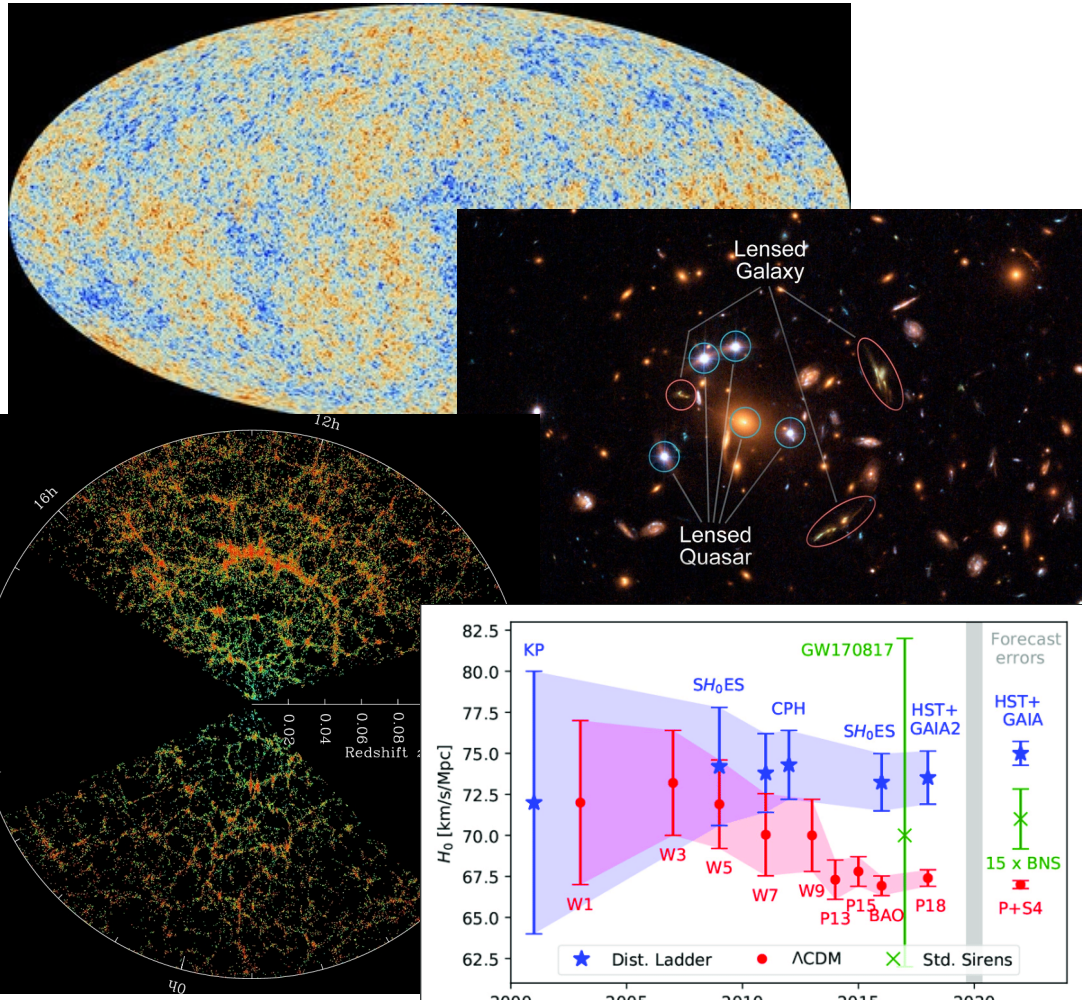
Machine learning and simulations in cosmology

Balázs PÁL

[1]*Department of Physics of Complex Systems*, Eötvös Loránd University
[2]*Heavy ion physics research group*, Wigner Research Centre for Phyics

Goals of modern cosmology

- Improve the standard model of cosmology
- Solve challanges of the ΛCDM model
  - H_0 tension (CMB vs. SN)
  - Violations of homogeneity and isotropy
  - ...
  - Dark energy(???)
  - Eg. AvERA model

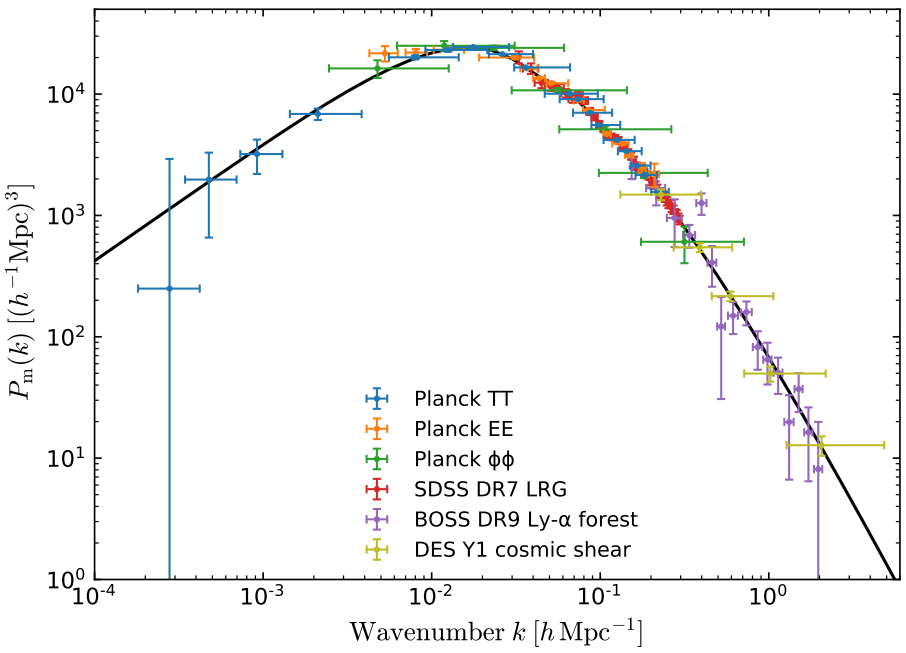


ELTE EÖTVÖS LORÁND UNIVERSITY

WIGNER

Simulations in cosmology

- “There is only a single real universe we can observe, but with simulations we can create ourselves a million more.”
- Goal is to gather copious amount of statistical information
  - Analyzing statistical quantities like power spectrum
  - Derive relevant physical parameters
- Simulating both small and large sections of the universe
  - Small scale: Galaxies or a single galaxy groups
  - Large scale: Hundreds to thousands of Mpc large sections of the universe



Source: Planck 2018. results

Flavors of cosmological simulations

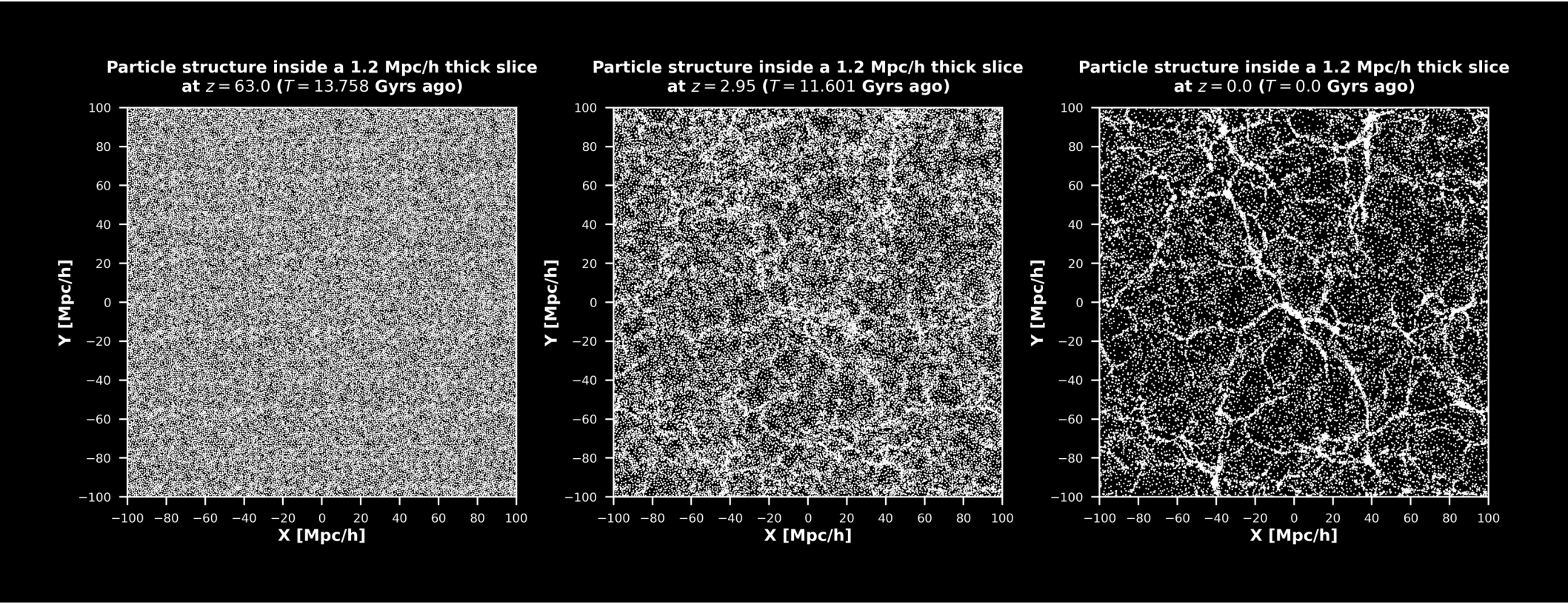
- N-body simulations
  - Simulated objects: ***particles***
  - Usually pure dark matter simulations (dark matter == galactic halo)
  - Solves the Newtonian equations of motion
- Hydrodynamic simulations
  - Simulated objects: ***volume elements***
  - Used to simulate gas, stars etc.
  - Much harder to program but still completely feasible
  - Employs classical formulae, just like N-body codes

How to do simulations in cosmology?

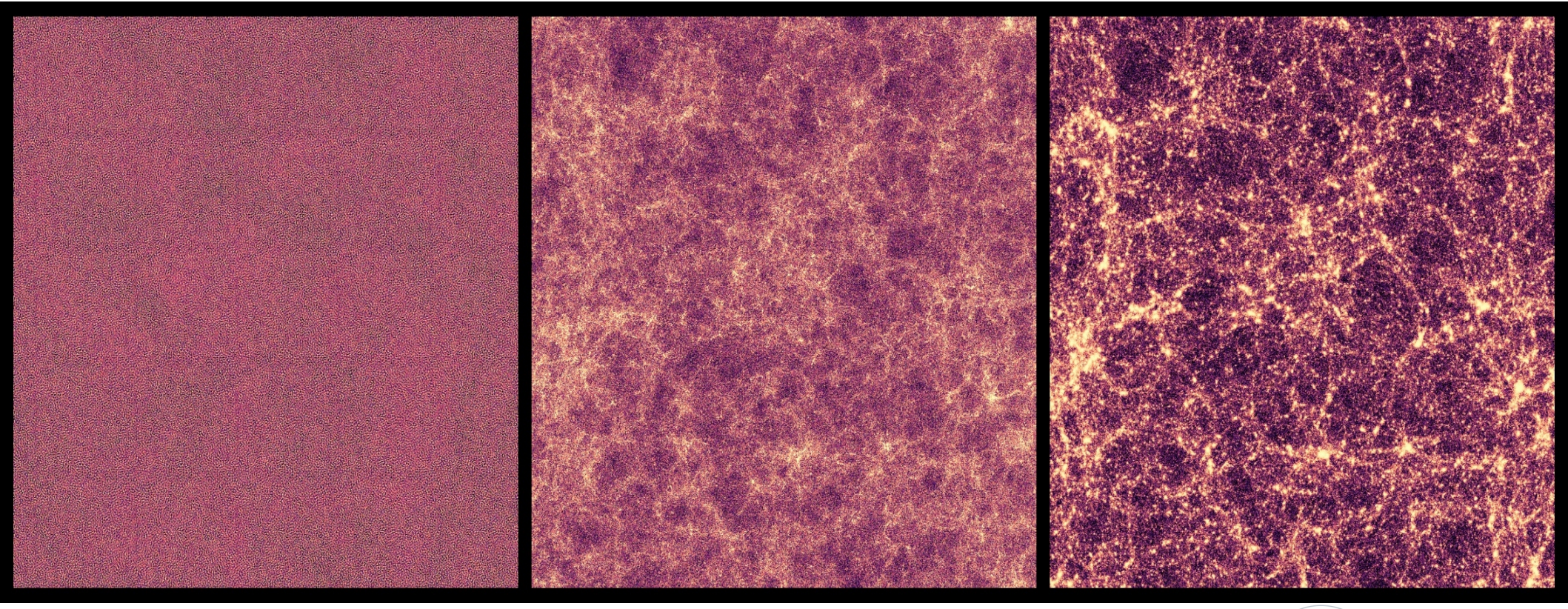
- Usually large scale simulations are used
- The idea behind them is simple

1) A nearly homogeneous matter density distribution is sampled from the CMB
   - In case of N-body sims., individual particles represent this
   - In case of HD sims., we use the density field itself
2) Particles or just the density field itself, as well as relevant physical parameters are evolved using classical equations of motion and the Friedmann equations
3) The present time ($z = 0$) is usually the end of the simulation, giving us the fully-formed large scale web-like structure of the universe

Simulations in cosmology



Simulations in cosmology

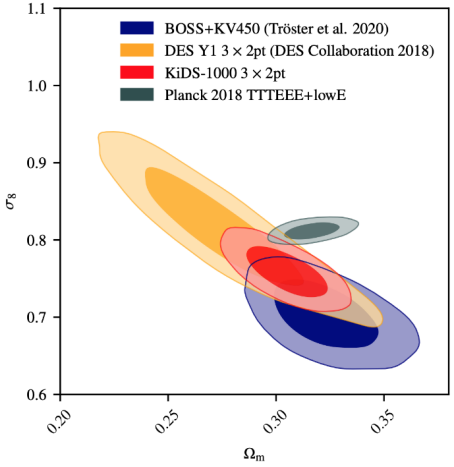


ML in astronomy

- Involved in virtually every topic nowadays
- Astronomy was among the first fields in history that produced a dataset referred to as “big data” (SDSS)
- Just like in other fields, ML contributed heavily to astronomical research (and probably will continue doing so for some years at least).
- Can ML be used in cosmological simulations? **Yes!**

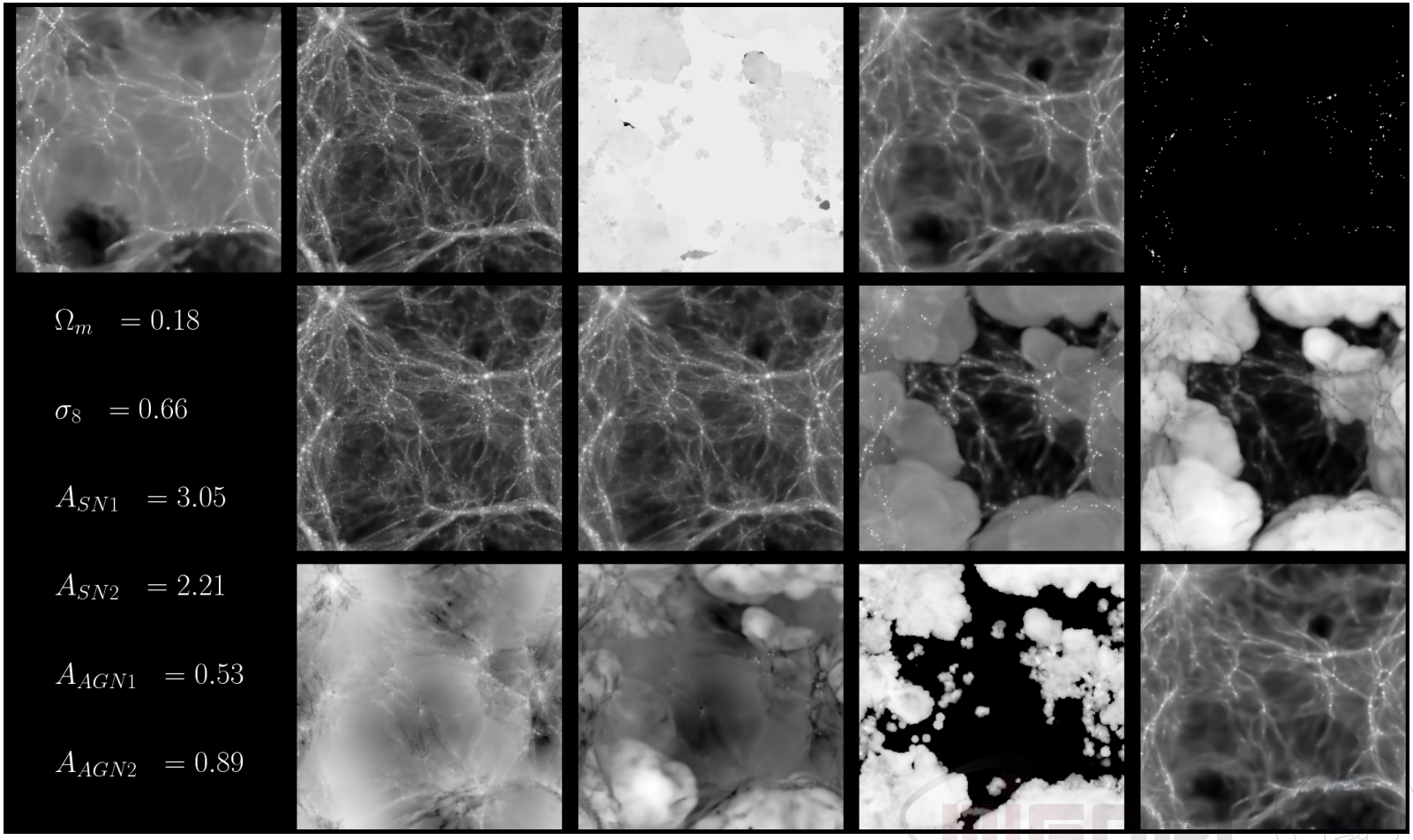
CAMELS - Connecting ML and simulations

- **C**osmology and **A**strophysics with **M**achin**E** **L**earning **S**imulations
- Created to study the $\Omega_m - \sigma_8$ tension
- A database of cosmological simulations of several (12+1) physical parameters (density, pressure, metallicity etc.) with 6 labels each (Ω_m, σ_8, and 4 astrophysical “noise-like” quantities).
- Can be shown that ML is able to determine the accurate cosmological parameters from simulations and thus possibly from observations too
- The final goal would be to determine the real cosmological parameters of our own universe from observations

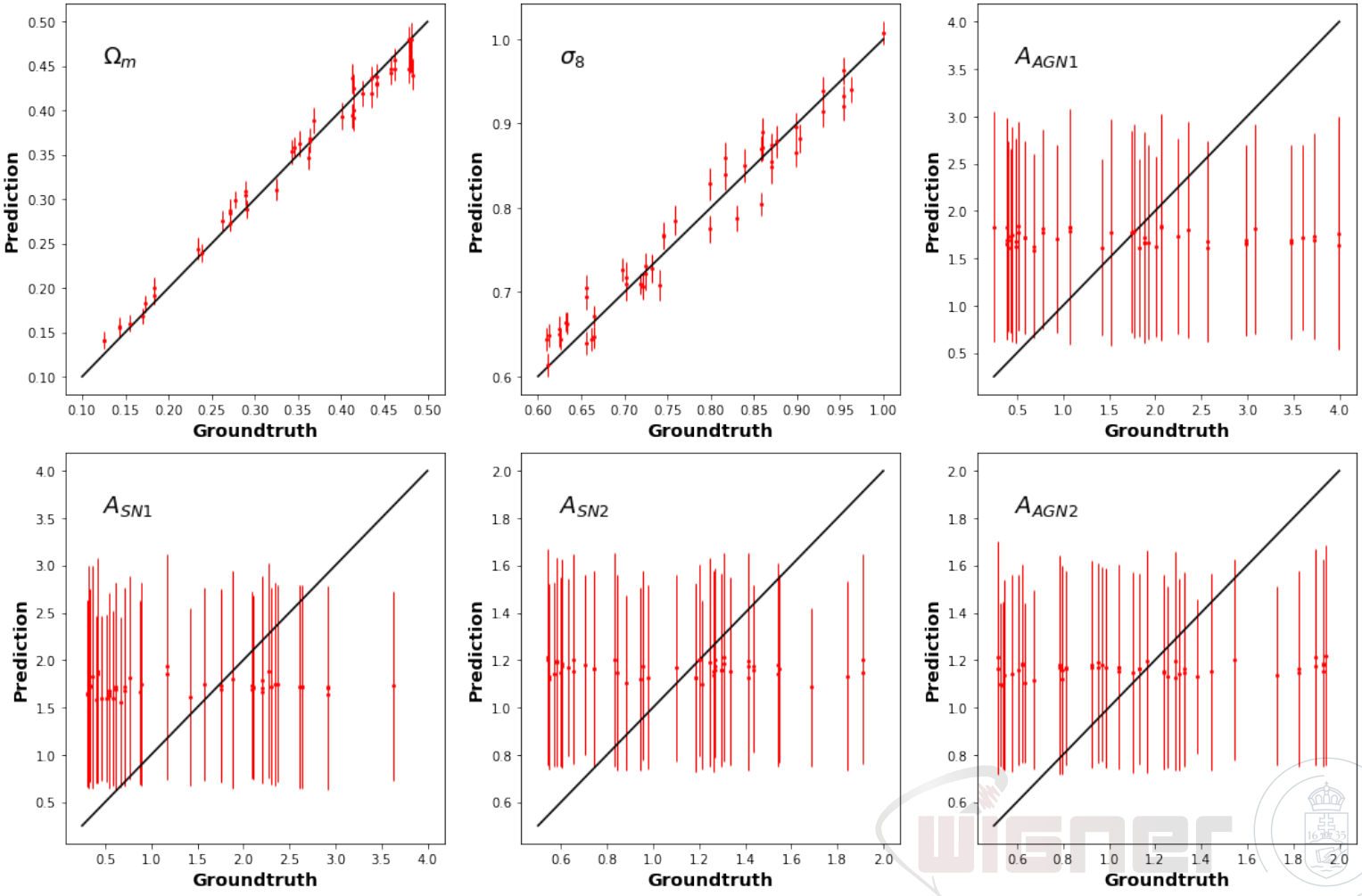


Source: Valentino *et al*. (2021)

CAMELS - Connecting ML and simulations



CAMELS - Connecting ML and simulations



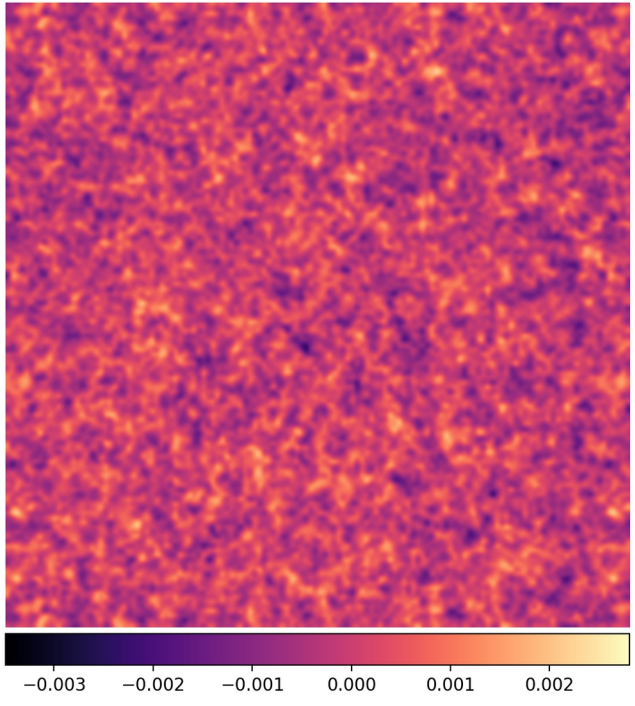
My current research

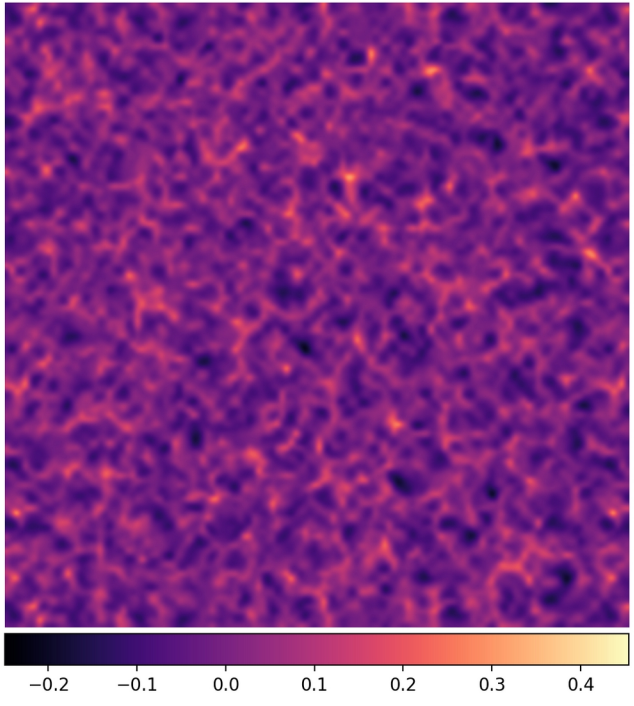
- The pinnacle of cosmological simulations should be true GR simulations of the universe
  - All large scale physical effects should be detectable (depending on the resolution)
  - Problem: They’re insanely hard to program and require much more computational power than classical codes
- There are only 2 public codes that realize true GR simulations
  - EinsteinToolkit
  - CosmoGRaPH

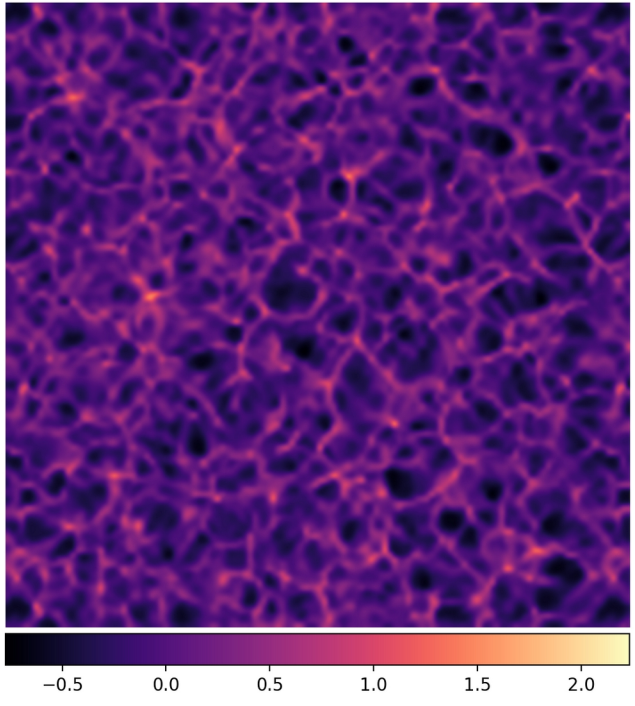
EinsteinToolkit + *FLRWSolver*

- EinsteinToolkit (ET) is the most robust tool to create GR simulations
- Utilizes the BSSN formalism that uses a 3+1D (3 spatial + 1 time dimension) foliation of spacetime
- A publicly developed package (*FLRWSolver*) can be used to generate initial conditions for ET from the CMB using linear perturbation theory
- The *McLachlan* software suite of ET solves the BSSN equations and iterates the initial conditions until $z = 0$
- Density field is evolved by solving the geodesic equation

EinsteinToolkit + *FLRWSolver*







Future goals

- Much deeper understanding of GR simulations
  - How the BSSN formalism works on a numerical level? (Possible recreation of a ”minimal example”.)
  - How numerical calculations of the Einstein’s equations can be used in other GR-related problems?
- Explore how ML can support GR simulations and alternative cosmologies?