From 30 May 2023 to 1 June 2023

Neural networks in cosmological data analysis

Isidro Gómez-Vargas[1]♠ **and José Alberto Vázquez**[1]

[1]**Instituto de Ciencias Físicas, UNAM.**

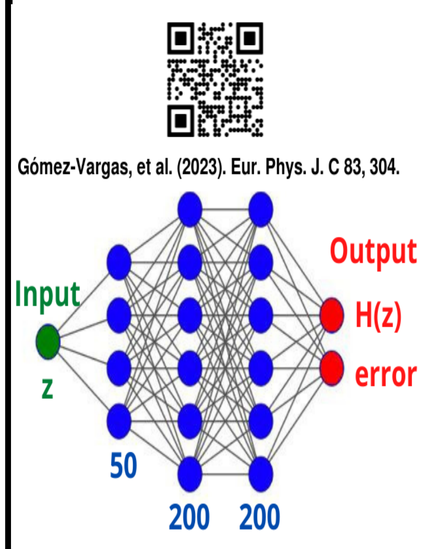
Overview

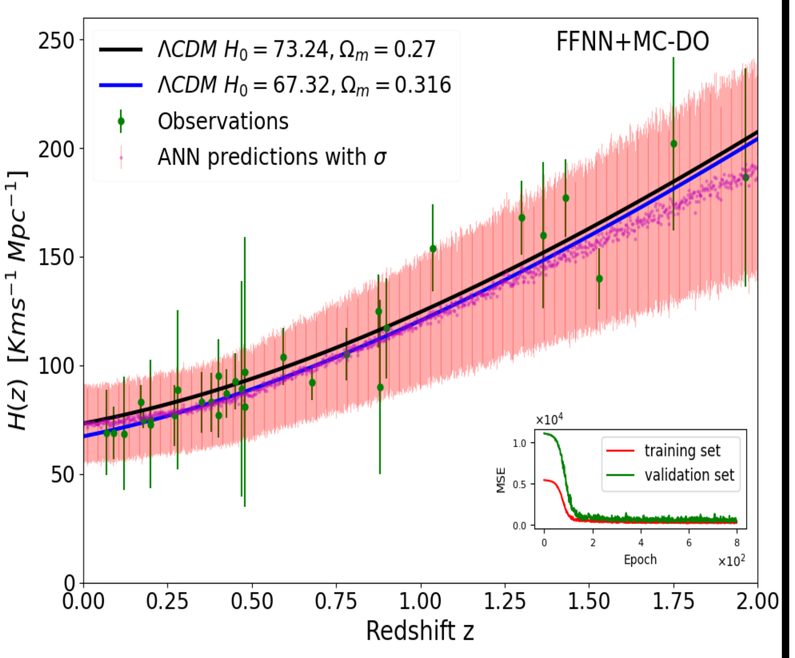
We present our recent insights in different cosmological contexts. We show a methodology for neural networks to perform model-independent reconstructions of cosmological functions. In addition, we suggest using genetic algorithms to find their hyperparameters. Furthermore, we include some applications for N-body simulations. Finally, we propose a method to accelerate Bayesian inference using neural networks.

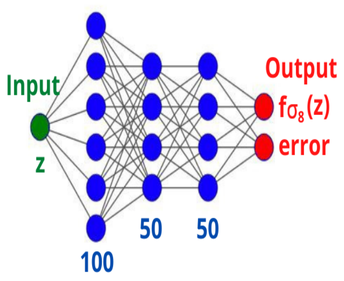
Neural reconstructions

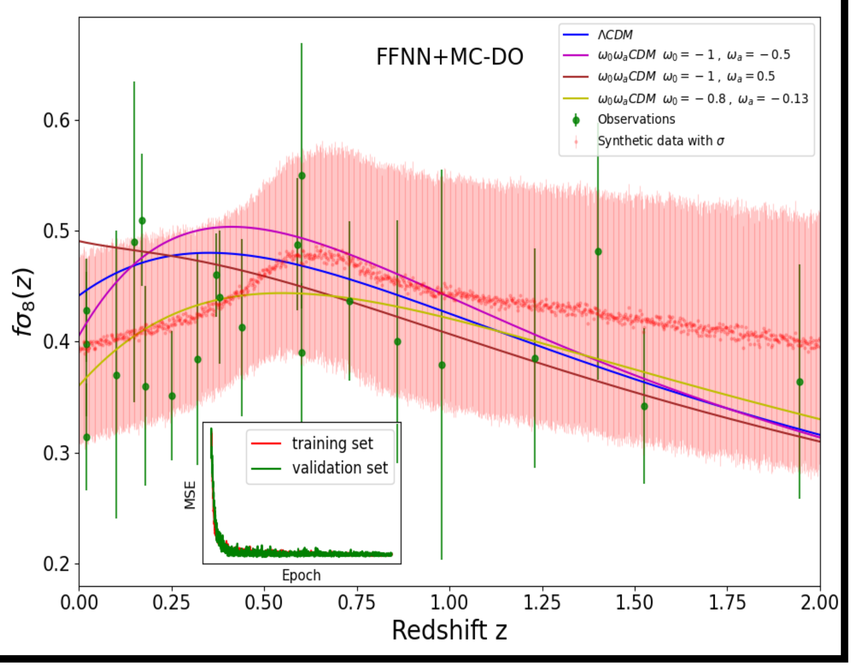
Model-independent reconstructions for cosmological functions with neural networks and Monte Carlo Dropout for small observational datasets, without any statistical or theoretical assumptions.

Gómez-Vargas, et al. (2023). Eur. Phys. J. C 83, 304.



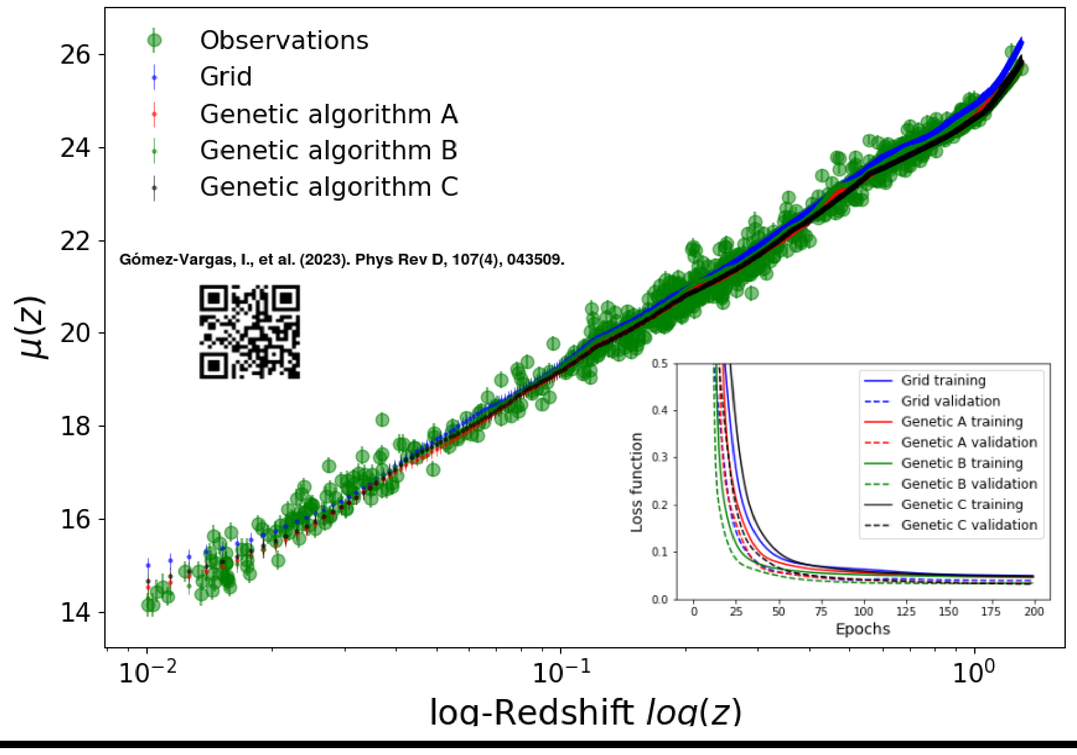






Tunning with genetic algorithms

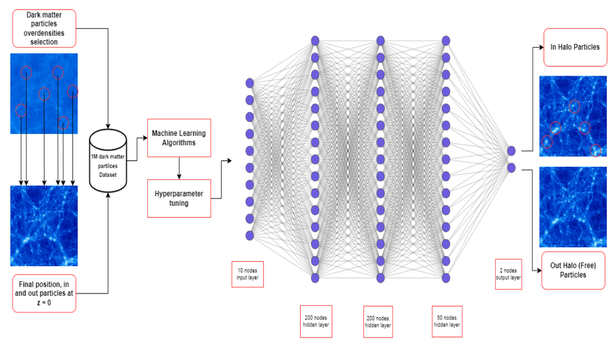
A correct selection of the hyperparameters in a neural network is crucial. A bad neural model could suggest meaningless physical interpretations.

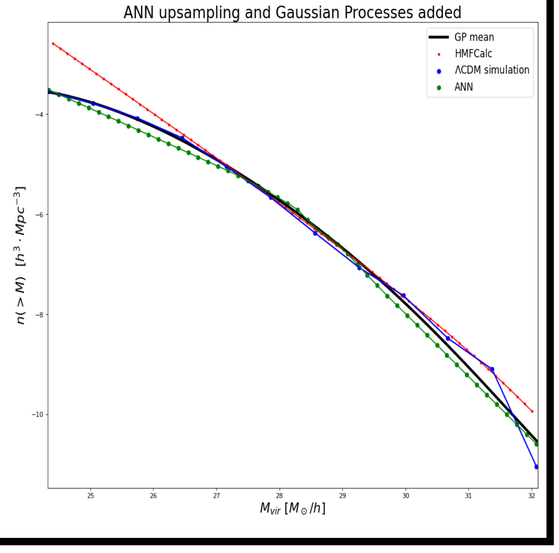


N-body simulations

Classification and regression applications using information from N-body simulations.

Chacón, J., Gómez-Vargas, I., et al (2023). arXiv:2303.09098. Accepted in Phys. Rev. D.



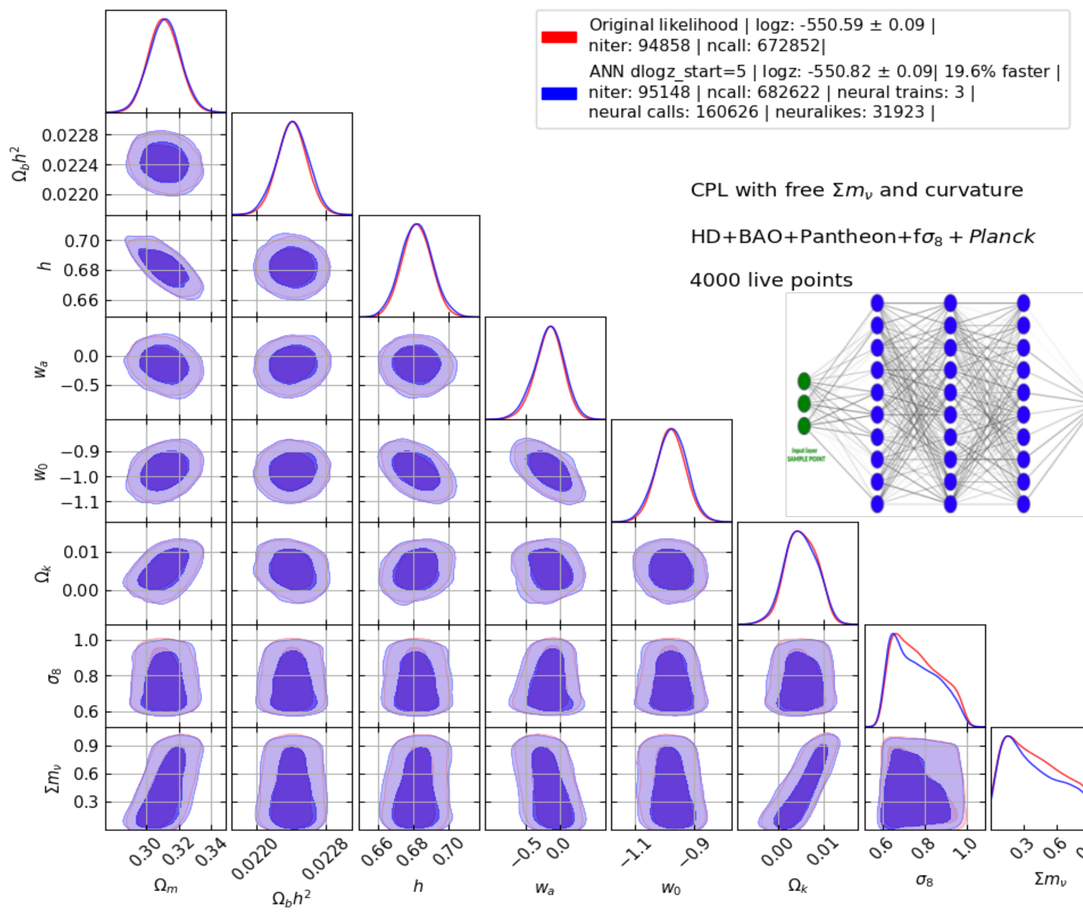


Accelerating Bayesian Inference

We employ neural networks to accelerate the real-time execution of nested sampling. We use `dlogz_start` as a criterion to start the training of the ANN using only the current set of live points. Optionally, genetic algorithms can generate the first live points or find the ANN architecture.

Pseudocode

using_neuralike == False
if *livegenetic == True (optional)* **then**
  Define genetic operators Generate a population P with Nind individuals
  Evolve population through Ngen generations
else
  Generate Nlive live points
for *i in range*(iteration) **do**
  if *(dlogz < dlogz*$_{\text{start}}$*))* **then**
    if *i % N == 0 AND using_neuralike == False* **then**
      Use nlive points as training dataset
      Optional: Use genetic algorithms to choose the ANN architecture
      if *loss function <criterion* **then**
        using_neuralike == True
        $\mathcal{L}$ = ANNmodel
      else
        continue with NS
    if *min(saved_logl) <neural-loglike i <max(saved_logl)* **then**
      continue **else**
        like=logL;
        using_neuralike == False
end



♠Contact

GitHub | https://igomezv.github.io | E-mail: igomez@icf.unam.mx |