Machine Learning Application for Anomalous Galaxy Detection

Fedir Boreiko, Francisco Villaescusa-Navarro

Affiliation 1: Center for Computational Astrophysics, Flatiron Institute, New York NY 10010

Affiliation 2: Department of Physics and Astronomy, University of Manchester, Manchester M13 9PL

Affiliation 3: Department of Physics, National University of Kyiv, Kyiv 01601

Abstract

This project employs Machine Learning techniques to search for outlier galaxies within a multidimensional parameter space, aiming to establish connections between these anomalies and their underlying astrophysical and cosmological parameters. Through this exploration, we not only contribute to anomaly detection but also offer insights that can refine existing models, potentially advancing our comprehension of the universe's formation and evolution processes.

Normalizing Flows

Normalizing Flows - a generative model which learns a probability model by transforming a simple distribution into a more complicated one using a deep network. Normalizing flows can both sample from this distribution and evaluate the probability.

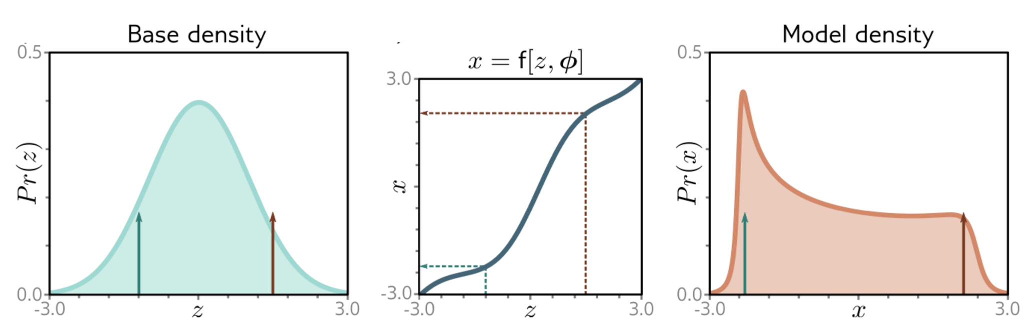

Figure 1: Transforming probability distribution

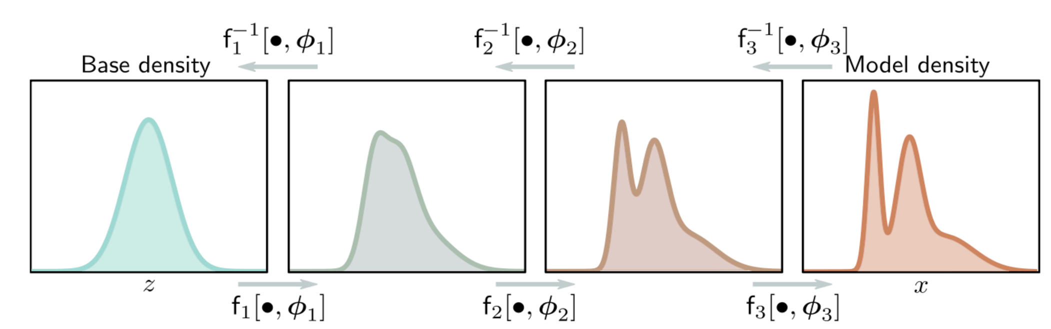

Figure 2: Forward and inverse mappings of a deep neural network

The defining property of flow-based models is: transformation f must be reversible and differentiable.

Given a base probability distribution π_u, the probability of a given point: $p(\mathbf{x}) = \pi_u(\mathbf{z}) \left| \frac{\partial \mathrm{f}[\mathbf{z}, \phi]}{\partial \mathbf{z}} \right|^{-1}$, where $\mathbf{z} = \mathrm{f}^{-1}[\mathbf{x}, \phi]$ - latent.

Negative log-likelihood for one flow with a dataset $\{\mathbf{x}_i\}$: $\hat{\phi} = \max_{\phi} \left[\prod_{i=1}^{I} p(\mathbf{x}_i | \phi) \right] = \min_{\phi} \left[\sum_{i=1}^{I} \left(\log \left| \frac{\partial \mathrm{f}[\mathbf{z}_i, \phi]}{\partial \mathbf{z}_i} \right| - \log \pi_u(\mathbf{z}_i) \right) \right]$.

Forward mapping of a deep neural network (K layers): $\mathbf{x} = \mathrm{f}[\mathbf{z}, \phi] = \mathrm{f}_K[\mathrm{f}_{K-1}[\ldots \mathrm{f}_2[\mathrm{f}_1[\mathbf{z}, \phi_1], \phi_2], \ldots \phi_{K-1}], \phi_K]$,

Determinant of Jacobian of a deep neural network: $\det D\mathrm{f} = \prod_{k=1}^{K} \det D\mathrm{f}_k$,

Negative log-likelihood for K flows: $\hat{\phi} = \min_{\phi} \left[\sum_{i=1}^{I} \left(\sum_{k=1}^{K} \left(\log \left| \frac{\partial \mathrm{f}_k[\mathbf{z}_i, \phi]}{\partial \mathbf{z}_i} \right| \right) - \log \pi_u(\mathbf{z}_i) \right) \right]$.

Autoregressive flows - Normalizing Flow model using the autoregressive property: $p(\mathbf{x}) = \prod_{d=1}^{D} p(x_d | \mathbf{x}_{1:d-1})$, where

Neural Spline - Normalizing Flow model which uses K rational-quadratic functions, with boundaries set by $K+1$ coordinates $\{(x^{(k)}, y^{(k)})\}_{k=0}^{K}$ known as *knots*. The knots monotonically increase between $(x^{(0)}, y^{(0)}) = (-B, -B)$ and $(x^{(K)}, y^{(K)}) = (B, B)$, where B - *boundary* parameter. We give the spline $K-1$ arbitrary positive values $\{\delta^{(k)}\}_{k=1}^{K-1}$ for the derivatives at the internal points and set boundary derivatives $\delta^{(0)} = \delta^{(K)} = 1$ to match the linear tails. Splines create smooth and flexible mappings between simple and complex distributions.

Autoregressive Spline - $p(x_d | x_{1:d-1})$ is modeled as a linear Spline, where the free parameters (knots' x-positions, knots' y-positions, derivatives) are obtained by a neural network as a

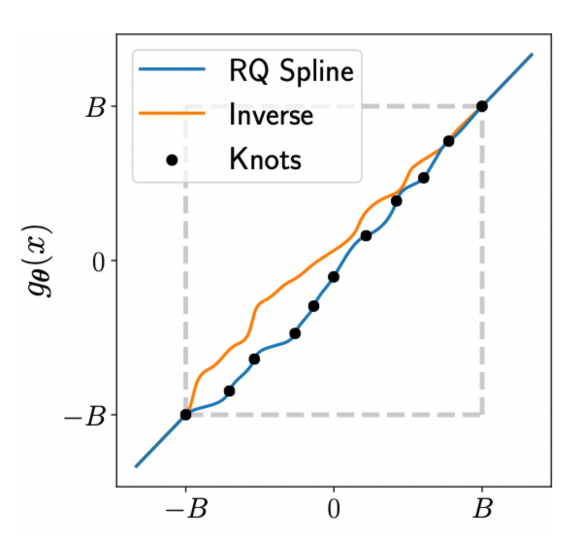

Figure 3: Neural Spline

Data

The galaxy catalogs we used to train, validate, and test our models come from the Cosmology and Astrophysics with MachinE Learning Simulations – CAMELS project - comprehending different setups of hydrodynamical simulations. They have been run with numerous codes that solve the hydrodynamic equations differently and implement various subgrid models: IllustrisTNG, SIMBA, and Astrid. The ones used in the project correspond to the IllustrisTNG model. Specifically, we used the Cosmic Variance (CVs) set, which corresponds to different seeds for the fiducial cosmological and astrophysical parameters: $\Omega_m = 0.3$, $\Omega_b = 0.049$, $h = 0.6711$, $n_s = 0.9624$, $\sigma_8 = 0.8$, $\omega = -1$, $M_\nu = 0\,\mathrm{eV}$. The 030-033 galaxy snapshots used for training have a redshift of 0.15, 0.10, 0.05, and 0.00 respectively.

1-Dimensional case

- Toy Model Architecture: 1 layer of Batch Normalization and Spline Transform (Pyro implementation).
- Base distribution: Gaussian.
- Loss function: Negative log-likelihood (torch implementation), which scores the sample by inverting the transforms and computing the $\log p(\mathrm{Y})$ using $\log p(\mathrm{X}) + \log |\det(\frac{d\mathrm{X}}{d\mathrm{Y}})|$.

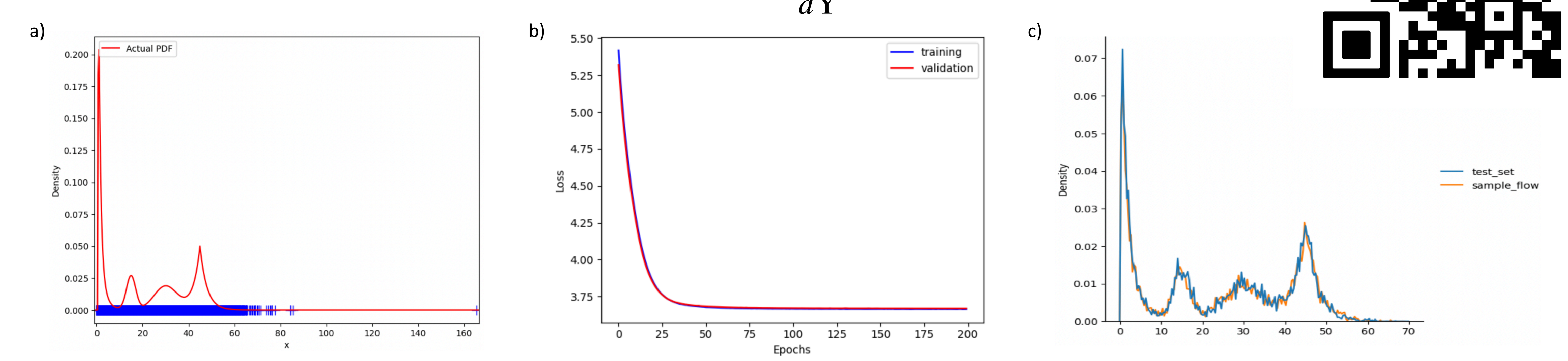

Figure 4: a) generated data from a given probability distribution function, b) training process, c) histogram of test set and sampled data.

2-Dimensional case

The model was trained to investigate the Stellar Mass vs Entire Mass dependency. In blue - data from simulations, in red - sampled data.

- Model Architecture: 3 layers of Autoregressive Spline Transforms (Pyro implementation).
- Base distribution: multivariate Gaussian.
- Loss function: Negative log-likelihood.

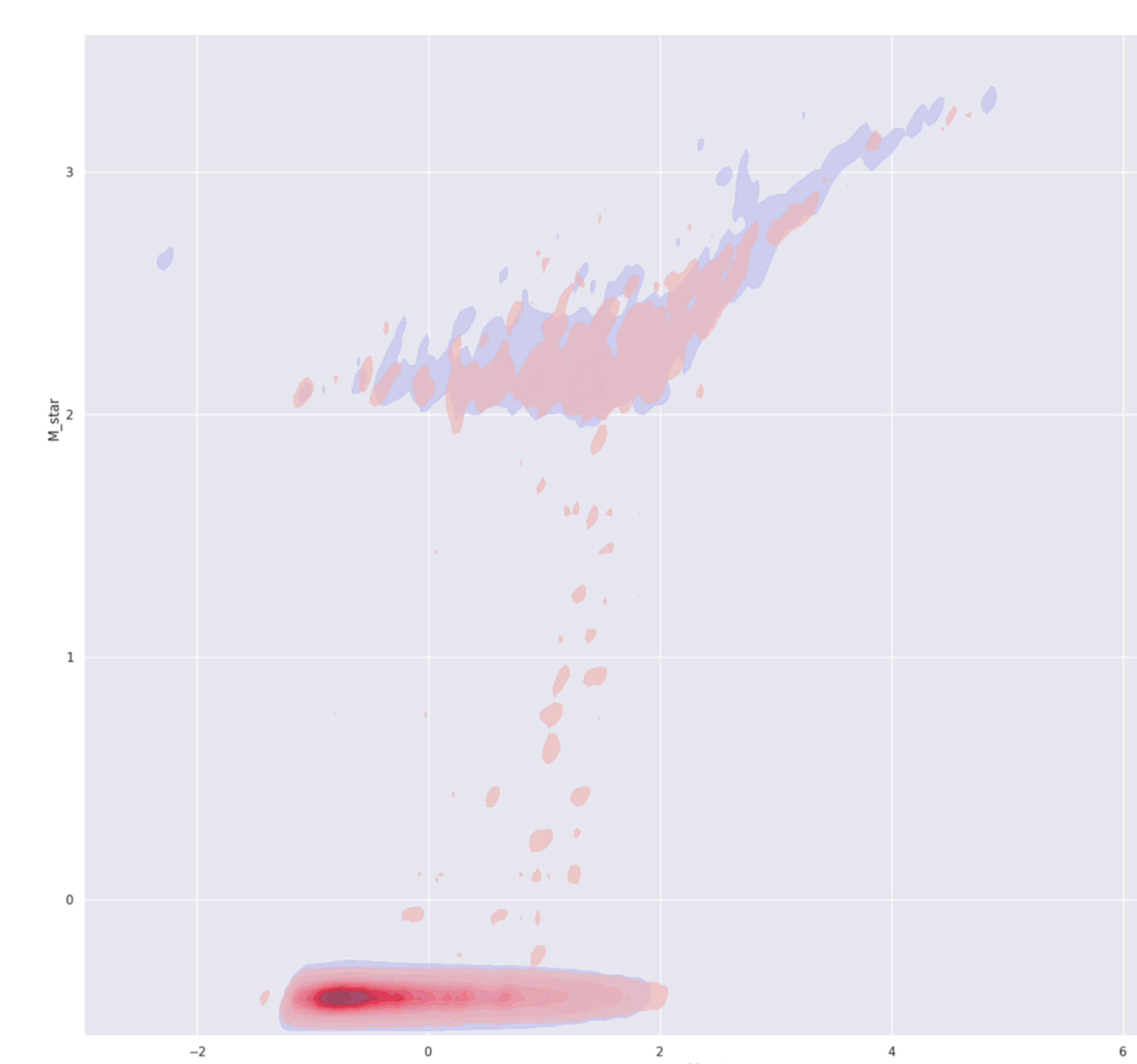

Figure 5: model density estimation

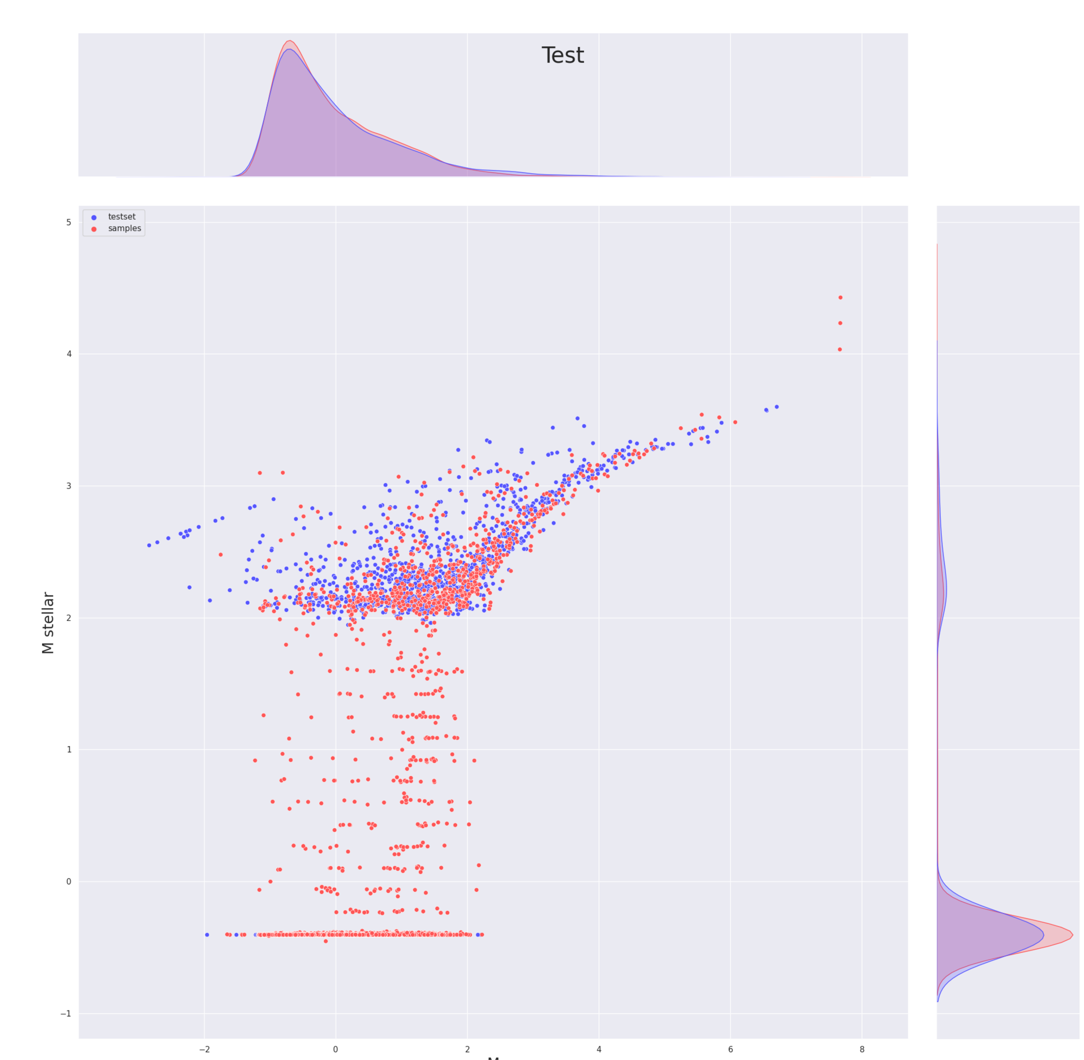

Figure 6: histogram of samples

References

[1] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, 2019.
[2] Simon J.D. Prince, Understanding Deep Learning, 2023.
[3] Francisco Villaescusa-Navarro, Daniel Anglés-Alcázar, Shy Genel, David N. Spergel, Rachel S. Somerville… The CAMELS project: Cosmology and Astrophysics with MachinE Learning Simulations, 2021.

Acknowledgements

This work was graciously supported by the Simons Foundation and accomplished under the supervision and mentorship of Francisco Villaescusa-Navarro and Natalí de Santi.