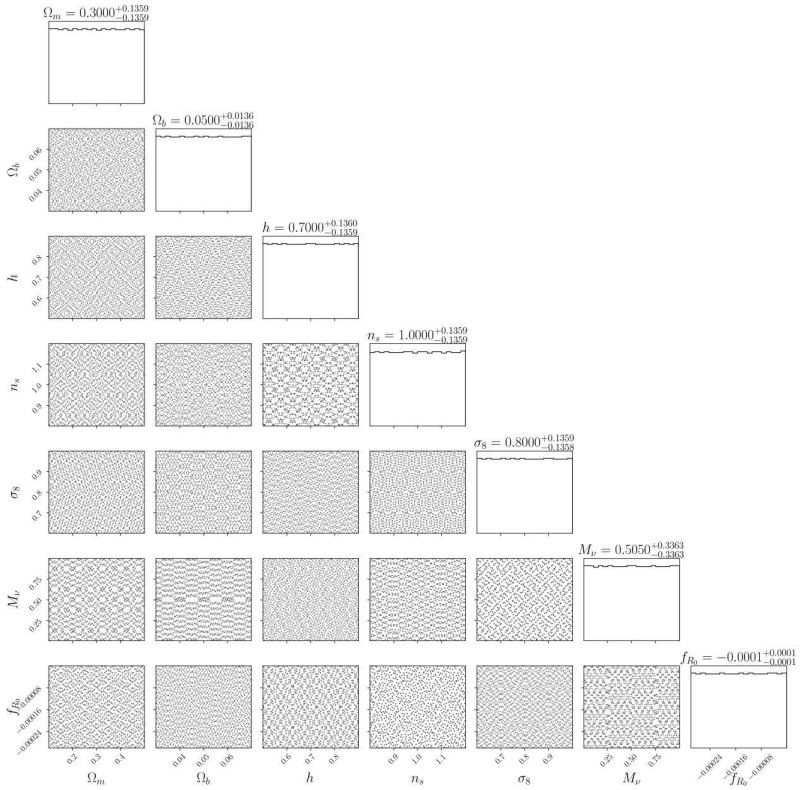
Quijote-MG

2048 sets

SB7

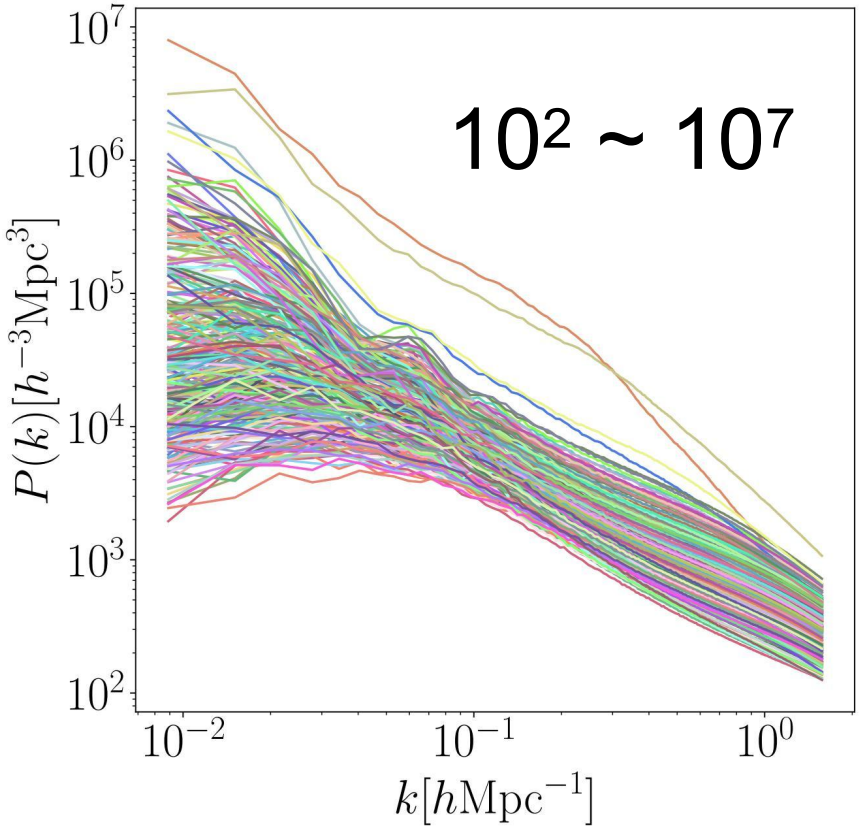
Build an emulator - Targets

Parameters



Map

Power spectrum



Build an emulator - Boosts

$$B(k) = P^{\mathrm{nonlin}}_{f(R)}(k) / P^{\mathrm{halofit}}_{\Lambda\mathrm{CDM}}(k)$$

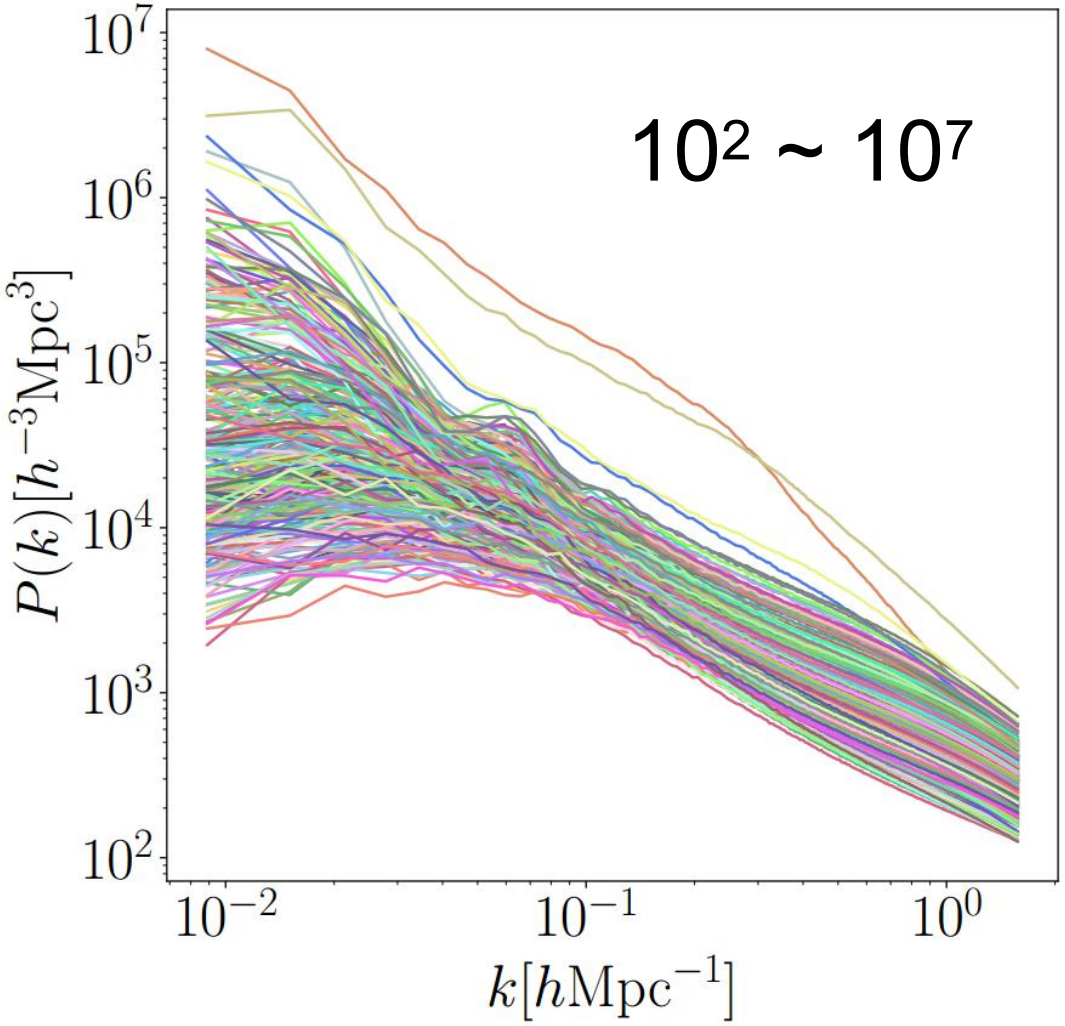
Halofit: Mead2020_feedback computed with CAMB

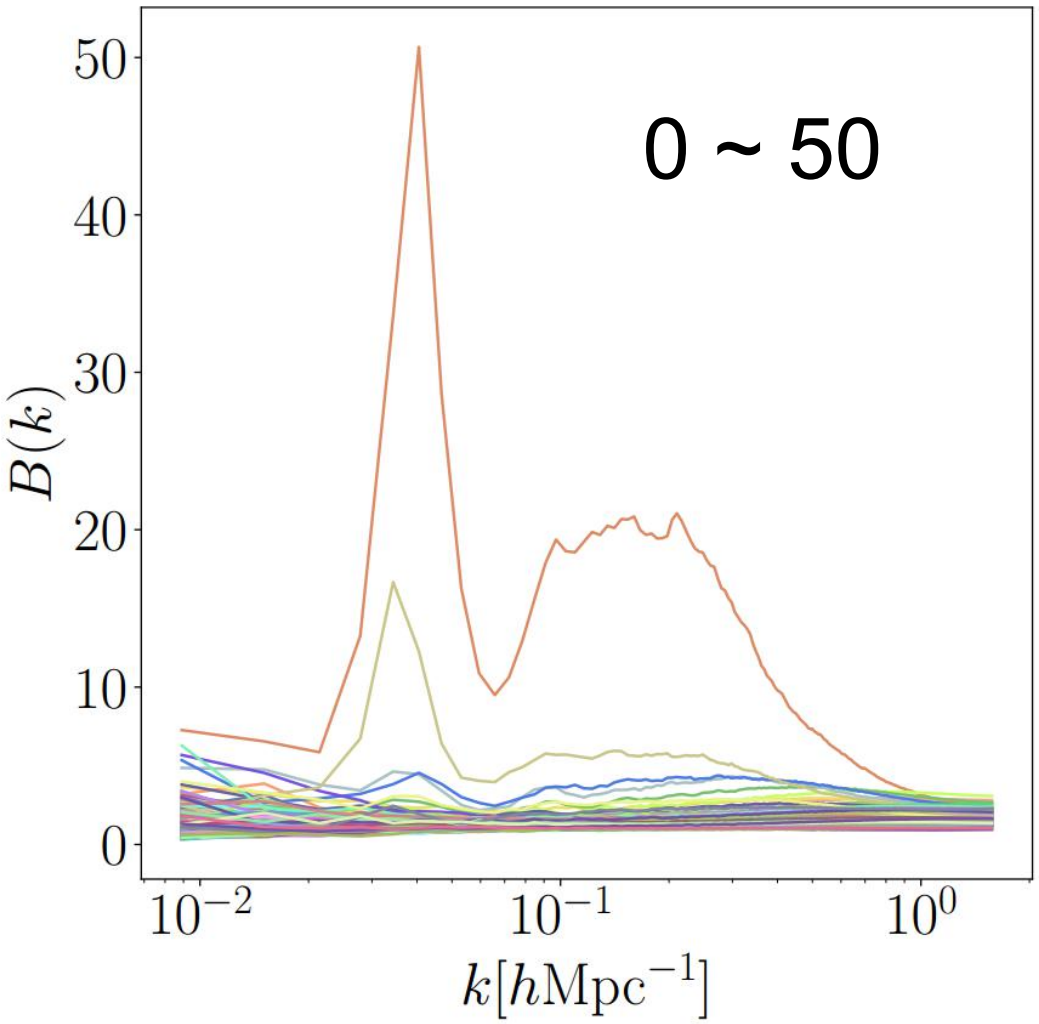
$$\mathrm{P_{f(R)}(k)} = \mathrm{P_{halofit}(k)} \times \mathrm{B(k)}$$

Fast Fast

Focus on f(R) & M_ν features

Build an emulator - Boosts





Build an emulator - PCA

7 parameters

Map

Hundreds of points

PCA

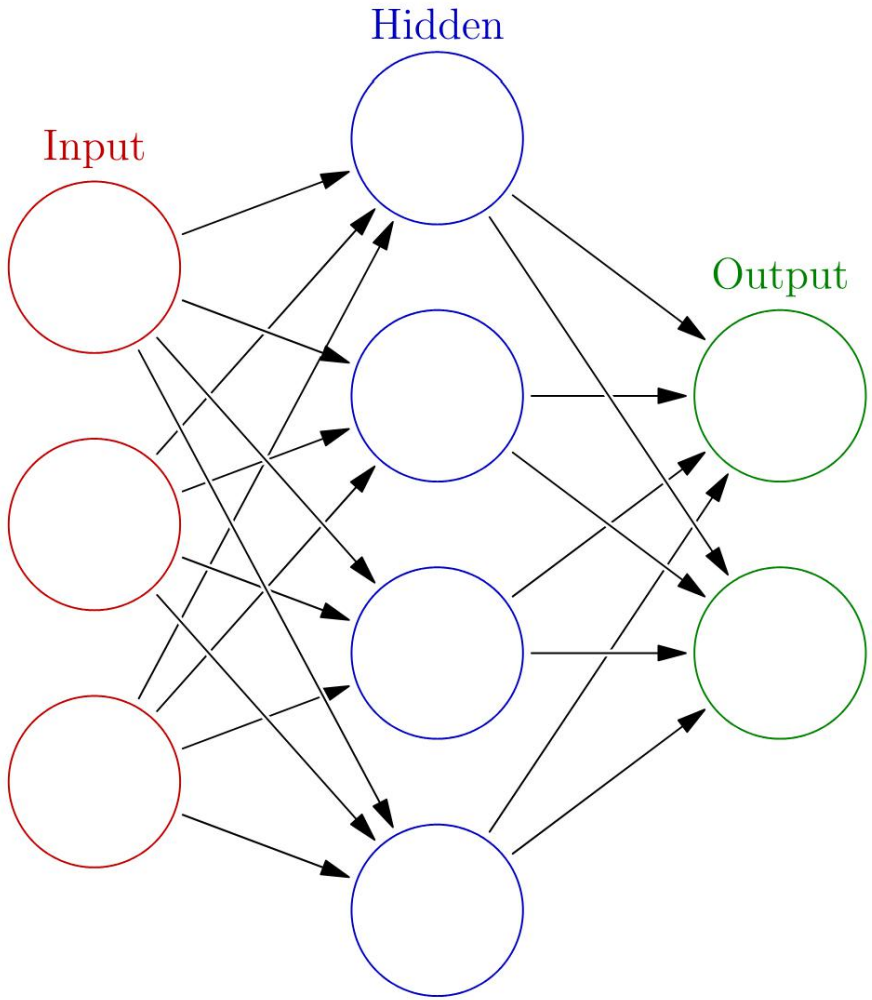
7 parameters

Map

30 PCA coefficients

$$B(k,z;\theta) = \mu_B(k,z) + \sum_{i=1}^{N_{pc}} \phi_i(k,z) w_i(\theta) + \epsilon$$

Build an emulator - ANN



7 Input

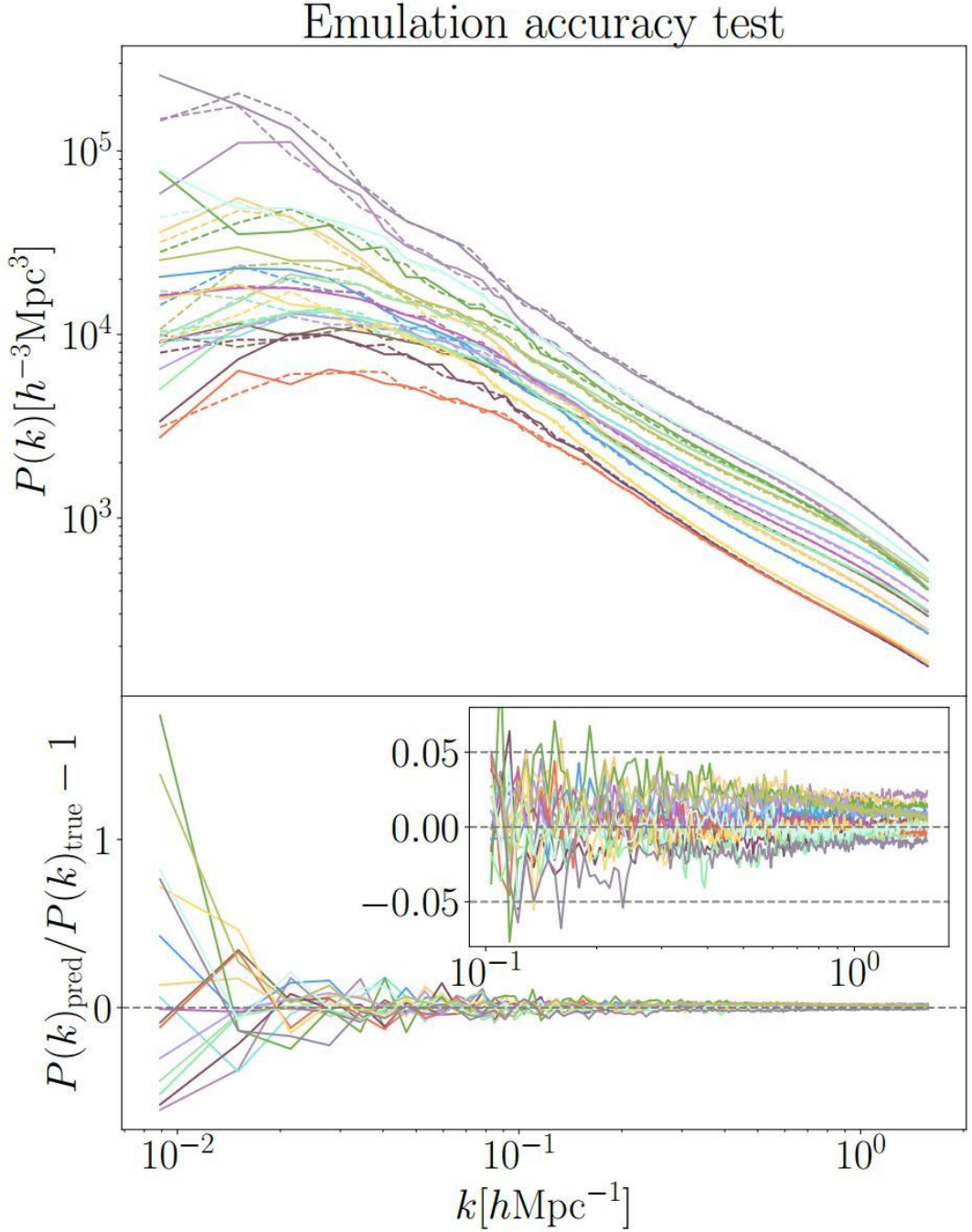
650 Hidden

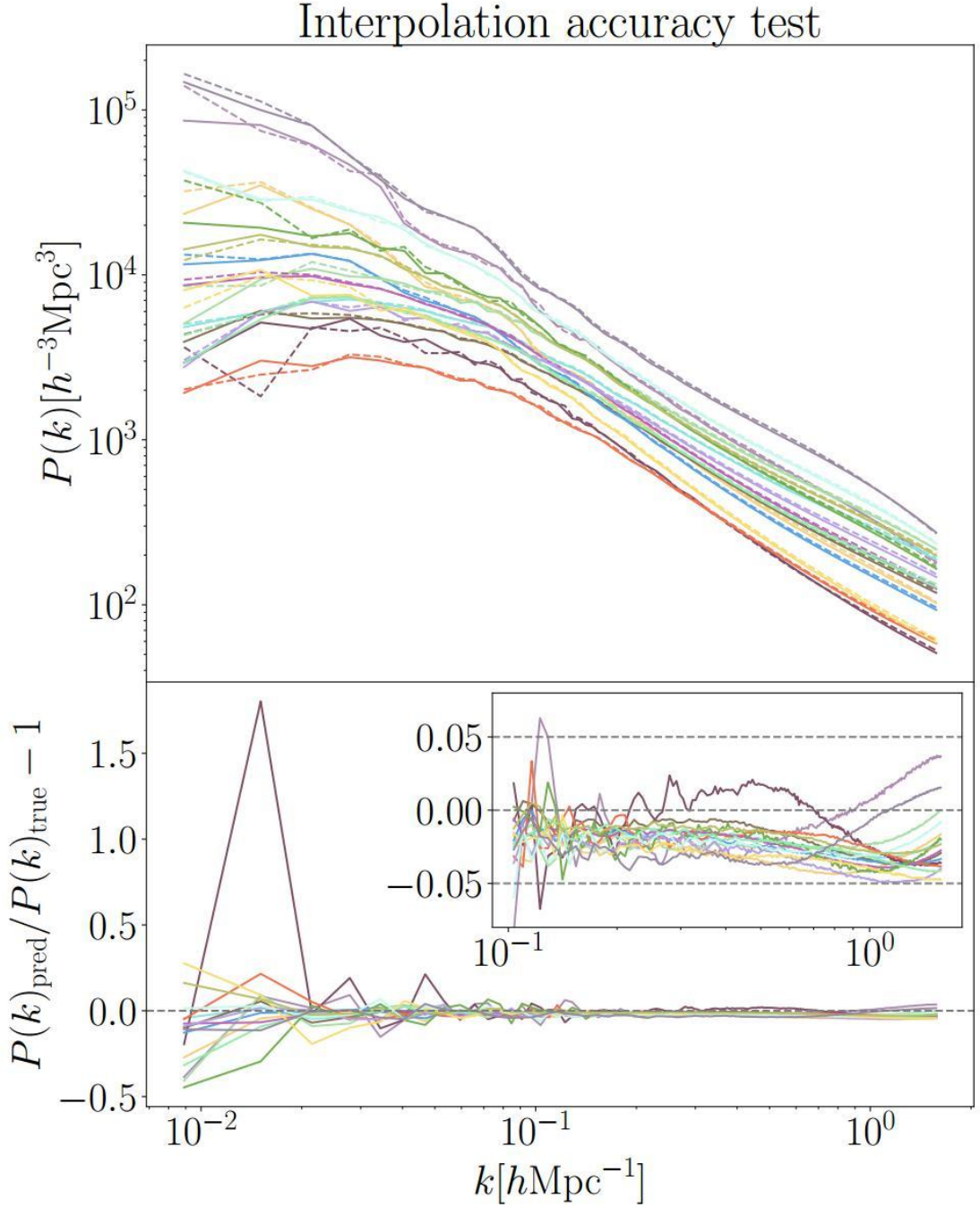
30 Output

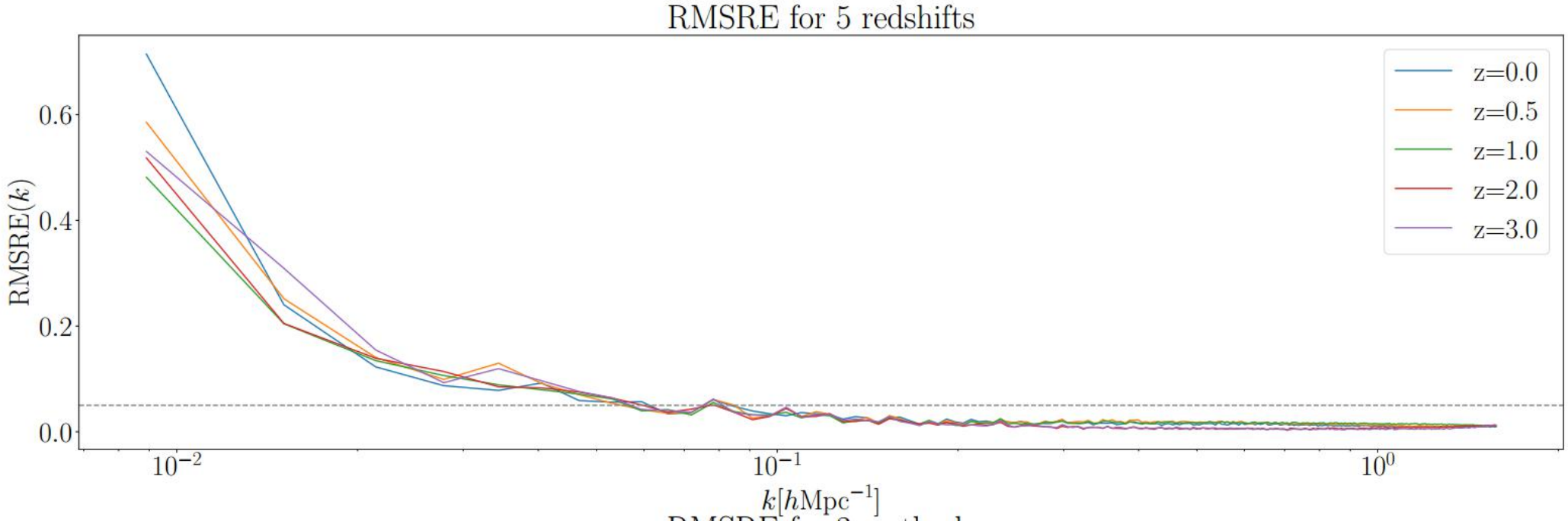
Linear - Sigmoid - Linear

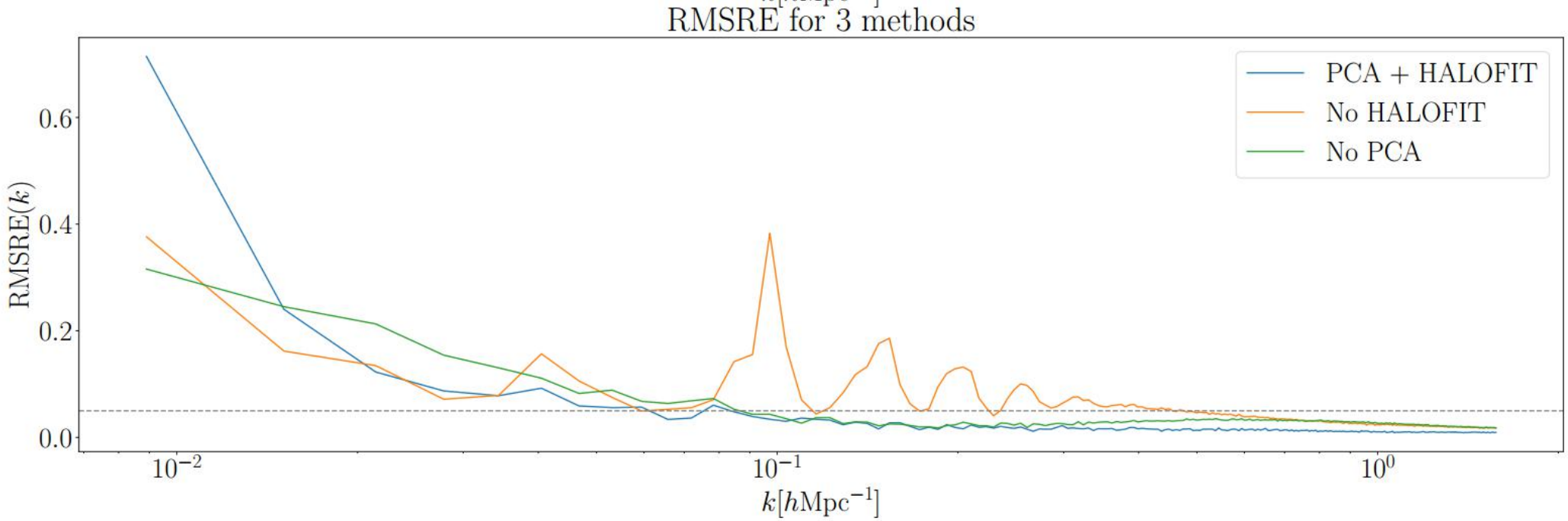
Results

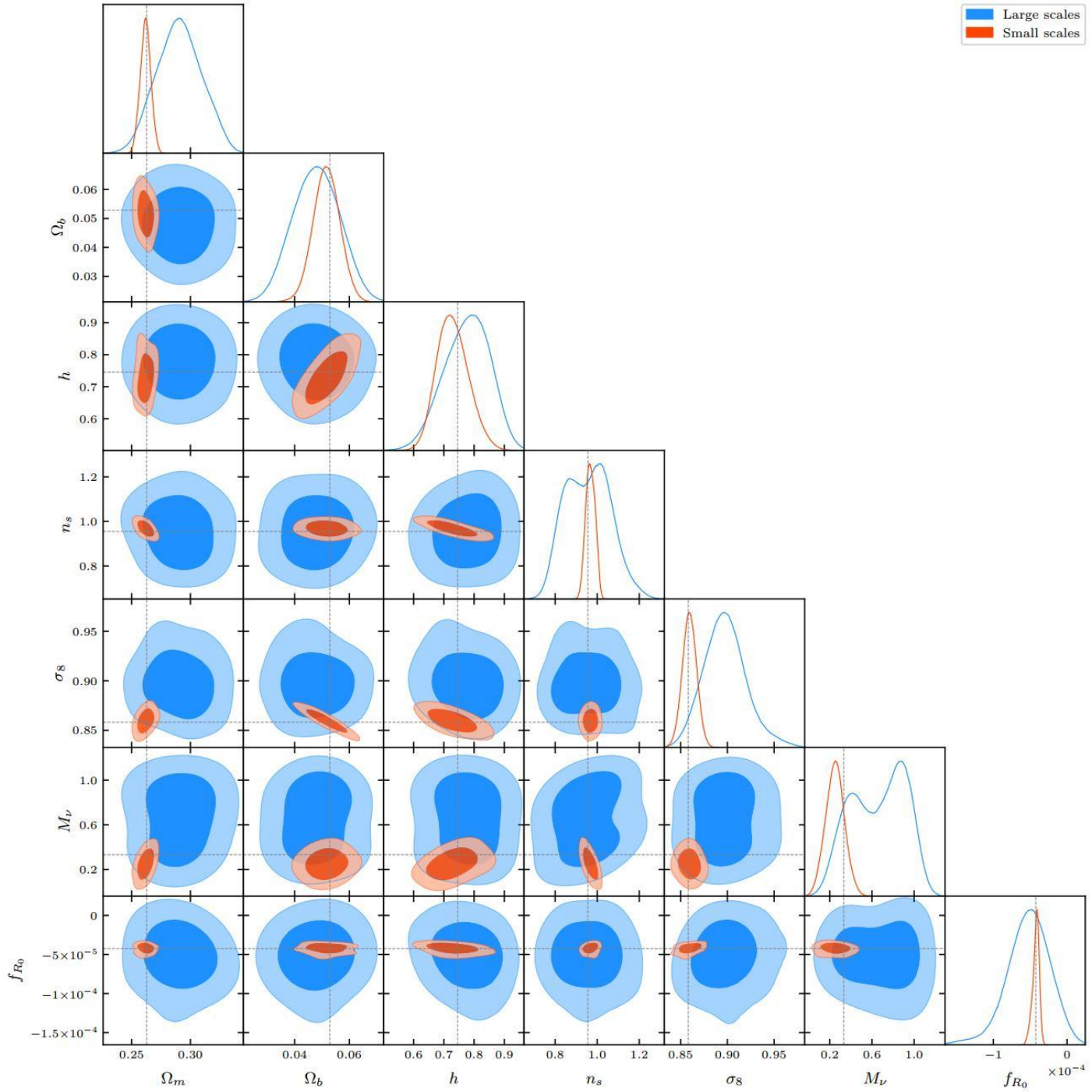
Accuracy & Application


Emulation accuracy test
$P(k)[h^{-3}\mathrm{Mpc}^3]$
$P(k)_{\mathrm{pred}}/P(k)_{\mathrm{true}} - 1$
$k[h\mathrm{Mpc}^{-1}]$


Interpolation accuracy test
$P(k)[h^{-3}\mathrm{Mpc}^3]$
$P(k)_{\mathrm{pred}}/P(k)_{\mathrm{true}} - 1$
$k[h\mathrm{Mpc}^{-1}]$


RMSRE for 5 redshifts
RMSRE(k)
0.0
0.2
0.4
0.6
10^{-2}
10^{-1}
10^{0}
$k[h\mathrm{Mpc}^{-1}]$
z=0.0
z=0.5
z=1.0
z=2.0
z=3.0


RMSRE for 3 methods
RMSRE(k)
0.0
0.2
0.4
0.6
10^{-2}
10^{-1}
10^{0}
$k[h\mathrm{Mpc}^{-1}]$
PCA + HALOFIT
No HALOFIT
No PCA



Application

MCMC!

$$\Sigma\left(k_1, k_2\right) = \left[\frac{2}{N_{k_1}} + \sigma^2_{\mathrm{sys}}\right] P^2\left(k_1\right) \delta_{k_1 k_2}$$

$$\mathcal{L}(\theta \mid D) = \exp\left(-\frac{1}{2}(\mathbf{O}(D) - \mathbf{M}(\theta))^T \Sigma^{-1} (\mathbf{O}(D) - \mathbf{M}(\theta))\right)$$

Parameter	True	Small-scale	Large-scale
$\boldsymbol{\Omega_m}$	0.263	$0.2619^{+0.0091}_{-0.0093}$	$0.290^{+0.035}_{-0.035}$
$\boldsymbol{\Omega_b}$	0.053	$0.0516^{+0.0095}_{-0.0088}$	$0.049^{+0.014}_{-0.015}$
$\boldsymbol{h}$	0.746	$0.729^{+0.10}_{-0.097}$	$0.78^{+0.13}_{-0.14}$
$\boldsymbol{n_s}$	0.955	$0.968^{+0.042}_{-0.042}$	$0.95^{+0.17}_{-0.17}$
$\boldsymbol{\sigma_8}$	0.858	$0.860^{+0.016}_{-0.015}$	$0.898^{+0.042}_{-0.042}$
$\boldsymbol{m_\nu}$	0.332	$0.25^{+0.17}_{-0.17}$	$0.66^{+0.36}_{-0.44}$
$\boldsymbol{f_{R_0}}$	$-4.24 \cdot 10^{-5}$	$\left(-4.22^{+0.81}_{-0.98}\right) \cdot 10^{-5}$	$\left(-52^{+51}_{-61}\right) \cdot 10^{-6}$

Thank you!

FREmu

```
pip install fremu
```

Documentation: https://astrobai.github.io/codes/fremu

Source code: https://github.com/AstroBai/FREmu