The Astrophysical Journal, 997:65 (16pp), 2026 January 20 https://doi.org/10.3847/1538-4357/ae201d

Photo-z Estimation with Normalizing Flow

Yiming Ren[1,2], Kwan Chuen Chan[1,2], Le Zhang[1,2], Yin Li[3], Haolin Zhang[1,2], Ruiyu Song[1,2], Yan Gong[4,5,6], Xian-Min Meng[4], and Xingchen Zhou[4]
[1] School of Physics and Astronomy, Sun Yat-Sen University, 2 Daxue Road, Tangjia, Zhuhai 519082, People's Republic of China; chankc@mail.sysu.edu.cn
[2] CSST Science Center for the Guangdong-Hongkong-Macau Greater Bay Area, SYSU, Zhuhai 519082, People's Republic of China
[3] Department of Mathematics and Theory, Peng Cheng Laboratory, Shenzhen, Guangdong 518066, People's Republic of China
[4] National Astronomical Observatories, Chinese Academy of Sciences, 20A Datun Road, Beijing 100012, People's Republic of China
[5] School of Astronomy and Space Sciences, University of Chinese Academy of Sciences (UCAS), Yuquan Road No.19A Beijing 100049, People's Republic of China
[6] Science Center for China Space Station Telescope, National Astronomical Observatories, Chinese Academy of Sciences, 20A Datun Road, Beijing 100101, People's Republic of China
Received 2025 October 15; revised 2025 November 13; accepted 2025 November 14; published 2026 January 14

Abstract

Accurate photometric redshift (photo-z) estimation is a key challenge in cosmology, as uncertainties in photo-z directly limit the scientific return of large-scale structure and weak lensing studies, especially in upcoming Stage IV surveys. The problem is particularly severe for faint galaxies with sparse spectroscopic training data. In this work, we introduce nflow-z, a novel photo-z estimation method using the powerful machine learning technique of normalizing flow. nflow-z explicitly models the redshift probability distribution conditioned on the observables such as fluxes and colors. We build two nflow-z implementations, dubbed cINN and cNSF, and compare their performance. We demonstrate the effectiveness of nflow-z on several datasets, including a CSST mock, the COSMOS2020 catalog, and samples from Dark Energy Survey (DES) Y1, the Sloan Digital Sky Survey, and DESCaLS. Our evaluation against state-of-the-art algorithms shows that nflow-z performs favorably. For instance, cNSF surpasses random forest, multilayer perceptron, and convolutional neural network on the CSST mock test. We also achieve a 30% improvement over official results for the faint DESCaLS sample and outperform conditional generative adversarial network and mixture density network methods on the DES Y1 dataset test. Furthermore, nflow-z is computationally efficient, requiring only a fraction of the computing time of some of the competing algorithms. Our algorithm is particularly effective for the faint sample with sparse training data, making it highly suitable for upcoming Stage IV surveys.

Unified Astronomy Thesaurus concepts: Redshift surveys (1378); Galaxy photometry (611); Photometry (1234)

1. Introduction

Wide-area imaging surveys provide powerful cosmological probes to constrain cosmology. Weak gravitational lensing is a leading application (M. Bartelmann & P. Schneider 2001; C. Heymans et al. 2013; H. Hildebrandt et al. 2017; M. A. Troxel et al. 2018; C. Hikage et al. 2019; M. Asgari et al. 2021; A. Amon et al. 2022; L. F. Secco et al. 2022; R. Dalal et al. 2023; X. Li et al. 2023; A. H. Wright et al. 2025). Angular clustering emerges as another indispensable cosmological probe, and it is often bundled with weak lensing to form the 3×2 point analysis to enhance the cosmological constraining power (T. M. C. Abbott et al. 2018; C. Heymans et al. 2021; T. Abbott et al. 2022; H. Miyatake et al. 2023; S. Sugiyama et al. 2023). As a standalone probe, it can be used to measure the transverse baryon acoustic oscillations (N. Padmanabhan et al. 2007; H.-J. Seo et al. 2012; T. Abbott et al. 2019; T. M. C. Abbott et al. 2022; K. C. Chan et al. 2022; T. M. C. Abbott et al. 2024; R. Song et al. 2024). Ongoing or forthcoming Stage IV surveys will provide an enormous amount of photometric data with a significant increase in depth, leading to substantially more exquisite cosmological findings. Among them, the flagship imaging surveys include the Rubin Observatory Legacy Survey of Space and Time (R. Mandelbaum et al. 2018; Ž. Ivezić et al. 2019), Euclid (R. Laureijs et al. 2011), the Chinese Space Station Survey Telescope (CSST; H. Zhan 2011; Y. Gong et al. 2019; CSST Collaboration et al. 2025), and the Roman Space Telescope (D. Spergel et al. 2015; T. Eifler et al. 2021).

To achieve the impressive cosmological results from imaging surveys, accurate photometric redshift (photo-z) estimation is a prerequisite. photo-zs are often derived from the photometry information measured from a few broadband filters. Two primary approaches for photo-z estimation are template fitting methods and data-driven training techniques (see M. Salvato et al. 2019; J. A. Newman & D. Gruen 2022 for a review).

The template-fitting method (e.g., S. Arnouts et al. 1999; N. Benítez 2000; M. Bolzonella et al. 2000; O. Ilbert et al. 2006) derives photo-zs by fitting observed galaxy colors or magnitudes to spectral energy distributions (SEDs) with the photo-z treated as a free parameter in the model. The fitting process can incorporate prior information. However, the accuracy of this approach depends critically on the completeness and representativeness of the adopted SED templates and the reliability of the priors.

Training-based approaches typically employ machine learning techniques. Various algorithms have been explored, and an incomplete list includes multilayer perceptron (MLP; A. A. Collister & O. Lahav 2004; I. Sadeh et al. 2016), nearest neighbor fitting (J. De Vicente et al. 2016), random forest (S. Carliles et al. 2010; R. Zhou et al. 2021; J. Lu et al. 2024),

(boosted) decision tree (D. W. Gerdes et al. 2010; M. Carrasco Kind & R. J. Brunner 2013), sparse Gaussian process (I. A. Almosallam et al. 2016), self-organizing map (M. Carrasco Kind & R. J. Brunner 2014; D. Masters et al. 2015; R. Buchs et al. 2019; A. H. Wright et al. 2020), convolutional neural network (CNN; B. Hoyle 2016; A. D'Isanto & K. L. Polsterer 2018; S. Schuldt et al. 2021; R. Li et al. 2022; X. Zhou et al. 2022a), Bayesian neural network (X. Zhou et al. 2022b), recurrent neural network (Z. Luo et al. 2024a), and conditional generative adversarial networks (cGAN; M. Garcia-Fernandez 2025). These methods require spectroscopic redshift (spec-z) or sometimes high-quality photo-z data for training, meaning that their accuracy is contingent on both the size and representativeness of the spec-z sample.

In general, if there is little spec-z data available, especially at high redshifts, then we have to rely on template fitting using locally observed or synthetic SEDs. On the other hand, if there are sufficient spec-z training data, the training methods usually deliver higher quality results. Currently, the primary photo-z estimates in all the major large-scale wide field imaging surveys are based on training-based methods. However, there is still room for improvement, especially in the faint magnitude ends, where there are large measurement errors, yet with sparse training spec-z data.

In addition, clustering-based methods provide an independent route to calibrate the true redshift distribution of a photo-z sample. It can be categorized into clustering-z (CZ) and self-calibration (SC). CZ, based on cross correlations with overlapping spectroscopic samples (J. A. Newman 2008; M. McQuinn & M. White 2013; B. Ménard et al. 2013; S. J. Schmidt et al. 2013; J. L. van den Busch et al. 2020) has been has been widely used in current surveys, while SC relies only on photometric autocorrelations and cross correlations (M. Schneider et al. 2006; P. Zhang et al. 2010; L. Zhang et al. 2017; H. Peng et al. 2022; H. Xu et al. 2023). Since spectroscopic coverage becomes sparse at high redshift, CZ calibration is limited there, and SC is expected to play an increasingly important role in future surveys. Combining CZ with SC is an effective means to extend the clustering-based method to high redshift (W. Zheng et al. 2024).

In this work, we apply a powerful machine learning method (MLM), normalizing flow (NF; I. Kobyzev et al. 2020; G. Papamakarios et al. 2021), to estimate photo-z, and call this photo-z estimation framework with NF, nflow-z. NF learns the coordinate transformation to some simple base distribution to approximate the (complex) target distribution. The NF networks are invertible by construction. In particular, the invertible nature of this type of network was emphasized in L. Ardizzone et al. (2018, 2019). Degeneracy in the redshift–photometry relation is a key factor limiting the accuracy of the photo-z estimation. This problem is at best tackled implicitly in most of the existing methods. By addressing the degeneracy issue directly, nflow-z holds the promise to improve the photo-z estimation accuracy. Moreover, nflow-z directly models the probability distribution of the data, and it is expected to yield an accurate error estimate. We note that J. F. Crenshaw et al. (2024) use the NF to estimate the joint probability distribution between redshift and magnitudes in order to forward model the photometric galaxy catalog. The redshift estimation is then obtained by marginalizing over the extra degrees of freedom. Instead, we apply the conditional network to estimate photo-z by treating the photometry information as a condition. This simple network structure enables us to focus on the redshift information and eliminate the extra degrees of freedom effectively. Our network structure is similar to Z. Sun et al. (2023), which briefly mentions the NF method and applies it to estimate photo-z on the HSC-SSP survey PDR3 data (H. Aihara et al. 2022). Here, we study the performance of nflow-z in detail, and we contrast the results for two different NF architectures. We test nflow-z on a number of datasets and compare them against other state-of-the-art algorithms. Plus, a primary motivation of this work is to prepare a photo-z estimation pipeline for CSST.

This paper is organized as follows. In Section 2, we review the general idea of NF, present the nflow-z framework, and describe two principal architectures to implement nflow-z. We test nflow-z with different datasets and compare our test results against those from other methods in Section 3. We conclude in Section 4.

2. Normalizing Flow and Its Implementations

2.1. A Brief Review on NF

NF (E. G. Tabak & C. V. Turner 2013; D. Rezende & S. Mohamed 2015) is a powerful MLM capable of estimating a complex high-dimensional probability distribution. Here, we briefly describe its key ideas and refer the reader to S. J. Prince (2023), I. Kobyzev et al. (2020), and G. Papamakarios et al. (2021) for more details.

The central idea of NF is to learn a coordinate transformation g that maps a complex target distribution $p_X(x)$ to some tractable base distribution $p_Y(y)$. Mathematically, with the bijective differentiable transformation $x = g(y;c)$, we have

$$p_X(x;\, c)dx = p_X(x;\, c)\left|\frac{dx}{dy}\right| dy = p_X(g(y;\, c);\, c)\left|\frac{dg}{dy}\right| dy, \tag{1}$$

where p_X describes the probability distribution in physical space and $|dg/dy|$ is the Jacobian determinant. In this work, the physical space is taken to be the redshift space, i.e., $x = z$. Our notation reflects that our target distribution is in the form of a 1D conditional distribution with the condition denoted by c. In our case, the condition represents the observables such as fluxes and colors. We then demand that the base distribution in latent space, p_Y to satisfy [7]

$$p_Y(y) = p_X(g(y;\, c);\, c)\left|\frac{dg}{dy}\right|. \tag{2}$$

In words, the target distribution is normalized to the base distribution by the coordinate transformation (flow). The base distribution is chosen to be some tractable probability distribution, and the standard normal (or Gaussian) distribution, which is sometimes called noise in the machine learning literature, is commonly adopted. We stick to the normal distribution in this work. Transformation g is also a function of the network parameters, which are learned in the training process, so that Equation (2) is satisfied to a high accuracy.

[7] In the machine learning literature, the latent space variable is often denoted as z. Since redshift is the key subject in this work, we reserve the notation z for redshift.

We train the network parameters by maximizing the likelihood of the data

$$p_X(x; c) = p_Y(f(x; c)) \left| \frac{df}{dx} \right|, \tag{3}$$

where $y = g^{-1}(x) \equiv f(x)$. Note that, in this case, we need to evaluate f, while to generate samples, we use g instead:

$$y \sim p_Y(y), \quad x = g(y). \tag{4}$$

The key to successful probability distribution modeling is to be able to learn a general bijective differentiable transformation. However, for the method to be practical, the model needs to perform both the inference and sampling efficiently. This implies that the transformation g, its inverse f, and the Jacobian determinant must be efficiently calculated.

An approach to realize an invertible differentiable function of sufficient complexity is through the composition of simpler invertible differentiable functions. More precisely, if

$$g = g_N \circ g_{N-1} \circ \cdots \circ g_1, \tag{5}$$

with g_i being an invertible function, then its inverse is also invertible and is given by

$$f = f_1 \circ f_2 \circ \cdots \circ f_N, \tag{6}$$

where $f_i = g_i^{-1}$. With the help of the chain rule, the corresponding Jacobian determinant can be conveniently written as

$$\left| \frac{df}{dx} \right| = \left| \frac{df_1}{df_2} \right| \left| \frac{df_2}{df_3} \right| \cdots \left| \frac{df_N}{dx} \right|. \tag{7}$$

This composition is implemented using layers of networks in deep learning. Each layer represents a differentiable invertible function. We discuss two types of architecture to construct f_i in Section 2.2.

Both NF and the generative adversarial network (GAN; I. J. Goodfellow et al. 2014) are well-known members of the generative modeling family in deep learning. Compared to GAN, NF has the advantage of exact probability distribution modeling and is relatively easy to train. In the literature, NF is often perceived to deliver less dazzling results than GAN, but this is often framed in the context of high-dimensional problems, such as image generation. In Section 3.3, we demonstrate that our NF results are superior to the GAN ones.

2.2. Implementation of nflow-z

An illustration of the nflow-z network structure is shown in Figure 1. The physical space X is a 1D redshift space, and the latent space is also 1D. The observables, including fluxes, colors, and their error bars, play the role of conditions. The conditions are injected into some neural networks, whose output then serves as the transformation parameters for the function f_i. The direction of the arrows indicates the function evaluation required: f for forward modeling and g for backward sampling.

Differences in architecture implementation lie in the central NF block. Many efforts have been devoted to constructing an expressive yet easy-to-compute function f_i. Here, we consider two widely used implementations of NF, the conditional invertible neural network (cINN; L. Ardizzone et al. 2019) and conditional neural spline flow (cNSF; C. Durkan et al. 2019; H. M. Dolatabadi et al. 2020). Roughly speaking, in cINN, f_i is

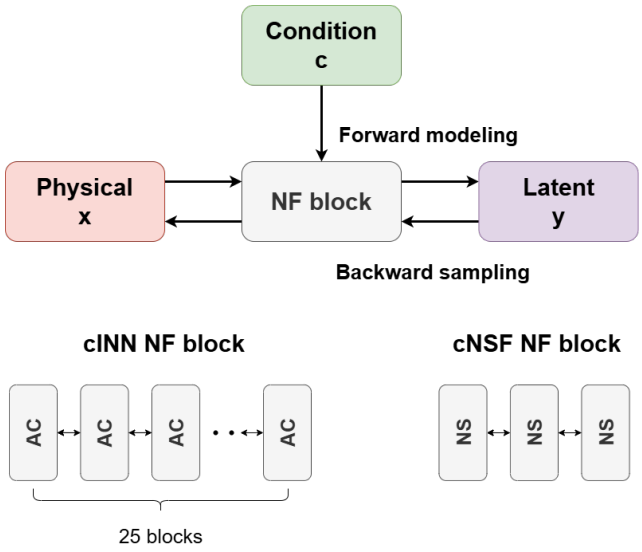

Figure 1. The general framework of the nflow-z network is shown in the upper part of the figure. The schematic structure of the two implementations of the NF block, cINN and cNSF, is shown in the lower part of the figure.

implemented using an affine transformation and many layers (25 here) are used, while in cNSF, each layer is realized by a more sophisticated spline transformation and thus fewer layers (3 here) are necessary.

2.2.1. cINN

cINN is proposed in L. Ardizzone et al. (2019), which is built upon the previous version, INN (L. Ardizzone et al. 2018). The software `FrEIA` (L. Ardizzone et al. 2018–2022) offers a flexible framework to construct such a neural network. This architecture has been used in various inverse problems in astronomy, including stellar parameter determination (V. F. Ksoll et al. 2020; D. E. Kang et al. 2022), cosmic ray source inference (T. Bister et al. 2022), exoplanet characterization (J. Haldemann et al. 2023), and radio sky map making (H. Zhang et al. 2023).

Following the original cINN (L. Ardizzone et al. 2019), we implement the NF block using the GLOW coupling block (L. Dinh et al. 2016; D. P. Kingma & P. Dhariwal 2018). In the GLOW framework, each data vector is split into two parts, and a set of affine coupling transformations is applied to them. The merit of this transformation is that g, f, and their associated Jacobian determinant can be computed efficiently. The operation, together with the permutation among dimensions, enables different dimensions of the data vector to couple with each other.

The GLOW affine coupling transformations are based on the real nonvolume preserving transformation developed by L. Dinh et al. (2016). In this algorithm, to allow the components of the input array to be coupled, the input data array is split into two parts as (u_1, u_2), and the forward affine transformation from (u_1, u_2) to (v_1, v_2) is then given by

$$v_1 = u_1 \exp[s_2(u_2)] + t_2(u_2), \tag{8}$$

$$v_2 = u_2 \exp[s_1(v_1)] + t_1(v_1), \tag{9}$$

where s_i and t_i are arbitrary functions of the condition c, implemented using neural networks. The structure of the transformation is so constructed that the inverse transformation

can be easily written down as well:

$$u_2 = (v_2 - t_1(v_1))\exp[-s_1(v_1)], \tag{10}$$

$$u_1 = (v_1 - t_2(u_2))\exp[-s_2(u_2)]. \tag{11}$$

In particular, the general functions s_i and t_i do not need to be inverted. The log-determinant of the transformation is simply equal to $s_2(u_2) + s_1(v_1)$. This can be shown by performing the transformation (Equations (8) and (9)) in two steps: $(u_1, u_2) \to (v_1, u_2) \to (v_1, v_2)$. However, our input data array is 1D (a redshift); to directly match this algorithm, we duplicate it to form a 2D array. We point out that the `FrEIA` implementation still works for a 1D input without duplication, but we find that its performance is not as good as the 2D case. Thus, we opt to use the 2D setting, at the expense of more computation time.

Our cINN NF block consists of 25 layers, with each layer being a GLOW affine coupling block. The scale function s_i and shift function t_i in each block are modeled by a two-layer fully connected neural network with 512 hidden units and ReLU activation functions. The conditional inputs are injected into every block through the `FrEIA` ConditionNode mechanism.

2.2.2. cNSF

We consider an alternative implementation using the rational spline bijections of linear order (C. Durkan et al. 2019; H. M. Dolatabadi et al. 2020) implemented in `pyro` (E. Bingham et al. 2019; D. Phan et al. 2019). We call this architecture cNSF. In place of the affine transformation in cINN, f is realized by the piecewise rational spline. To model the spline transform, the interval within some bounds $[-B, B]$ is divided into $K - 1$ bins by K points, known as knots. Within each bin, the transformation function is given by a rational spline of linear order function, whose numerator and denominator are linear functions. The function in the whole interval $[-B, B]$ must be monotonically increasing. The transformation parameters include the position of the nodes and the derivatives at the knots. In particular, in the formulation of H. M. Dolatabadi et al. (2020), the piecewise rational linear spline is differentiable at the knots as well. As in cINN, in the `pyro` implementation, the conditions are incorporated by feeding them into a shallow neural network, whose output are the parameters for the transformation f. We refer the readers to the original papers (C. Durkan et al. 2019; H. M. Dolatabadi et al. 2020) for more details on the transformation.

The advantage of the rational spline transformation is that it provides a more flexible parameterization than the affine transformation due to the division into bins by the knots and the usage of the rational linear function. Yet with these complexities, the inversion and derivative can still be written in closed form and hence can be computed efficiently.

Our cNSF block consists of three layers, and each is implemented by a linear rational spline flow block, within which we have used eight segments for the spline transform. In each block, the conditions are injected into an MLP network of size $32 \times 32 \times 32$, which returns the transformation parameters for the spline flow transformation.

2.3. Setting of the Network Parameters

Here, we mention some technical details of the implementations and describe the setting of the network parameters. Unless otherwise specified, they apply to both implementations.

To put all the data, both the redshift and the condition data, such as fluxes, on a similar scale, we preprocess the data by the standard scaler transformation $(d - \mu)/\sigma$ on the data vector d, where μ and σ are the mean and the standard deviation of d, respectively. The full dataset is split into training, validation, and test parts with the proportion of 8:1:1. In both implementations, the training data are further divided into batches with 256 samples in each batch.

Recall that Equation (3) enables us to transform the likelihood of the data to some tractable distribution p_Y, which is taken to be the normal distribution. However, the transformation to reach this final distribution is unknown; at least it has to be solved for iteratively. To proceed, we assume that we can treat p_Y as normal, and this approximation holds best when the likelihood is maximized for a given NF model. The exactness of this approximation depends on the capacity of the NF model. It is convenient to consider the loss function in the form of the negative of the log-likelihood, and we have

$$-\log p_X(x; c) = \frac{f^2(x; c)}{2} - \log \left| \frac{df}{dx} \right| + \text{const.} \tag{12}$$

The desired loss function is obtained by averaging over the individual loss function of the samples (and ignoring the irrelevant constant) as

$$\mathcal{L} = \frac{1}{N}\sum_i \left(\frac{f_i^2(x_i; c_i)}{2} - \log \left| \frac{df_i}{dx_i} \right| \right), \tag{13}$$

where N is the total number of samples. For the cINN case with duplication, f_i is a 2D array, and Equation (13) is generalized straightforwardly.

Adam optimizer (D. P. Kingma & J. Ba 2015) is used to train the network by minimizing $\mathcal{L}$. The initial learning rate is set to 10^{-3} and is subsequently reduced by a factor of 10^{-1} after 150 and 250 epochs, respectively.

It is crucial not to overfit the network; otherwise, the trained model becomes sensitive to the fluctuations in the training data and thus does not generalize well to new data. We use the validation data to check for overfitting. In the event of overfitting, the loss function for the training data keeps on decreasing while the validation loss starts to rise. We evaluate the validation loss at the end of each epoch by averaging over the results from all the batches. The training is terminated if the validation loss stays higher than the lowest recorded validation loss for 40 consecutive evaluations. Moreover, to prevent excessive computation, we set the maximum number of training epochs to 400. In either case, the model corresponding to the lowest validation loss is selected.

NF naturally produces the probability distribution function (abbreviated as PDF) for the redshift estimate. After training, we can make predictions by inserting the observables as conditions and generating samples from the latent space. To get a smooth estimate of the PDF, 20,000 samples are generated for each object. From the PDF, different point estimates can be calculated, including the mean, the median, and the mode of the distribution. We find that the median generally leads to the most accurate results, and so it is

adopted as the default point estimate below. For the error estimate, we adopt the half-width of the 68th percentile about the median rather than the standard deviation, as the former gives a more reliable error estimate.

We run the codes on the NVIDIA GeForce RTX 3080 GPU. Using the CSST mock dataset as an example, the cINN model typically finishes training at around 350 epochs, while the cNSF requires about 200 epochs. The total time, training plus testing, is quite modest, approximately 25 minutes for cINN and 6 minutes for cNSF.

3. Algorithm Testing with Datasets

In this section, we test nflow-z using different datasets and contrast the results with other algorithms. We conduct tests on four different datasets: a CSST mock, COSMOS2020 catalogs, an Sloan Digital Sky Survey (SDSS) data sample, and a Dark Energy Survey (DES) Y1 sample, with special emphasis on the CSST mock. These diverse datasets differ in the redshift span, magnitude ranges, and size of the training sample, and hence, these help to test the algorithm under different situations. When comparing with other algorithms, if possible, we use the results obtained by others on the same dataset. This avoids misusing others' algorithms or running their code with unoptimized parameters.

3.1. Testing on the CSST Mock Catalog

3.1.1. CSST Mock Catalog Overview

The first dataset that we consider is a CSST mock catalog. The CSST survey is a stage IV galaxy survey (H. Zhan 2011; Y. Cao et al. 2018; Y. Gong et al. 2019; CSST Collaboration et al. 2025), conducted on a two-meter telescope onboard a satellite orbiting in tandem with the China Space Station. The satellite is expected to be launched in 2027, and is scheduled to operate over a 10-year period covering 17,500 deg^2. Among its tasks is the wide field cosmological survey, which is realized by a slitless spectroscopy device and a multiband photometric instrument. In the wide-field photometric survey, there are seven broadband filters: *NUV*, *u*, *g*, *r*, *i*, *z*, and *y*. Their wavelengths range from 250 to 1000 nm, and 5σ limiting magnitudes are 25.4, 25.4, 26.3, 26.0, 25.9, 25.2, and 24.4, respectively. Further details on the filter properties can be found in Y. Cao et al. (2018). A primary goal of this work is to develop an effective photo-z estimation code for the CSST.

We test our method on a CSST mock catalog, which was first used in X. Zhou et al. (2021). This catalog includes realistic mock images consistent with the CSST observing conditions, and has subsequently been used in a number of other CSST photo-z preparation studies (X. Zhou et al. 2022a; J. Lu et al. 2024; Z. Luo et al. 2024a). Since we only use the flux information rather than the full image, we limit ourselves to describing only the generation of the flux data.

This catalog is built upon the COSMOS2015 catalog (C. Laigle et al. 2016). On galaxies with reliable photometry measurements in the COSMOS2015 catalog, SED fitting is performed to determine their SED properties. The SED fitting is done through `LePhare` (S. Arnouts et al. 2002; O. Ilbert et al. 2006) with the photo-z fixed to be the fiducial value from COSMOS2015. The SED templates used are modified from the original 31 templates from `LePhare` following the prescription in Y. Cao et al. (2018). First, the SEDs are extended from about 90 to 900 Å in order to fit galaxies with $z > 2$. Second, the elliptical and spiral galaxy templates are interpolated so that the final number of SED templates totals 37. Furthermore, the emission lines Lyα, Hα, Hβ, [O II], and [O III] are included in the continuum spectra by `LePhare`.

Armed with the best fit SED, S, the expected total photon count in a band j can be calculated as

$$F_j = t_j A_j N_j \int d\lambda \frac{\lambda}{hc} S(\lambda)\tau_j(\lambda), \tag{14}$$

where τ_j is total system throughput including the intrinsic filter transmission, detector quantum efficiency, and total mirror efficiency, respectively (Y. Cao et al. 2018; X. Zhou et al. 2022a), while h and c are Planck constant and light speed, respectively. The prefactors t_j, A_j, and N_j denote the exposure time, the effective telescope aperture area, and the number of exposures for this band, respectively. The Poisson noise and background noise are introduced in the flux modeling. The latter incorporates the effects of dark current, readout noise, and sky background (see X. Zhou et al. 2022a for more details).

The full catalog consists of 99,095 galaxies. Following X. Zhou et al. (2022a), we select high-quality galaxies with a signal-to-noise ratio larger than 10 in the i or g bands. After this quality cut, we end up with 44,991 galaxy samples, and we use these galaxies in the following analyses. In Figure 2, we plot the redshift and magnitude distribution of this sample.

Our input condition data include fluxes and their errors in seven bands, and the six colors computed from them. However, some fluxes are negative due to error fluctuations, and direct computation of the magnitude and hence color is not possible. Instead, we use the asinh magnitude (R. H. Lupton et al. 1999), which is well-defined even for negative flux.

We plot the loss function (Equation (13)) for the training and validation samples from a typical run in Figure 3. The results shown are from cNSF, and cINN shares a similar trend. At the beginning, both loss functions decrease rapidly as the epoch increases, and the reduction becomes mild after $\sim$100 epochs. The sudden drop at epoch 150 corresponds to the reduction of the learning rate (by a factor of 10^{-1}). After that, the validation loss merely drops, although the training loss still keeps on decreasing. The training is terminated at epoch 191 when the average validation loss starts to rise.

3.1.2. Quantification of the Photo-z Accuracy

We quantify the accuracy of the results using the following metrics, which are commonly used in photo-z estimation analyses:

1. *Bias, b.* the median of the distribution of $\Delta z \equiv z_{\rm phot} - z_{\rm ref}$, where $z_{\rm phot}$ and $z_{\rm ref}$ denote the estimated photo-z and the reference redshift. The latter is either the spec-z or high-quality photo-z.
2. *Normalized median absolute deviation*, $\sigma_{\rm NMAD}$. $1.48 \times {\rm median}(|\Delta z - {\rm median}(\Delta z)|/(1 + z_{\rm ref}))$ (G. B. Brammer et al. 2008), which quantifies the spread of $\Delta z/(1 + z_{\rm ref})$, and it is not sensitive to extreme outliers.
3. *Outlier fraction*, η. The fraction of galaxies with $|\Delta z|/(1 + z_{\rm ref}) > 0.15$. This encodes the significance of catastrophic photo-z error by measuring the fraction of

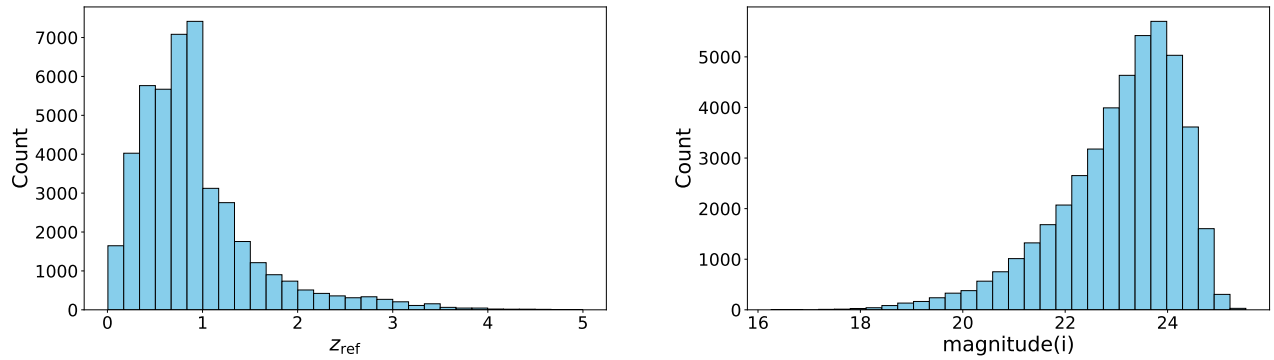

Figure 2. Redshift (left) and i-band magnitude (right) distribution of the CSST mock catalog.

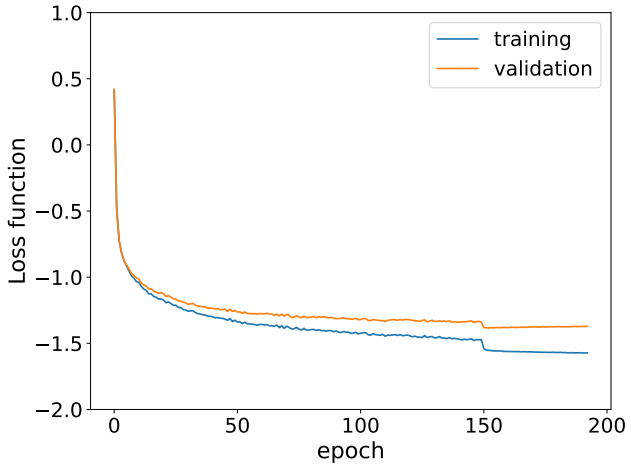

Figure 3. Loss function for the training (blue) and validation (orange) samples as a function of the epoch. The reduction in learning rate at epoch 150 causes the sudden drops in the loss functions. The training is stopped when the average validation loss starts to rise at epoch 191.

galaxies with substantial deviation from the reference redshift.

In Figure 4, we show the scatter plot between the reference redshift and the photo-z estimated by cINN and cNSF. The metrics b, $\sigma_{\rm NMAD}$, and η are also printed. We see that cNSF gives lower η and $\sigma_{\rm NMAD}$, but cINN has a smaller bias.

As we mentioned, a number of other MLMs have been applied to this dataset. Here, we contrast ours with other available results:

1. cINN: $\sigma_{\rm NMAD} = 0.018 \pm 0.0004$, $\eta = 1.60\% \pm 0.16\%$ (this work);
2. cNSF: $\sigma_{\rm NMAD} = 0.017 \pm 0.002$, $\eta = 0.83\% \pm 0.07\%$ (this work);
3. Random forest: $\sigma_{\rm NMAD} = 0.025$, $\eta = 2.0\%$ (J. Lu et al. 2024);
4. MLP: $\sigma_{\rm NMAD} = 0.023$, $\eta = 1.4\%$ (X. Zhou et al. 2022a);
5. CNN: $\sigma_{\rm NMAD} = 0.025$, $\eta = 1.2\%$ (X. Zhou et al. 2022a);
6. Hybrid transfer network: $\sigma_{\rm NMAD} = 0.020$, $\eta = 0.90\%$ (X. Zhou et al. 2022a);
7. Recurrent neural network: $\sigma_{\rm NMAD} = 0.027$, $\eta = 1.6\%$ (Z. Luo et al. 2024a).

To quantify the level of fluctuations, for cINN and cNSF, we have shown the mean and the associated standard deviation from 10 runs. Among these methods, the most sophisticated network is the hybrid transfer network, which combines the Bayesian CNN and Bayesian MLP so that it takes both images and flux data as input. It is also the best performer among the non-NF methods. We find that the performance of cINN is comparable to hybrid transfer network, with $\sigma_{\rm NMAD}$ being marginally better but η less appealing. Remarkably, cNSF achieves the best results across these two metrics.

It is worthwhile commenting on the key properties of nflow-z here. First, the redshift–color degeneracy is a key physical factor limiting the accuracy of the photo-z estimation. nflow-z explicitly accounts for it by assigning a latent variable to each training galaxy sample. To elaborate this point further, suppose that there is a degenerate point in the color space, so that many galaxies with different redshifts are collapsed to this point. In NF, each galaxy training sample (z) is assigned a unique redshift distribution corresponding to a latent variable y. This degenerate point in color space is effectively lifted into some higher-dimensional surface in the latent space. By resolving the degeneracy point, its impact on the photo-z estimation is contained. This issue is not addressed in other methods, at least not explicitly. Second, the nflow-z network structure is relatively simple, and so the redshift information is highly concentrated, while the redshift information can be diluted in more complicated models due to an excessively large number of degrees of freedom. This comment is especially in reference to the finding that nflow-z outperforms CNN. In principle, all the information, including the flux and color information, is contained in the images. Ideally, the algorithms like CNN that take the whole image as input should deliver better results than nflow-z, which uses only the distilled information from the image. However, it is conceivable that the degrees of freedom in the algorithm are so massive that the redshift information may not be effectively extracted. On the other hand, the usage of the distilled information can help the AI algorithm concentrate on the most physically relevant information. The redshift information in nflow-z is highly concentrated in the sense that the latent variables encode only the redshift distribution information. Practically, a simple architecture structure means that it takes much less time to run and is easy to tune its parameters to optimize the results.

3.1.3. Accuracy of the Error Bar and PDF Estimation

We now check the accuracy of the error estimation using the normalized deviation

$$\epsilon \equiv \frac{\Delta z}{\sigma}, \tag{15}$$

where σ is the error estimate of the photo-z. In the ideal case, we anticipate the error bar to give an accurate estimate of the

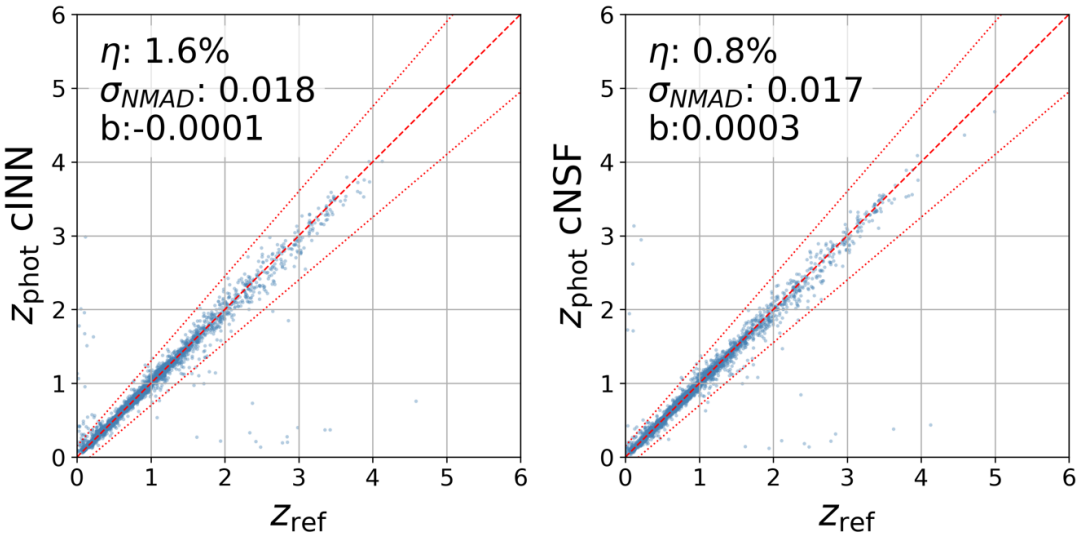

Figure 4. Scatter plot between the spectroscopic redshift, $z_{\rm spec}$, and the photo-z, $z_{\rm phot}$, estimated by cINN (left panel) and cNSF (right panel), respectively. The diagonal line indicates perfect prediction, and the accompanying dotted lines depict the boundaries for the outlier fraction.

deviation Δz, and ϵ follows a standard normal distribution. In Figure 5, we show ϵ for both cINN and cNSF. Overall, both distributions are in good agreement with the standard normal. Closer inspection reveals that the cNSF result is slightly wider than the standard normal, suggesting that the error bars are slightly underestimated.

Probability integral transform (PIT; R. Bordoloi et al. 2010) offers a model-independent way to verify if the PDF derived from the model is consistent with the data. Mathematically, PIT is defined as

$$\mathrm{PIT}(z_{\rm ref}) = \int_0^{z_{\rm ref}} P(z')dz', \tag{16}$$

where P is the estimated PDF. The idea behind it is that if the probability distribution is correct, then the probability that a galaxy appears should be uniform when the distance is measured in terms of the cumulative probability. We plot the PIT for cINN and cNSF in Figure 6. Overall, both PITs are close to the uniform distribution. In greater detail, the cINN PIT is more consistent with the uniform distribution, while the cNSF PIT peaks at both ends with a mild trough at the center. The cNSF PIT pattern indicates that the derived PDF tends to underestimate the width of the distribution. This trend is consistent with the implication from Figure 5 that cNSF slightly underestimates the error bar.

We now examine the characteristics of the PDFs, in particular, if the distribution exhibits multimodal behavior. For convenience, we first define $\delta = |\Delta z|/(1+z_{\rm ref})$ and consider its values in a few different ranges: [0, 0.01], [0.01, 0.02], [0.02, 0.05], [0.05, 0.1], [0.1, 0.15], and [0.15, ∞]. We show the PDFs for a few representative examples in Figure 7. Among the cases examined, the cINN PDF exhibits only a single peak for $\delta \lesssim 0.05$, and it has a small probability ($\sim$5%) showing a double-peak feature for $\delta \gtrsim 0.05$. For cNSF, the PDF is also single-peak only for $\delta \lesssim 0.05$, but it has a more appreciable chance ($\sim$20%) of exhibiting a double-peak feature for $\delta \gtrsim 0.05$.

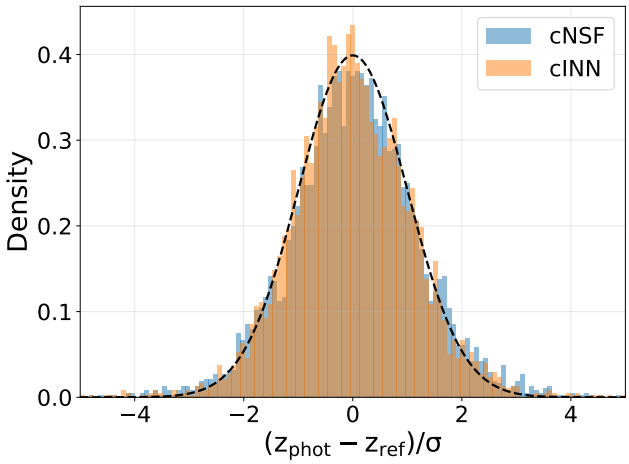

Figure 5. Distribution of the normalized deviation ϵ for both cINN (orange) and cNSF (blue). The standard normal (black dashed) is overplotted for comparison.

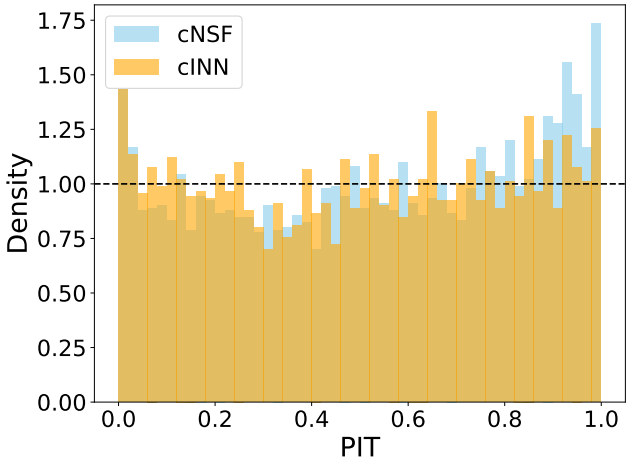

Figure 6. PIT transform for the photo-z PDF obtained from cINN (orange) and cNSF (blue). If the PDF is exact, then the PIT transform yields a uniform distribution, which is indicated as a dashed line.

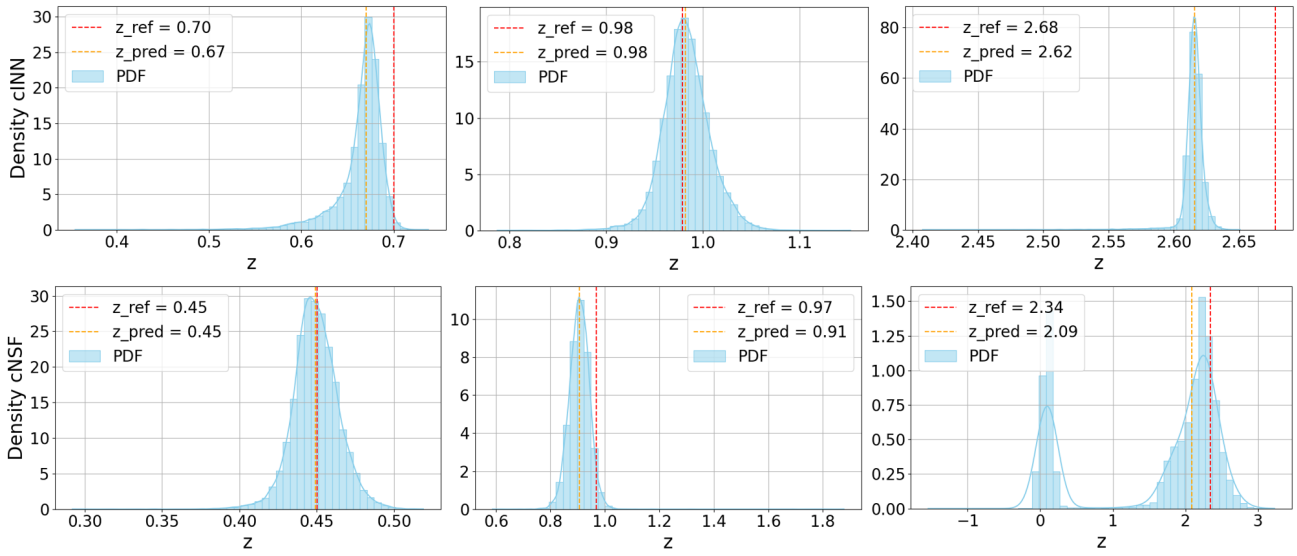

Figure 7. Some representative PDFs (histograms) obtained from cINN (upper panel) and cNSF (lower panel). The true spec-z and the NF predicted point estimate are indicated as red and orange dashed lines, respectively. Both cINN and cNSF predominantly show only a single peak when the prediction is in good agreement with the true value; however, they may show a double-peak feature when the difference is large.

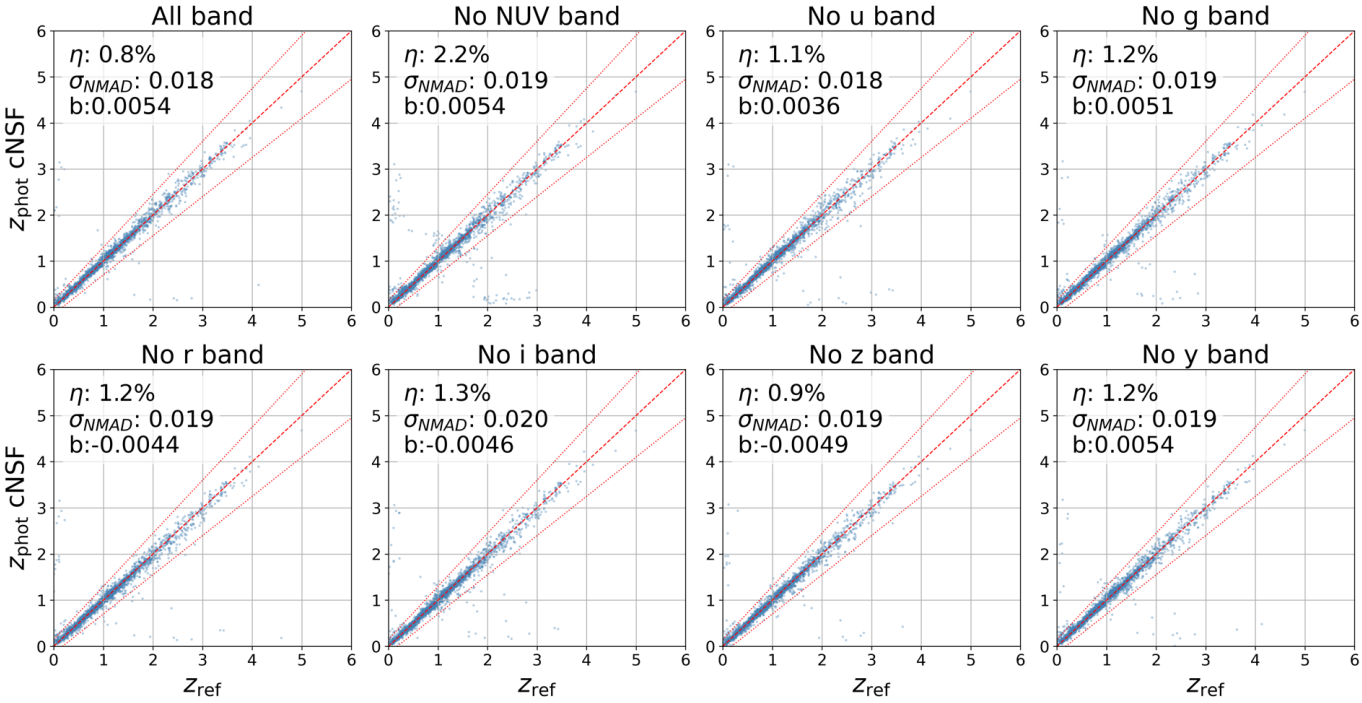

Figure 8. Impact of removing one of the photometric bands. Shown are the scatter plots between spec-z and photo-z estimated with full band information or one of the bands missing. The photo-zs are derived from cNSF.

3.1.4. The Importance of the Individual Bands

It is instructive to check the importance of various bands in the performance of photo-z estimation. To do so, we consider the scenario when one of the photo-z band data is missing.

In Figure 8, we show the scatter plot between the spec-z and cNSF photo-z using photometric data with full band data or one of the bands missing. Note that for the missing band case, we use the data with one band missing in both training and testing. The trend for cINN is broadly similar. We find that missing a single band does not significantly influence the performance. Among the bands, while missing i, g, or r gives a noticeable increase in η and $\sigma_{\rm NMAD}$, missing *NUV* leads to the most significant impact, with $\sigma_{\rm NMAD}$ and especially η boosted most substantially. We note that the *NUV* band is particularly effective in distinguishing the Lyman break from the Balmer break, reducing the catastrophic error caused by the confusion between these breaks.

There is a similar study on the importance of different bands in CSST photo-z estimation by Y. Cao et al. (2018); their Table 3 shows the impact of removing one of the CSST bands on the photo-z estimation using the SED fitting method. They find that r, g, and i are the most important ones as they lead to the most sizable increase in error when either of them is

Table 1
Photo-z Quality Metrics, η and $\sigma_{\rm NMAD}$, from the cNSF Algorithm on the DESCaLS Data

	$z < 21$					$z > 21$				
	P	P + Perr	P + M	P + Perr + M	Zhou + 2021	P	P + Perr	P + M	P + Perr + M	Zhou + 2021
η	2.17%	1.81%	1.45%	1.41%	1.51%	17.85%	16.65%	17.04%	16.29%	24.6%
$\sigma_{\rm NMAD}$	0.0162	0.0151	0.0140	0.0139	0.0133	0.0538	0.0524	0.0527	0.0517	0.0725

Note. The dataset is split into $z < 21$ and $z > 21$ based on the z-band magnitude. The results obtained using various combinations of the flux (P), flux error (Perr), and morphological (M) information are compared. We have also included the random forest results from R. Zhou et al. (2021), which make use of both photometric and morphological information.

missing, while missing *NUV* causes only a modest increase in the metrics. There are a couple of notable differences between their study and ours. First, they use an old CSST mock, which is less realistic than the present one. Second, their conclusion is based on SED fitting with `LePhare` only.

To investigate the cause of the discrepancy, we first perform SED fitting on this old catalog using `LePhare`. While Y. Cao et al. (2018) refine the COSMOS SED templates by performing SED interpolation, here, for simplicity, we directly use the default COSMOS SED templates. Although our resultant η and $\sigma_{\rm NMAD}$ are larger than the values shown in Y. Cao et al. (2018) by a factor of a few, we also find a similar pattern that, in the case of missing g, r, or i band data, the photo-z estimation suffers the largest reduction in accuracy. We go on to apply cNSF on this dataset, and we find that the photo-z precision metrics are smaller than theirs by an order of magnitude or so. The excellent performance of our algorithm is partly due to the fact that the mock is relatively simple. For cNSF, we also find similar results that missing *NUV* causes the most significant reduction in the photo-z estimation accuracy, and missing g, r, or i band data contribute to a mild decrease in accuracy only.

Our results suggest that the importance of the band depends on the level of accuracy in question. We can understand this in terms of the concepts of coarse and fine features in the data. For the low level of accuracy attained by SED fitting, the coarse features are mostly efficiently captured by g, r, or i. However, for the high-level algorithms such as cNSF, even missing one of the bands, the coarse features can still be captured well, and the differentiating features are the fine ones, which are the *NUV* in this case.

SED fitting is often used to guide survey design for its simplicity, e.g., in the case of CSST, it is used to find out the optimal photometric band configuration. Our results suggest that this may only lead to a partial result, and the conclusion so drawn may not be optimal.

3.1.5. Unrepresentative Training Sample and Its Mitigation

So far, we have considered the ideal scenario that the training dataset is a faithful representation of the final testing dataset. In reality, we may face the situation where the training data are not representative of the actual dataset to which it is applied. In particular, the training sample is likely to be underrepresentative at the faint end of the sample.

To investigate the issue of an unrepresentative training sample, we consider a toy sample that is complete up to an i-band magnitude of 22, beyond which only a fraction of galaxies of the full sample are available. We choose the fraction to be $\exp[-(i-22)/\sigma]$ with $\sigma = 0.5$.

In Table 2, we compare the results obtained with the fiducial full sample with the underrepresentative case. In the latter case, the training is carried out on the unrepresentative sample, while testing is performed on the full sample. We find that for both algorithms, the outlier fraction η is substantially inflated, and $\sigma_{\rm NMAD}$ also increases, albeit less significantly.

The accuracy deteriorates because the underrepresentative part constitutes a reduced weighting in the estimation of the full conditional probability distribution. To alleviate the situation, we consider boosting the weight of the underrepresentative part by resampling it. We do so by resampling the underrepresentative part with replacement until the distribution of galaxies is the same as the full sample.

Here, we randomly resample the underrepresentative part. If the underrepresentative sample is biased in color–magnitude space, we can obtain color-dependent weighting using methods like kNN to get a representative sample (M. Lima et al. 2008; I. Sadeh et al. 2016). In case of a very low galaxy sample in the faint end, there is a possibility of generating the faint sample using the galaxy formation model to supplement (e.g., N. Ramachandra et al. 2022; I. Moskowitz et al. 2024).

Table 2 shows that this simple upweighting indeed improves the overall results, and it is particularly significant for η. This again indicates that η is sensitive to the sample size of the training data. However, we note that there is an exceptional case where $\sigma_{\rm NMAD}$ actually increases slightly when resampling is applied in cNSF. This is not easy to understand, given that η does decrease. We note that even in the underrepresentative case, cNSF already gives a tight $\sigma_{\rm NMAD}$, so any error introduced may degrade the results. It is conceivable that the resampled galaxies are less representative (noisy), and thus can compromise all galaxies when they are resampled. Anyway, even in this case, the rise in the accuracy in η is quite likely to compensate for the drop in $\sigma_{\rm NMAD}$, so it is worthwhile to perform resampling.

3.2. Test on the COSMOS2020 Catalog

COSMOS2020 catalog (J. R. Weaver et al. 2022) is the latest galaxy catalog compilation in the 2 deg^2 COSMOS field. This field has been observed by numerous surveys across wavelengths in the electromagnetic spectrum. Many of these measurements in the UV, optical, and IR regimes have been subsumed in the COSMOS2020 catalog.

This catalog provides photometry measured using two different pipelines, CLASSIC and FARMER, based on flux measurements in the bands $izJYHK_{\rm s}$. The CLASSIC photometry is processed with the traditional pipeline, which has been used in previous releases of the COSMOS catalog (P. Capak et al. 2007; O. Ilbert et al. 2009; C. Laigle et al. 2016). In this pipeline, the image is first homogenized to a

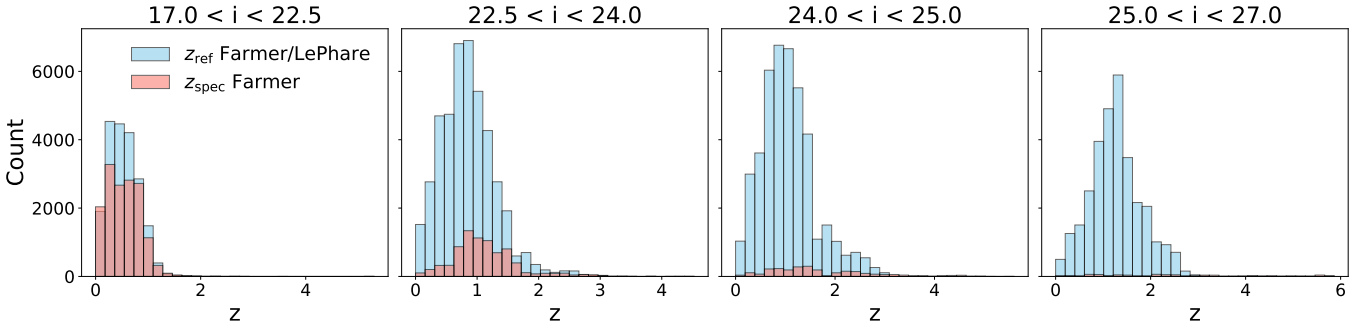

Figure 9. Redshift distribution histogram of the COSMOS2020 galaxy sample (blue) and the spec-z catalog (red) in the COSMOS field. The samples are divided into four i-magnitude bins. The reference redshifts for these samples are the high-quality COSMOS2020 photo-z and spec-z, respectively.

Table 2
Impact of Underrepresentative Training Sample and Its Mitigation by Resampling, Relative to Fiducial Full Sample

	cINN			cNSF		
	Full	Underrepresentative	Resampled	Full	Underrepresentative	Resampled
η	1.60 ± 0.16%	8.64% ± 0.83%	3.37% ± 0.018%	0.83% ± 0.07%	7.01% ± 0.62%	3.58% ± 0.43%
$\sigma_{\rm NMAD}$	0.018 ± 0.0004	0.027 ± 0.0015	0.022 ± 0.0004	0.017 ± 0.0002	0.022 ± 0.0004	0.023 ± 0.0005
b	−0.0001 ± 0.0004	−0.0024 ± 0.0028	−0.0004 ± 0.0005	0.0003 ± 0.001	−0.0012 ± 0.0014	.0002 ± 0.0006

Note. The resampling process increases the weight of the underrepresentative part. Both cINN and cNSF results are shown in the form of mean and standard deviation estimated over ten runs.

target point-spread function, and fluxes are then extracted within circular apertures. On the other hand, the FARMER photometry is measured with the profile-fitting code, `TRACTOR` (D. Lang et al. 2016), which derives a parametric model from the images assuming some morphological models. The FARMER photometry is shown to be more robust for faint galaxies.

The redshifts of galaxies in the catalog are given by high-quality photo-zs, derived from two template fitting codes, `LePhare` (S. Arnouts et al. 2002; O. Ilbert et al. 2006) and `EAZY` (G. B. Brammer et al. 2008), respectively. The broad, medium, and narrow-band data in the UV, optical, and IR frequency range are employed, totaling about 30 bands. The precise number of galaxies in the catalog depends on the photometry and photo-z estimator used. For example, for FARMER and `LePhare`, there are 137,892 galaxies, and for CLASSIC and `EAZY`, there are 240,746 galaxies. Besides the high-quality photo-z catalogs, a spec-z catalog in the COSMOS field is also available (A. A. Khostovan et al. 2025). In this catalog, there are 26,379 galaxies for FARMER photometry. As an illustration, in Figure 9, we show the reshift histograms for the FARMER-`LePhare` sample and the FARMER-spec-z catalog.

We illustrate the performance of both cINN and cNSF via a scatter plot in Figure 10. To be in line with the CSST mock setup, we only consider the photometric data from the CSST-like photometric bands, i.e., *NUVugrizy*. The reference redshift is the high-quality photo-z from COSMOS2020. In addition, we also display the results obtained using `LePhare` on the same seven CSST-like bands. In the `LePhare` fit, in addition to the 31 COSMOS SEDs offered alongside `LePhare`, we also use 12 SEDs from G. Bruzual & S. Charlot (2003). Following the extinction treatment in J. R. Weaver et al. (2022), we utilize the four extinction laws considered in J. R. Weaver et al. (2022) with the extinction amplitude as a free parameter.

Figure 10 demonstrates that the nflow-z results from cINN and cNSF are similar, and they are much better than the `LePhare` fit. In terms of η and $\sigma_{\rm NMAD}$, these metrics from the nflow-z algorithms are generally lower by a factor of a few relative to the `LePhare` results. As expected, the metrics get worse for fainter galaxies because of poor photometric measurements and a reduction in training data size.

In Table 3, we show the photo-z metrics b, $\sigma_{\rm NMAD}$, and η for different galaxy samples built from the COSMOS catalogs. Different combinations of the galaxy photometry (FARMER or CLASSIC) and reference redshift (high-quality `LePhare` or `EAZY` photo-z from COSMOS2020 or spec-z) are considered. Again, we consider four i-band magnitude groups. For each galaxy sample, the trends are similar to those found in Figure 10. That is, cINN and cNSF give comparable results, while the `LePhare` fit is worse by a factor of a few. In greater detail, cNSF tends to yield more accurate photo-z estimations across different samples. This is apparent for the FARMER-spec-z sample, especially in the faint groups ($24 < i < 25$ and $25 < i < 27$), where the available training sample is scarce. Thus, we conclude that cNSF is more robust than cINN.

We note that Table 3 also reveals that the resultant photo-z accuracy is dependent on the photometry and photo-z estimators. However, this trend is largely driven by the fact that there are varying numbers of galaxies in different galaxy samples. Indeed, we generally find that the overall photo-z metrics are worse for a sample with a larger number of galaxies. This is particularly obvious for the faint groups. It is conceivable that the extra galaxies are the culprit for the poor performance. This also means that some of these algorithms, such as FARMER and `LePhare`, are better tuned to get rid of the low-quality galaxies. We can verify this using the same sample of galaxies. For the spec-z sample, there are unique IDs allowing us to identify galaxies in the FARMER and CLASSIC photometry. For the same spec-z subsample, we find that the variations in the photo-z metrics are much reduced

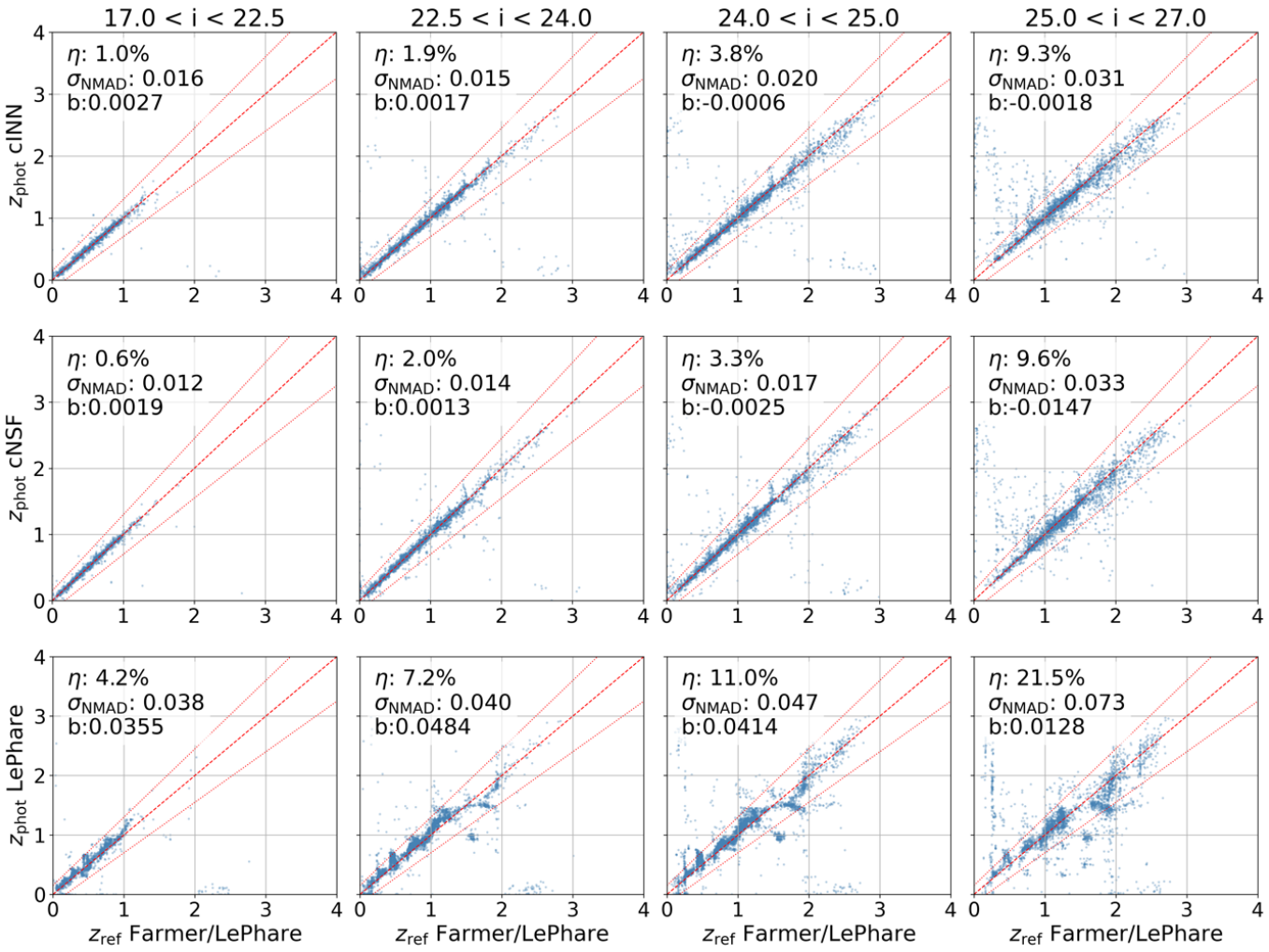

Figure 10. Scatter plot between the photo-z estimated using the seven CSST-like band data vs. the fiducial reference redshift from COSMOS2020. The results from cINN, cNSF, and `LePhare` are compared (top to bottom). The samples are divided into four i-magnitude groups (left to right).

for the choice of FARMER or CLASSIC photometry. We stress that our goal here is to compare the performance of different photo-z estimation algorithms; as long as we concentrate on the same sample, this sample dependence does not affect our conclusions.

Finally, we comment on the missing photometry data issue and its mitigation strategies. When the reference redshift is photo-z, there are relatively abundant galaxies available. Thus, we can afford to select only those galaxies with data in all bands intact. However, for the spec-z case, the available spec-z galaxies are scarce, especially in the faint end. We have to consider galaxies with some flux data in certain bands unavailable. However, for the MLM, the data in all channels must be used, and so the missing data should be somehow filled in. We test a few different methods to impute the missing data, including replacing the missing data with the mean, median, minimum of the data array, or assigning an arbitrary value such as -99.9. Besides these commonly used simple methods, we have also considered the imputation algorithm GAIN (J. Yoon et al. 2018), which is based on the GAN algorithm (see Section 3.3). Z. Luo et al. (2024b) demonstrate that GAIN is effective for photometric data imputation. Among these methods, GAIN stands out in our tests, giving the lowest η and $\sigma_{\rm NMAD}$ in all the cases considered, consistent with Z. Luo et al. (2024b). Therefore, for the spec-z case in Table 3, we have employed the GAIN to fill in the missing data.

3.3. Comparison with cGAN and MDN

GAN (I. J. Goodfellow et al. 2014) is another well-known generative modeling network in deep learning. In GAN, rather than training the network with the maximum likelihood, a novel min–max game framework is introduced, where two adversarial models compete against each other. An adversarial model, called generative model G, learns the underlying data distribution, while another adversarial model, dubbed discriminative model D, evaluates the probability that a sample originates from the training data rather than from G. Both G and D can be implemented as nonlinear mapping functions, such as MLPs. M. Mirza & S. Osindero (2014) extend GAN to a conditional generalization, cGAN, by feeding the extra conditional information to both adversarial models G and D. Thus, the treatment of the condition is similar to our nflow-z models. M. Garcia-Fernandez (2025) applied the cGAN model to photo-z estimation. As both NF and GAN are important

Table 3
Comparison of b, $\sigma_{\rm NMAD}$, and η Results Obtained from cINN, cNSF, and LePhare for Different Galaxy Samples

	$17 < i < 22.5$			$22.5 < i < 24$			$24 < i < 25$			$25 < i < 27$			
	b	$\sigma_{\rm NMAD}$	η	b	$\sigma_{\rm NMAD}$	η	b	$\sigma_{\rm NMAD}$	η	b	$\sigma_{\rm NMAD}$	η	galaxy sample
cINN	**0.001**	0.015	1.0%	**0.000**	0.015	2.0%	**0.000**	0.020	4.0%	**−0.005**	0.035	9.7%	FARMER
cNSF	0.002	**0.012**	**0.6%**	0.001	**0.014**	**1.9%**	−0.002	**0.017**	**3.3%**	−0.015	**0.033**	**9.6%**	LePhare
LePhare	0.036	0.038	4.2%	0.048	0.040	7.2%	0.041	0.047	11.0%	0.013	0.073	21.5%	(137,892)
cINN	**0.000**	**0.013**	**1.7%**	**−0.001**	0.016	**2.7%**	**0.000**	**0.022**	8.0%	−0.004	0.051	21.2%	FARMER
cNSF	**0.000**	**0.013**	1.9%	**−0.001**	**0.014**	2.9%	0.004	0.023	**6.6%**	**0.001**	**0.043**	**20.3%**	EAZY
LePhare	0.032	0.042	6.7%	0.049	0.041	7.8%	0.041	0.054	13.7%	−0.006	0.096	29.3%	(161,297)
cINN	0.001	0.012	1.8%	−0.004	0.019	3.9%	**0.001**	0.027	**7.1%**	−0.006	0.051	**13.0%**	CLASSIC
cNSF	**0.000**	**0.011**	**0.8%**	**0.000**	**0.017**	**3.7%**	0.002	**0.025**	7.6%	**−0.002**	**0.045**	14.7%	LePhare
LePhare	0.026	0.043	4.5%	0.047	0.057	9.1%	0.038	0.063	14.3%	0.038	0.092	26.1%	(145,779)
cINN	**0.000**	0.015	3.3%	**0.000**	0.022	8.7%	**0.000**	0.038	**17.0%**	−0.008	0.096	**32.7%**	CLASSIC
cNSF	-0.002	**0.013**	**2.8%**	**0.000**	**0.020**	**7.2%**	0.001	**0.037**	17.9%	0.006	**0.092**	33.1%	EAZY
LePhare	0.031	0.049	9.8%	0.060	0.056	13.9%	0.053	0.078	24.2%	**0.004**	0.153	40.7%	(240,746)
cINN	0.001	0.014	3.1%	**0.001**	0.017	3.5%	0.021	0.062	15.8%	**0.000**	0.158	40.7%	FARMAER
cNSF	**0.000**	**0.012**	**2.4%**	**0.001**	**0.012**	**2.5%**	**−0.001**	**0.027**	**6.0%**	−0.005	**0.032**	**28.4%**	spec-z
LePhare	0.029	0.044	12.1%	0.057	0.046	13.0%	0.075	0.060	20.6%	0.160	0.151	42.9%	(26,379)

Note. In the first four big rows, the reference redshifts are high-quality photo-zs (LePhare or EAZY) based on the photometry from FARMER or CLASSIC, while the last big row shows the results for the FARMER sample with spec-z. The results for four i-magnitude bins are shown (from left to right). The last column shows the galaxy sample used, with the number of galaxies shown in brackets. To guide the eyes, the best metric in each subcolumn is in bold. Sample built from the COSMOS2020 catalog.

members in generative modeling, it is interesting to check how nflow-z fairs relative to cGAN.

In addition, M. Garcia-Fernandez (2025) also compares their results against those from the mixture density network (MDN; C. Bishop 1994). In MDN, a sum of base distributions is used to parameterize the target probability distribution, and the weights of each distribution and the parameters for each base distribution are optimized by neural networks. The base distribution is commonly taken to be the normal distribution, and in this case, the base distribution parameters are the mean and variance of the normal distribution. M. Garcia-Fernandez (2025) used the MDN implementation ported from Z. Ansari et al. (2021).

Here, we compare our results against the cGAN and MDN results from M. Garcia-Fernandez (2025). To do so, we apply our method to the same dataset provided by M. Garcia-Fernandez (2025), which can be downloaded alongside the cGAN code. The dataset is a subset of the DES Y1 galaxies (A. Drlica-Wagner et al. 2018). Only galaxies in the Stripe 82 region with the associated spec-z from SDSS falling in $0 < z_{\rm spec} < 0.8$ are selected. The redshift distribution of this galaxy sample is shown in Figure 11. This dataset contains griz bands magnitude only, and in particular, no error bars for the magnitudes are provided. Thus, in our NF runs, we only use the fluxes and colors computed from these magnitudes.

In Figure 12, we plot the bias, η, and $\sigma_{\rm NMAD}$ from different algorithms. The cGAN and MDN results are extracted from M. Garcia-Fernandez (2025) directly. Note that, following M. Garcia-Fernandez (2025), the bias is defined as the mean of Δz rather than its median. cINN and cNSF perform similarly well and are better than cGAN and MDN when all the metrics are taken into account. The cGAN fares the worst in all the metrics. Although MDN yields competitive bias and η, its $\sigma_{\rm NMAD}$ is substantially higher than the NF results.

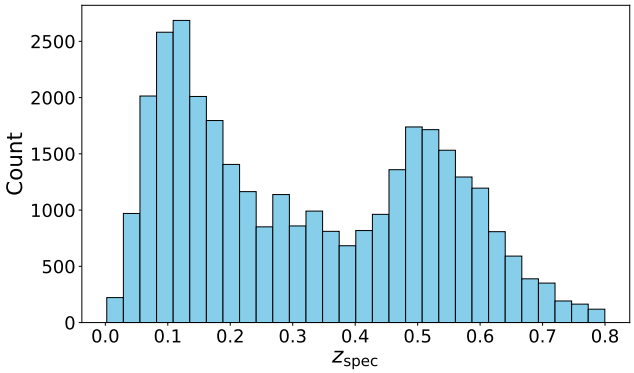

Figure 11. Redshift distribution of the galaxy sample used in the cGAN and MDN comparison test.

While both NF and GAN are members of generative modeling in deep learning, NF directly models the probability distribution of the target space, and GAN does not. Our studies suggest that direct probability distribution modeling is instrumental in enhancing the precision of photo-z estimation. MDN is similar to NF in spirit, as it uses a mixture of Gaussian distributions to model the target distribution. However, it is less flexible than NF because it still has a fixed function form, while NF allows for more degrees of freedom in constructing the coordinate transformation.

3.4. Comparison with ANNz2

ANNz2 (I. Sadeh et al. 2016) is a widely used photo-z estimation code, and it improves upon the previous version ANNz (A. A. Collister & O. Lahav 2004). This implementation employs multiple MLMs from the TMVA package (A. Hocker et al. 2007). Among the available methods,

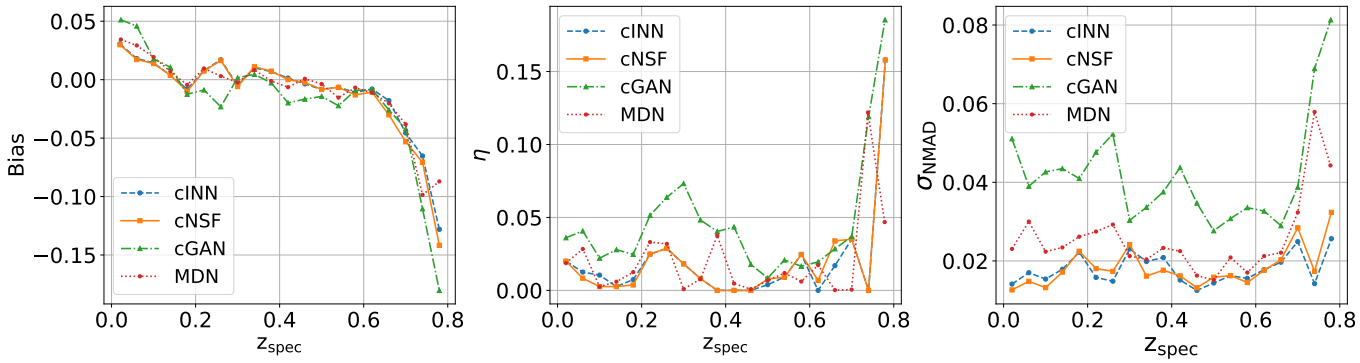

Figure 12. Bias, η, and $\sigma_{\rm NMAD}$ (from left to right) results for a DES Y1 sample shown in Figure 11. The cGAN (green, dotted-dashed) and MDN (red, dotted) results are extracted from M. Garcia-Fernandez (2025). The NF algorithms (cINN (blue, dashed) and cNSF (orange, solid)) perform similarly well, and their overall results are better than cGAN or MDN.

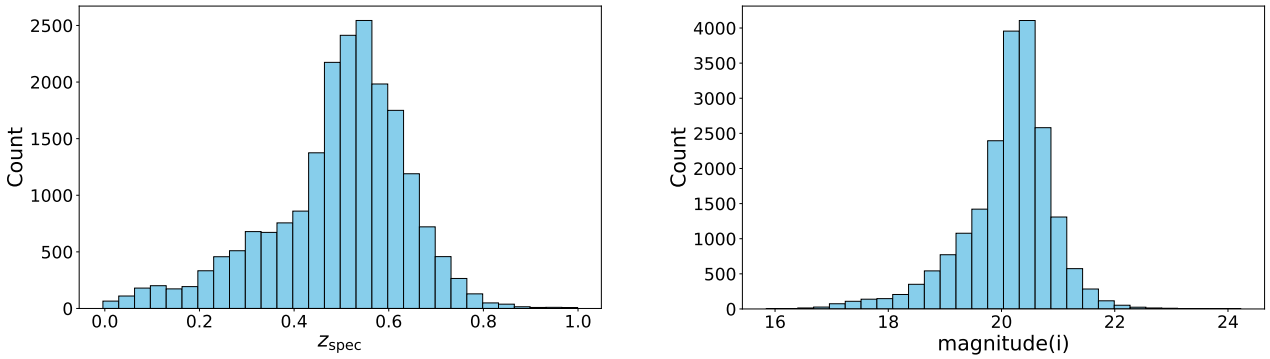

Figure 13. Redshift (left) and i-band magnitude (right) distribution for the galaxy sample used for ANNz2 comparison.

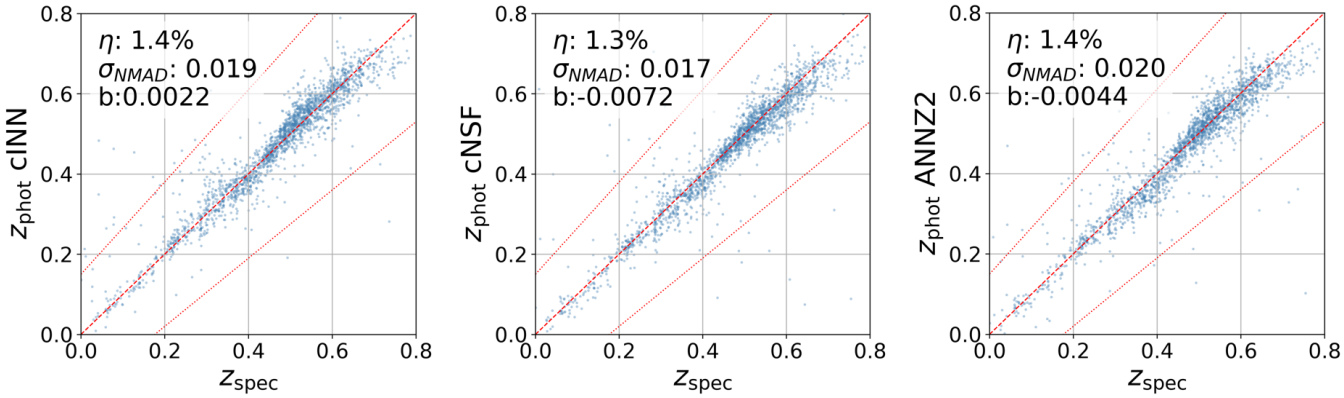

Figure 14. Scatter plot between the spec-z and photo-z estimated by cINN, cNSF, and ANNz2 (left to right). While the performance of these methods is similar, cINN and cNSF take significantly less time to run.

artificial neural networks (ANNs) and boosted decision trees (BDTs) are found to be the most useful in photo-z estimation.

There are two principal modes of operation in ANNz2: single regression and randomized regression. The former outputs a single photo-z estimate, while the latter is capable of generating PDFs. We focus on the Randomized Regression here. An ensemble of 100 MLMs is trained, which can be obtained by setting the initialization random seed or the parameters in the configuration. For instance, in ANNs, we can set the number of hidden layers and the number of neurons thereof, while the number of trees and the type of forest construction algorithm can be varied in BDTs. Both ANN and BDT are used in our runs. The performance of the MLMs is ranked based on metrics such as the bias and spread of the estimation, and the best-performing MLM is chosen to produce the photo-z prediction. Random weights are assigned to the MLMs, and the combined PDF yielding the best PIT test is adopted for the final PDF.

We conduct the tests on the dataset offered alongside ANNz2. This dataset consists of the ugriz band magnitudes and their associated errors from the SDSS DR10 dataset (C. P. Ahn et al. 2014). The redshift and magnitude distribution of this sample are shown in Figure 13. Figure 14 is a scatter plot between the spec-z and photo-z estimates from

cINN, cNSF, and ANNz2. The results from these methods are not significantly different, with cNSF giving the smallest η and $\sigma_{\rm NMAD}$, and cINN the smallest bias.

Because ANNz2 requires running many MLMs, it is much more time-consuming. In our test, the ANNz2 run takes nearly 1.5 days, while cINN and cNSF take about 15 and 3 minutes, respectively.

3.5. Application to DESCaLS Data

In this section, we apply our algorithm to the photometric data from the Dark Energy Camera Legacy Survey (DESCaLS), which is part of the DESI Legacy Imaging Surveys (A. Dey et al. 2019). Besides its original purpose for DESI target selection, the DESCaLS catalog has been used for numerous cosmological studies, e.g., R. Song et al. (2024), J. Qin et al. (2025).

As a value-added product, Photometric Redshifts for the Legacy Surveys (PRLS) catalog (R. Zhou et al. 2021) offers photo-z estimates for the Legacy Surveys sample. The photo-zs are derived from the random forest algorithm, based on the DR8 dataset of the Legacy Surveys. In addition to the photometric data in g, r, and z bands from DESCaLS, the near-infrared imaging data in $W1$ and $W2$ bands from Wide-field Infrared Survey Explorer (E. L. Wright et al. 2010) are also used to assist photo-z estimation. In the random forest analysis of R. Zhou et al. (2021), their input takes the r-band magnitude, $g-r$, $r-z$, $z-W1$, and $W1-W2$ colors, and additional morphological information by means of the half-light radius, axis ratio, and shape probability. The usage of the shape information was inspired by the study of J. Y. H. Soo et al. (2018), which finds that the morphological information can add substantial gain in photo-z accuracy if there are few bands available, such as grz only.

In Table 1, we present the photo-z estimate results on this DESCaLS catalog. The results are derived from the cNSF algorithm. We have isolated the impact of flux errors by utilizing flux information alone and both flux and flux error. Motivated by R. Zhou et al. (2021), we also consider the case when both photometry and morphological information are employed. We use the three types of morphological information as R. Zhou et al. (2021). In order to directly compare with R. Zhou et al. (2021), we divide the samples into two groups based on z-band magnitude: $z<21$ and $z>21$. Note that for these samples, we impute the missing entries using the GAIN algorithm.

For the bright group ($z<21$), while adding the photometry error gives some gain on the photo-z accuracy (17% on η and 8% on $\sigma_{\rm NMAD}$), including the morphological information significantly improves the photo-z accuracy (33% on η and 14% on $\sigma_{\rm NMAD}$). Further adding the flux error on top of the flux plus morphology case only marginally improves the results. For the faint group ($z>21$), adding the flux error or morphological information only mildly improves the accuracy, with flux error contributing a slightly larger gain. By including both, the accuracy is enhanced by (14% on η and 9% on $\sigma_{\rm NMAD}$). The finding that the morphological information is only significant for the bright group is consistent with R. Zhou et al. (2021).

Table 1 also shows the PRLS results, which are extracted from Appendix B in R. Zhou et al. (2021). The PRLS results are derived from both photometric and morphological information. For the bright group ($z<21$), our results are similar to PRLS. However, for the faint group ($z>21$), our algorithm gives substantial improvement over the PRLS results, with η reduced by 34% and $\sigma_{\rm NMAD}$ by 29%.

We have demonstrated that our algorithm can bring substantial improvement in the photo-z estimates in the faint end. In turn, this is expected to improve the cosmological application results. This exercise demonstrates that our algorithm is particularly advantageous for the faint galaxy samples, where the training samples are scarce and the measurements are noisy.

4. Conclusions

Many large-scale cosmological surveys are ongoing or upcoming, e.g., those mentioned in the Introduction. These surveys are expected to bring about an enormous amount of photometric data. In particular, the observed sample is expected to extend to high redshift and much deeper magnitude. Getting an accurate photo-z estimate for the photometric samples is a prerequisite to achieving exquisite scientific results from these data samples. Although many works have been devoted to photo-z estimation, in particular, various machine learning algorithms have been applied for photo-z estimation, there is still room for improvement, especially in the regime where the spec-z training data are sparse.

In this work, we apply NF, a powerful machine learning method, to develop a new photo-z estimation pipeline. We call the NF-based photo-z estimation framework, nflow-z. nflow-z directly models the redshift probability distribution, conditional on the observables such as fluxes and colors. It learns the coordinate transformation required to transform the complex target distribution to a simple base distribution, such as normal. We explore two types of architectures to implement nflow-z: cINN and cNSF, and contrast the performance of these implementations.

We apply nflow-z to several datasets and compare the results against other state-of-the-art algorithms. The datasets tested include a CSST mock, COSMOS2020 catalog, a DES Y1 sample, an SDSS sample, and a DESCaLS catalog. These datasets cover diverse redshift ranges, magnitude spans, and various training sample sizes, and thus, they can help to check the performance of the NF methods under different situations. Among others, we employ the normalized median absolute deviation $\sigma_{\rm NMAD}$, the outlier fraction η, and the bias b to quantify the photo-z accuracy. Our testing results generally indicate that the NF methods compare favorably to other algorithms. In the CSST mock test, the cNSF algorithm yields the lowest η and $\sigma_{\rm NMAD}$, outperforming others, including random forest, MLP, CNN, hybrid transfer network, and recurrent neural network (the list in Section 3.1.2). Moreover, the NF network is generally much more lightweight than many of these networks, and so it requires substantially fewer resources to run. For the tests on COSMOS2020 catalogs, we find that our results generally improve over the SED fit by `LePhare` by a factor of a few (Figure 10 and Table 3). In the DESCaLS dataset test, while for the bright sample ($z<21$), our results are similar to the official PRLS results from random forest, ours improve over the official ones by about 30% for the faint sample ($z>21$) in terms of η or $\sigma_{\rm NMAD}$ (Table 1). We have compared nflow-z against the results from cGAN and MDN on a DES Y1 sample. We find that nflow-z surpasses cGAN (in terms of η or $\sigma_{\rm NMAD}$) and beats MDN (in terms of

$\sigma_{\rm NMAD}$; Figure 12). Even though our test results are similar to the ANNz2 in performance on an SDSS sample, the NF method takes much less time to run (ten minutes versus a couple of days). Thus, we have demonstrated that the NF-based method is effective in estimating photo-z on a number of datasets. Some of these datasets have been applied to various cosmological and astrophysical applications; the improvement in photo-z precision brought by the NF method should be beneficial to these studies.

Here, we summarize a few key properties of nflow-z. It models the redshift probability distribution directly and naturally yields a PDF for the photo-z estimate. The redshift-color degeneracy is a physical effect limiting the accuracy of the photo-z estimation. nflow-z reduces the impact of this degeneracy by resolving the degeneracy into a higher-dimensional surface in latent space, thanks to the assignment of a latent variable to each training sample in the NF network structure. Moreover, the nflow-z network is relatively simple. It is a 1D network for redshift, and this enables the network to concentrate on the redshift information in the data. In practice, the simple network structure means that it takes little training and running time, so it is much less resource demanding than other competing algorithms and is easy to tune its parameters to optimize its performance. For example, in the CSST mock test, running the training and testing steps takes less than half an hour.

Our tests suggest that cNSF tends to deliver smaller η and $\sigma_{\rm NMAD}$ than cINN, although it seems that cNSF often gives a larger bias (this can be corrected for). Also, we find that cNSF is more robust in the sense that it still manages to deliver good results in the case of poor photometry measurements or a sparse training sample. Consequently, we recommend cNSF instead of cINN. The advantages of nflow-z are especially apparent in the faint end regime where the training sample is relatively sparse, and the data measurements are noisy. Thus, it can be particularly useful for deep surveys or noisy datasets. The codes implementing nflow-z are available for download on GitHub[8] or the frozen version on Zenodo.[9] They can be readily applied to other photometric datasets. We encourage readers to explore and experiment with them.

Acknowledgments

We thank Jonás Chaves-Montero, Dezi Liu, Jian Qin, and Ji Yao for useful discussions. Y.R., K.C.C., and R.S. are supported by the National Science Foundation of China under the grant Nos. 12273121 and 12533002. Y.L. is supported by the Major Key Project of Peng Cheng Laboratory and the National Key Research and Development Program of China under grant No. 2023YFA1605600. This work is also supported by the science research grants from the China Manned Space Project with No. CMS-CSST-2021-B01.

ORCID iDs

Kwan Chuen Chan https://orcid.org/0000-0001-8757-408X

References

Abbott, T., Aguena, M., Alarcon, A., et al. 2022, PhRvD, 105, 023520
Abbott, T., et al. 2019, MNRAS, 483, 4866
Abbott, T. M. C., Abdalla, F. B., Alarcon, A., et al. 2018, PhRvD, 98, 043526
Abbott, T. M. C., Adamow, M., Aguena, M., et al. 2024, PhRvD, 110, 063515
Abbott, T. M. C., et al. 2022, PhRvD, 105, 043512
Ahn, C. P., Alexandroff, R., Allende Prieto, C., et al. 2014, ApJS, 211, 17
Aihara, H., AlSayyad, Y., Ando, M., et al. 2022, PASJ, 74, 247
Almosallam, I. A., Jarvis, M. J., & Roberts, S. J. 2016, MNRAS, 462, 726
Amon, A., Gruen, D., Troxel, M. A., et al. 2022, PhRvD, 105, 023514
Ansari, Z., Agnello, A., & Gall, C. 2021, A&A, 650, A90
Ardizzone, L., Bungert, T., Draxler, F., et al. 2018-2022, Framework for Easily Invertible Architectures (FrEIA)
Ardizzone, L., Kruse, J., Wirkert, S., et al. 2018, arXiv:1808.04730
Ardizzone, L., Lüth, C., Kruse, J., Rother, C., & Köthe, U. 2019, arXiv:1907.02392
Arnouts, S., Cristiani, S., Moscardini, L., et al. 1999, MNRAS, 310, 540
Arnouts, S., Moscardini, L., Vanzella, E., et al. 2002, MNRAS, 329, 355
Asgari, M., Lin, C.-A., Joachimi, B., et al. 2021, A&A, 645, A104
Bartelmann, M., & Schneider, P. 2001, PhR, 340, 291
Benítez, N. 2000, ApJ, 536, 571
Bingham, E., Chen, J. P., Jankowiak, M., et al. 2019, JMLR, 20, 1
Bishop, C. 1994, Technical Report, Aston University
Bister, T., Erdmann, M., Köthe, U., & Schulte, J. 2022, EPJC, 82, 171
Bolzonella, M., Miralles, J. M., & Pelló, R. 2000, A&A, 363, 476
Bordoloi, R., Lilly, S. J., & Amara, A. 2010, MNRAS, 406, 881
Brammer, G. B., van Dokkum, P. G., & Coppi, P. 2008, ApJ, 686, 1503
Bruzual, G., & Charlot, S. 2003, MNRAS, 344, 1000
Buchs, R., Davis, C., Gruen, D., et al. 2019, MNRAS, 489, 820
Cao, Y., Gong, Y., Meng, X.-M., et al. 2018, MNRAS, 480, 2178
Capak, P., Aussel, H., Ajiki, M., et al. 2007, ApJS, 172, 99
Carliles, S., Budavári, T., Heinis, S., Priebe, C., & Szalay, A. S. 2010, ApJ, 712, 511
Carrasco Kind, M., & Brunner, R. J. 2013, MNRAS, 432, 1483
Carrasco Kind, M., & Brunner, R. J. 2014, MNRAS, 438, 3409
Chan, K. C., Avila, S., Carnero Rosell, A., et al. 2022, PhRvD, 106, 123502
Chan, K. C., & Ren, Y. 2025, Photo-z Estimation with Normalizing Flow
Collister, A. A., & Lahav, O. 2004, PASP, 116, 345
Crenshaw, J. F., Kalmbach, J. B., Gagliano, A., et al. 2024, AJ, 168, 80
CSST Collaboration, Gong, Y., Miao, H., et al. 2025, arXiv:2507.04618
Dalal, R., Li, X., Nicola, A., et al. 2023, PhRvD, 108, 123519
De Vicente, J., Sánchez, E., & Sevilla-Noarbe, I. 2016, MNRAS, 459, 3078
Dey, A., Schlegel, D. J., Lang, D., et al. 2019, AJ, 157, 168
Dinh, L., Sohl-Dickstein, J., & Bengio, S. 2016, arXiv:1605.08803
D'Isanto, A., & Polsterer, K. L. 2018, A&A, 609, A111
Dolatabadi, H. M., Erfani, S., & Leckie, C. 2020, in Int. Conf. on Artificial Intelligence and Statistics (PMLR), 4236
Drlica-Wagner, A., Sevilla-Noarbe, I., Rykoff, E. S., et al. 2018, ApJS, 235, 33
Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. 2019, in Advances in Neural Information Proc. Systems (Curran Associates, Inc.), 7511
Eifler, T., Miyatake, H., Krause, E., et al. 2021, MNRAS, 507, 1746
Garcia-Fernandez, M. 2025, arXiv:2501.06532
Gerdes, D. W., Sypniewski, A. J., McKay, T. A., et al. 2010, ApJ, 715, 823
Gong, Y., Liu, X., Cao, Y., et al. 2019, ApJ, 883, 203
Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., et al. 2014, in Advances in Neural Information Proc. Systems (Curran Associates Inc.)
Haldemann, J., Ksoll, V., Walter, D., et al. 2023, A&A, 672, A180
Heymans, C., Grocutt, E., Heavens, A., et al. 2013, MNRAS, 432, 2433
Heymans, C., Tröster, T., Asgari, M., et al. 2021, A&A, 646, A140
Hikage, C., Oguri, M., Hamana, T., et al. 2019, PASJ, 71, 43
Hildebrandt, H., Viola, M., Heymans, C., et al. 2017, MNRAS, 465, 1454
Hocker, A., Speckmayer, P., Stelzer, J., et al. 2007
Hoyle, B. 2016, A&C, 16, 34
Ilbert, O., Arnouts, S., McCracken, H. J., et al. 2006, A&A, 457, 841
Ilbert, O., Capak, P., Salvato, M., et al. 2009, ApJ, 690, 1236
Ivezić, Ž., Kahn, S. M., Tyson, J. A., et al. 2019, ApJ, 873, 111
Kang, D. E., Pellegrini, E. W., Ardizzone, L., et al. 2022, MNRAS, 512, 617
Khostovan, A. A., Kartaltepe, J. S., Salvato, M., et al. 2025, arXiv:2503.00120
Kingma, D. P., & Ba, J. 2015, arXiv:1412.6980
Kingma, D. P., & Dhariwal, P. 2018, in Advances in Neural Information Proc. Systems, 31, (Curran Associates Inc)
Kobyzev, I., Prince, S. J., & Brubaker, M. A. 2020, ITPAM, 43, 3964
Ksoll, V. F., Ardizzone, L., Klessen, R., et al. 2020, MNRAS, 499, 5447
Laigle, C., McCracken, H. J., Ilbert, O., et al. 2016, ApJS, 224, 24
Lang, D., Hogg, D. W., & Mykytyn, D., 2016 The Tractor: Probabilistic astronomical source detection and measurement, Astrophysics Source Code Library, ascl:1604.008
Laureijs, R., Amiaux, J., Arduini, S., et al. 2011, arXiv:1110.3193

[8] https://github.com/kcc274/nflow-z.git
[9] DOI: 10.5281/zenodo.17595685 (K. C. Chan & Y. Ren 2025)

Li, R., Napolitano, N. R., Roy, N., et al. 2022, ApJ, 929, 152
Li, X., Zhang, T., Sugiyama, S., et al. 2023, PhRvD, 108, 123518
Lima, M., Cunha, C. E., Oyaizu, H., et al. 2008, MNRAS, 390, 118
Lu, J., Luo, Z., Chen, Z., et al. 2024, MNRAS, 527, 12140
Luo, Z., Li, Y., Lu, J., et al. 2024a, MNRAS, 535, 1844
Luo, Z., Tang, Z., Chen, Z., et al. 2024b, MNRAS, 531, 3539
Lupton, R. H., Gunn, J. E., & Szalay, A. S. 1999, AJ, 118, 1406
Mandelbaum, R., Eifler, T., Hložek, R., et al. 2018, arXiv:1809.01669
Masters, D., Capak, P., Stern, D., et al. 2015, ApJ, 813, 53
McQuinn, M., & White, M. 2013, MNRAS, 433, 2857
Ménard, B., Scranton, R., Schmidt, S., et al. 2013, arXiv:1303.4722
Mirza, M., & Osindero, S. 2014, arXiv:1411.1784
Miyatake, H., Sugiyama, S., Takada, M., et al. 2023, PhRvD, 108, 123517
Moskowitz, I., Gawiser, E., Crenshaw, J. F., et al. 2024, ApJL, 967, L6
Newman, J. A. 2008, ApJ, 684, 88
Newman, J. A., & Gruen, D. 2022, ARA&A, 60, 363
Padmanabhan, N., Schlegel, D. J., Seljak, U., et al. 2007, MNRAS, 378, 852
Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., & Lakshminarayanan, B. 2021, JMLR, 22, 1
Peng, H., Xu, H., Zhang, L., Chen, Z., & Yu, Y. 2022, MNRAS, 516, 6210
Phan, D., Pradhan, N., & Jankowiak, M. 2019, arXiv:1912.11554
Prince, S. J. 2023, Understanding Deep Learning (MIT Press)
Qin, J., Zhang, P., Chen, Z., et al. 2025, PhRvD, 112, 103514
Ramachandra, N., Chaves-Montero, J., Alarcon, A., et al. 2022, MNRAS, 515, 1927
Rezende, D., & Mohamed, S. 2015, in ICML'15: Proc. 32nd Int. Conf. Machine Learning, Vol. 37, (ACM), 1530
Sadeh, I., Abdalla, F. B., & Lahav, O. 2016, PASP, 128, 104502
Salvato, M., Ilbert, O., & Hoyle, B. 2019, NatAs, 3, 212
Schmidt, S. J., Ménard, B., Scranton, R., Morrison, C., & McBride, C. 2013, MNRAS, 431, 3307
Schneider, M., Knox, L., Zhan, H., & Connolly, A. 2006, ApJ, 651, 14
Schuldt, S., Suyu, S. H., Cañameras, R., et al. 2021, A&A, 651, A55
Secco, L. F., Samuroff, S., Krause, E., et al. 2022, PhRvD, 105, 023515
Seo, H.-J., Ho, S., White, M., et al. 2012, ApJ, 761, 13
Song, R., Chan, K. C., Xu, H., & Zheng, W. 2024, MNRAS, 530, 881
Soo, J. Y. H., Moraes, B., Joachimi, B., et al. 2018, MNRAS, 475, 3613
Spergel, D., Gehrels, N., Baltay, C., et al. 2015, arXiv:1503.03757
Sugiyama, S., Miyatake, H., More, S., et al. 2023, PhRvD, 108, 123521
Sun, Z., Speagle, J. S., Huang, S., Ting, Y.-S., & Cai, Z. 2023, arXiv:2310.20125
Tabak, E. G., & Turner, C. V. 2013, CPA, 66, 145
Troxel, M. A., MacCrann, N., Zuntz, J., et al. 2018, PhRvD, 98, 043528
van den Busch, J. L., Hildebrandt, H., Wright, A. H., et al. 2020, A&A, 642, A200
Weaver, J. R., Kauffmann, O. B., Ilbert, O., et al. 2022, ApJS, 258, 11
Wright, A. H., Hildebrandt, H., van den Busch, J. L., & Heymans, C. 2020, A&A, 637, A100
Wright, A. H., Stölzner, B., Asgari, M., et al. 2025, arXiv:2503.19441
Wright, E. L., Eisenhardt, P. R. M., Mainzer, A. K., et al. 2010, AJ, 140, 1868
Xu, H., Zhang, P., Peng, H., et al. 2023, MNRAS, 520, 161
Yoon, J., Jordon, J., & Schaar, M. 2018, ICML, 80, 5689
Zhan, H. 2011, SSPMA, 41, 1441
Zhang, H., Zuo, S., & Zhang, L. 2023, RAA, 23, 075011
Zhang, L., Yu, Y., & Zhang, P. 2017, ApJ, 848, 44
Zhang, P., Pen, U.-L., & Bernstein, G. 2010, MNRAS, 405, 359
Zheng, W., Chan, K. C., Xu, H., Zhang, L., & Song, R. 2024, A&A, 692, A186
Zhou, R., Newman, J. A., Mao, Y.-Y., et al. 2021, MNRAS, 501, 3309
Zhou, X., Gong, Y., Meng, X.-M., et al. 2021, ApJ, 909, 53
Zhou, X., Gong, Y., Meng, X.-M., et al. 2022a, MNRAS, 512, 4593
Zhou, X., Gong, Y., Meng, X.-M., et al. 2022b, RAA, 22, 115017