PhotoVAEz: Photometric Redshifts Using Semi-Supervised Variational Autoencoders

J. O. Hjortlund[1,⋆], K. Stensbo-Smidt[2,⋆⋆], and C. Gall[1⋆⋆⋆]

[1] DARK, Niels Bohr Institute, University of Copenhagen, Jagtvej 128, 2200 Copenhagen, Denmark

[2] DTU Compute

October 16, 2023

ABSTRACT

Context. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Aims. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Methods. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Results. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Conclusions. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

1. Introduction

1. Upcoming surveys like LSST will produce a ton of data
2. No time for spectra, gotta do photo-z's
3. Current methods: template-based and ML-based. Introduce neural network and machine learning acronyms.
4. Issues for both are degeneracies and amount of spec-z's available to fine-tune / train
5. Science goals have requirements not met by current methods (some have 2/3)
6. Semi-supervised ML methods can harness information from data without spec-z's
7. VAEs are unsupervised ML models used in representation learning which could be used to break degeneracies
8. Extended to the semi-supervised setting - use for photo-z's!
9. Present PhotoVAEz, a trained semi-supervised VAE for SDSS photo-z's as well as package for building new models
10. Paper outline

Variational autoencoders (VAEs), first introduced by Kingma & Welling (2022) and Rezende et al. (2014), belong to a family of unsupervised neural networks that are trained to copy its input to its output in a probabilistic manner. This probabilistic mapping leads to a generative model that can be used to produce new samples from the input distribution. During this process VAEs learn useful representations of the data in a lower dimensional space referred to as the *latent space*. VAEs have shown to be competitive with state-of-the-art generative models (Child, 2021; Vahdat & Kautz, 2021; Maaløe et al., 2019). VAEs have also seen large interest in the field representation learning, where the goal is to learn disentangled representations of complex data in an unsupervised manner. In this setting the latent space learnt by VAEs has shown to be extremely competitive, being able to produce disentangled representations of complex inputs in a hierarchical manner Siddharth et al. (2017).

Although VAEs have been applied to astrophysical use-cases before to disentangle data [INSERT SOME OF THE ASTRO VAE PAPERS], they have seen little use in photo-z estimation. This is most likely due to the unsupervised nature of VAEs, which can only indirectly be applied to the case of photo-z estimation by using latent representations of photometric observations as training data for a supervised ML model. In such an approach there is no assurance that the latent representations are optimal for the secondary supervised learning task Kingma et al. (2014). Extensive work has been done in extending VAEs to the semi-supervised setting, jointly learning disentangled representations *and* classifications of data (Kingma et al., 2014; Maaløe et al., 2016, 2019). The majority of this work has focused on classification tasks, but the definition of such VAEs do not exclude continous regression tasks (Maaløe et al., 2016).

2. Data

- SDSS DR14? + WISE
- ugriz psf and model mags + w1mpro and w2mpro (model mags?)
- Colors + i-band extinction
- How much photometric data vs spectroscopic data
- Train, Val and Test splits

3. PhotoVAEz

- Introduce package
- Outline section
- TODO: Replace $\mathbf{a}, \mathbf{b}$ with $\mathbf{l}_0, \mathbf{l}_1$ to make latent space notation a bit nicer. Alternativel use $\mathbf{x}_0, \mathbf{x}_1$, but may cause confusion with input vector $\mathbf{x}$.
- TODO: Make naming conventions for different model distribution a bit clearer.
- TODO: Add sampling over inputs during photo-z estimation
- TODO: Discuss best sampling approach with Kristoffer and Christa
- TODO: Compare results for the two sampling approaches.

⋆ ORCID xxxx-xxxx-xxxx-xxxx
⋆⋆ ORCID xxxx-xxxx-xxxx-xxxx
⋆⋆⋆ ORCID 0000-0002-8526-3963

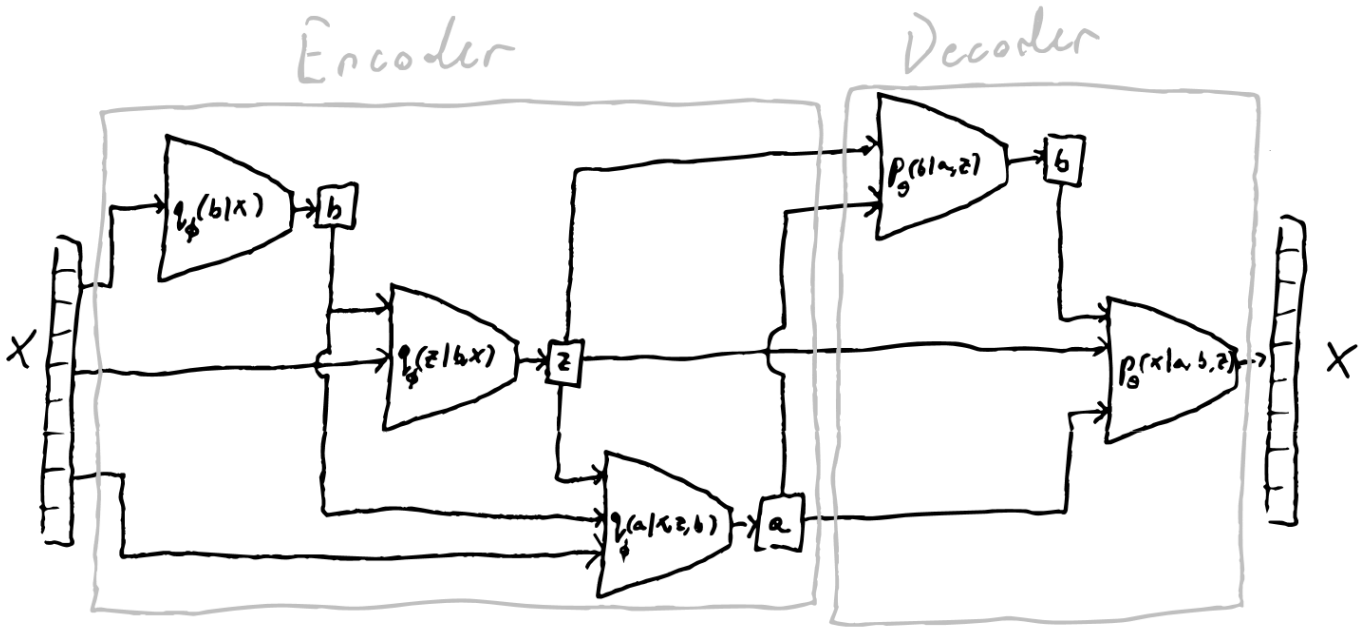

Fig. 1. Rough sketch, to be redone using tikz. Schematic illustrating the component distributions used in the SDGM, the base architecture of the PhotoVAEz model. The encoder takes as input the vector of photometric measurements $\mathbf{x}$ and consists of the variational distributions over the auxilliary latent space $\mathbf{b}$, the latent space $\mathbf{a}$ and $\log(z)$. The decoder takes as input samples from the encoder distributions and consists of the distributions over the auxilliary latent space $\mathbf{b}$ and the input photometry $\mathbf{x}$. All distributions shown are parameterized using neural networks.

3.1. The Skip Deep Generative Model

The PhotoVAEz-model is based on the *Skip Deep Generative Model (SDGM)*, a semi-supervised VAE, first introduced by Maaløe et al. (2016). Although the SDGM has been improved upon by later works (Maaløe et al., 2019), we choose this as a starting point due to the low dimensionality of the data as well as for the simpler implementation. In this section we review the architecture of the SDGM as well as the target of optimization, known as the *loss function*.

As inputs for the model we have observed data $\mathbf{X} = \mathbf{X}^p \cup \mathbf{X}^s$ where $\mathbf{X}^p = \{\mathbf{x}_1, \ldots, \mathbf{x}_{N_p}\}$ denotes the subset with only photometric data $\mathbf{x}_i$ available and $\mathbf{X}^s = \{(\mathbf{x}_1, z_1), \ldots, (\mathbf{x}_{N_s}, z_{N_s})\}$ denotes the subset of the data with both photometric data $\mathbf{x}_i$ and spectroscopic redshifts z_i. The SDGM assumes that the observed data is described by a generative model parameterized by a neural network with parameters θ

$$p_\theta(\mathbf{x}, z, \mathbf{a}, \mathbf{b}) = p_\theta(\mathbf{x}|\mathbf{a}, \mathbf{b}, z) p_\theta(\mathbf{b}|\mathbf{a}, z) p(\mathbf{a}) p(z). \tag{1}$$

Here $p(\mathbf{a})$ and $p(z)$ are priors over an unobserved set of latent variables $\mathbf{a} \in \mathbb{R}^A$ and the redshifts z respectively, $p_\theta(\mathbf{b}|\mathbf{a}, z)$ is a neural network parameterizing the conditional distribution over a set of auxilliary latent variables $\mathbf{b} \in \mathbb{R}^B$, and $p_\theta(\mathbf{x}|\mathbf{a}, \mathbf{b}, z)$ is a neural network parameterizing the likelihood of $\mathbf{x}$. The auxilliary latent variables $\mathbf{b}$ allow for dependencies between the latent variables $\mathbf{a}$ and redshifts z. This generative model corresponds to the decoder component in Fig. 1.

Given this generative model, the target of optimization for $(\mathbf{x}, z) \in \mathbf{X}^s$ is the marginal likelihood $p_\theta(\mathbf{x}, z)$ with respect to the parameters θ. This target is intractable due to the intractability of the posterior distribution $p(\mathbf{a}, \mathbf{b}|\mathbf{x}, z)$. To solve this, the posterior distribution is approximated by a variational distribution $q_\phi(\mathbf{a}, \mathbf{b}|\mathbf{x}, z) = q_\phi(\mathbf{a}|\mathbf{b}, \mathbf{x}, z) q_\phi(\mathbf{b}|\mathbf{x})$ parameterized by neural networks with parameters ϕ, leading to a variational lower bound (Maaløe et al., 2016)

$$\begin{aligned}
\log p(\mathbf{x}, z) &= \log \int_a \int_b p_\theta(\mathbf{x}, z, \mathbf{a}, \mathbf{b}) da db && (2)\\
&\geq \mathbb{E}_{q_\phi(\mathbf{a},\mathbf{b}|\mathbf{x},z)} \left[\log \frac{p_\theta(\mathbf{x}, z, \mathbf{a}, b}{q_\phi(\mathbf{a}, \mathbf{b}|\mathbf{x}, z)} \right] && (3)\\
&= -\mathcal{S}(\mathbf{x}, z). && (4)
\end{aligned}$$

For $\mathbf{x} \in \mathbf{X}^p$ we instead optimize the marginal likelihood $p_\theta(\mathbf{x})$ with respect to θ, where the redshift z is now considered a latent variable. As before a variational distribution $q_\phi(\mathbf{a}, \mathbf{b}, z|\mathbf{x}) = q_\phi(\mathbf{a}|z, \mathbf{b}, \mathbf{x}) q_\phi(z|\mathbf{b}, \mathbf{x}) q_\phi(\mathbf{b}|\mathbf{x})$ is introduced, leading to a variational lower bound

$$\begin{aligned}
\log p(\mathbf{x}) &= \log \int_a \int_b \int_z p_\theta(\mathbf{x}, z, \mathbf{a}, \mathbf{b}) da db dz && (5)\\
&\geq \mathbb{E}_{q_\phi(\mathbf{a},\mathbf{b},z|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, z, \mathbf{a}, b}{q_\phi(\mathbf{a}, \mathbf{b}, z|\mathbf{x})} \right] && (6)\\
&= -\mathcal{P}(\mathbf{x}). && (7)
\end{aligned}$$

The conditional redshift distribution $q_\phi(z|\mathbf{b}, \mathbf{x})$ appears in $-\mathcal{P}(\mathbf{x})$ but not $-\mathcal{S}(\mathbf{x}, z)$. To improve the predictive accuracy over redshift an explicit likelihood term over observed spectroscopic redshifts is added,

$$\tilde{\mathcal{S}}(\mathbf{x}, z) = \mathcal{S} + \alpha \mathbb{E}_{q_\phi(\mathbf{b}|\mathbf{x})} \left[-\log q_\phi(z|\mathbf{b}, \mathbf{x}) \right], \tag{8}$$

where α is a weight between the generative and predictive objectives. In this work we fix $\alpha = \frac{N_s + N_p}{N_s}$ to ensure equal weighting between the generative and predictive components of the loss function. The final loss function to be minimized over all observed data is then

$$\mathcal{J} = \sum_{(\mathbf{x},z)\in\mathbf{X}_s} \tilde{\mathcal{S}}(\mathbf{x}, z) + \sum_{x\in\mathbf{X}_p} \mathcal{P}(\mathbf{x}). \tag{9}$$

3.2. Architecture

VAEs such as the SDGM require the user to choose parameterizations for the distributions that make up the encoder and decoder. For ease of implementation and to improve training stability, we choose independent Gaussian distributions with shared standard deviations for all model distributions except redshift, were log-normal distributions are used,

$$p(\mathbf{a}) = \mathcal{N}(\mathbf{a}|\mathbf{0}, \mathbf{I}), \tag{10}$$
$$p(z) = \log \mathcal{N}(z|\mu_z, \sigma_z), \tag{11}$$
$$q_\phi(\mathbf{b}|\mathbf{x}) = \mathcal{N}(\mathbf{b}|\boldsymbol{\mu}_\phi(\mathbf{x}), \sigma^2_\phi(\mathbf{x})) \tag{12}$$
$$q_\phi(z|\mathbf{b}, \mathbf{x}) = \log \mathcal{N}(z|\mu_\phi(\mathbf{x}, \mathbf{b}), \sigma^2_\phi(\mathbf{x}, \mathbf{b})) \tag{13}$$
$$q_\phi(\mathbf{a}|\mathbf{x}, \mathbf{b}, z) = \mathcal{N}(\mathbf{a}|\mu_\phi(\mathbf{x}, \mathbf{b}, z), \sigma^2_\phi(\mathbf{x}, \mathbf{b}, z)) \tag{14}$$
$$p_\theta(\mathbf{b}|\mathbf{a}, z) = \mathcal{N}(\mathbf{b}|\boldsymbol{\mu}_\theta(\mathbf{a}, z), \sigma^2_\theta(\mathbf{a}, z)) \tag{15}$$
$$p_\theta(\mathbf{x}|\mathbf{a}, \mathbf{b}, z) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_\theta(\mathbf{a}, \mathbf{b}, z), \sigma^2_\theta(\mathbf{a}, \mathbf{b}, z)). \tag{16}$$

Log-normal distributions are chosen for the prior and approximate posterior redshift distributions to ensure that redshifts are strictly positive. The mean and log-variance μ_z and $\log(\sigma^2_z)$ of the prior redshift distribution are left as free parameters during training.

The encoder and decoder distributions in Eqs. 12 - 16 are parameterized by NNs consisting of $N_L = 3$ hidden layers with $N_H \in \{512, 256, 128\}$ neurons. The output layer of these NNs consists of $D_i + 1$ neurons, where $D_i \in \{D_\mathbf{a}, D_\mathbf{b}, D_z, D_x\}$ corresponds to the dimensionality of the distribution. These neurons have no activation function and output the mean vector $\boldsymbol{\mu}$ and log-variance $\log(\sigma^2)$. We choose the latent distributions to have shared dimensions $D_a = D_b = 10$ and we trivially have $D_z = 1$ and $D_x = 23$.

The expectation over the variational distributions in Eqs. 4, 7 and 8 are done using Monte-Carlo sampling (Kingma & Welling, 2022; Rezende et al., 2014),

$$\mathbb{E}_{q_\phi(\mathbf{a},\mathbf{b}|\mathbf{x},z)}\left[f(\mathbf{x}, z, \mathbf{a}, \mathbf{b})\right] \approx \frac{1}{N_{\rm MC}} \sum_i^{N_{\rm MC}} f(\mathbf{x}, z, a_i, b_i) \tag{17}$$
$$\mathbb{E}_{q_\phi(\mathbf{a},\mathbf{b},z|\mathbf{x})}\left[f(\mathbf{x}, z, \mathbf{a}, \mathbf{b})\right] \approx \frac{1}{N_{\rm MC}} \sum_i^{N_{\rm MC}} f(\mathbf{x}, z_i, a_i, b_i) \tag{18}$$
$$\mathbb{E}_{q_\phi(\mathbf{b}|\mathbf{x})}\left[f(\mathbf{x}, z, \mathbf{a}, \mathbf{b})\right] \approx \frac{1}{N_{\rm MC}} \sum_i^{N_{\rm MC}} f(\mathbf{x}, z, \mathbf{a}, b_i), \tag{19}$$

where $\mathbf{a}_i, b_i \sim q_\phi(\mathbf{a}, \mathbf{b}|\mathbf{x}, z)$, $\mathbf{a}_i, b_i, z_i \sim q_\phi(\mathbf{a}, \mathbf{b}, z|\mathbf{x})$ and $\mathbf{b}_i \sim q_\phi(\mathbf{b}|\mathbf{x})$, respectively. We choose $N_{\rm MC} = 1000$ as a compromise between computational speed and accuracy.

This architecture has been chosen by constructing the smallest non-probabilistic model that has the capacity to overfit the training data, see Appendix A.

3.3. Training

The model is trained using the Adam (Kingma & Ba, 2017) using an initial learning rate $r = 10^{-4}$ with an INSERT SCHEDULE HERE. Training is done for NUMBER OF EPOCHS with batch-size INSERT BATCHSIZE HERE.

During training VAEs are known to exhibit sudden large jumps in parameter gradients, leading to unstable regions of parameter space (Child, 2021; Vahdat & Kautz, 2021). To avoid this we adopt the approach of Child (2021) and apply gradient skipping. This approach skips a given parameter iteration if the gradient norm $||\nabla(\theta)||$ is larger than some chosen cutoff $\nabla_{\rm max}$. We choose $\nabla_{\rm max}$ =GRADIENT CUTOFF such that fewer than 0.01 percent of training iterations are skipped.

To inform the model of photometric uncertainties during training we resample each input vector as $\mathbf{x}^s_i \sim \mathcal{N}(\mathbf{x}|\mathbf{x}_i, \boldsymbol{\Sigma})$, where $\mathbf{x}_i$ is the input vector and $\boldsymbol{\Sigma}$ is the diagonal covariance matrix constructed from the photometric uncertainties. Using this resampling during training means that the model is exposed to INSERT NO. OF EPOCHS HERE samples of each input. The aim of this is to inform the model of regions of photometric space that have larger observational uncertainties, such that the model distributions become correspondingly wider.

3.4. Photo-z Estimation

After training, photo-z estimation for a photometric source $\mathbf{x}_i$ can be performed by marginalizing the predictive variational distribution $q_\phi(z|\mathbf{b}, \mathbf{x}_i)$ over the auxilliary latent variable $\mathbf{b}$,

$$q_\phi(z|\mathbf{x}_i) = \int q_\phi(z|\mathbf{b}, \mathbf{x}_i) db \tag{20}$$
$$= \int q_\phi(z|\mathbf{b}, \mathbf{x}_i) q_\phi(\mathbf{b}|\mathbf{x}_i) db \tag{21}$$
$$\approx \frac{1}{N} \sum_k^N q_\phi(z|b_k, \mathbf{x}_i). \tag{22}$$

The predictive and auxilliary latent NN parameters are informed by both the photometric and spectroscopic data used during training due to the joint loss function in Eq. 9. By approximating the marginalized variational distribution $q_\phi(z|\mathbf{x}_i)$ we discard the remaining information contained in the latent distribution generative distributions $q_\phi(\mathbf{a}|\mathbf{x}, z, \mathbf{b})$, $p_\theta(\mathbf{b}|\mathbf{a}, z)$ and $p_\theta(\mathbf{x}|\mathbf{a}, \mathbf{b}, z)$. To retain this information we instead approximate the true posterior redshift distribution using importance sampling (Bishop, 2006),

$$p_\theta(z|\mathbf{x}_i) = \int \frac{p_\theta(\mathbf{x}_i, z, \mathbf{a}, \mathbf{b})}{p_\theta(\mathbf{x}_i)} da db \tag{23}$$
$$\approx \frac{1}{N} \sum_{k=1}^N w_k(z, \mathbf{x}_i) \frac{p_\theta(\mathbf{x}_i|a_k, b_k, z) p(z)}{p(\mathbf{x}_i)}, \tag{24}$$

where $\mathbf{a}_k \sim q_\phi(\mathbf{a}|b_k, z, \mathbf{x}_i)$, $b_k \sim q_\phi(\mathbf{b}|\mathbf{x}_i)$ and

$$w_k(z, \mathbf{x}_i) = \frac{p_\theta(b_k|a_k, z) p(a_k)}{q_\phi(a_k|b_k, z, \mathbf{x}_i) q_\phi(b_k|\mathbf{x}_i)}. \tag{25}$$

For a detailed derivation, see Appendix B. Eq. 24 can not be evaluated analytically due to the intractable marginal distribution $p(\mathbf{x}_i)$. Instead we sample from the posterior redshift distribution using Hamiltonian Monte Carlo via the MCMC-package BlackJax (Cabezas et al., 2023).

1. OUTLIER DETECTION USING JOINT DISTRIBUTION

3.5. Evaluation Metrics

1. LSST performance metrics
2. PDF-based outliers
3. Percentile-Percentile plot / norm

4. Results

– Training results

4.1. Test-Set Evaluation

- Latent space separation
- Point-Estimate Metrics
- Point estimates vs spec-z
- PDF Metrics
- Outlier detection using joint distribution

5. Discussion

Appendix A: Model Architecture Optimization

To be written

Appendix B: Photo-z Posterior Sampling

To be written

References

Bishop, C. M. 2006, Pattern Recognition and Machine Learning, Information Science and Statistics (New York: Springer)
Cabezas, A., Lao, J., & Louf, R. 2023, Blackjax: A sampling library for JAX
Child, R. 2021, Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images
Kingma, D. P. & Ba, J. 2017, Adam: A Method for Stochastic Optimization
Kingma, D. P., Rezende, D. J., Mohamed, S., & Welling, M. 2014, Semi-Supervised Learning with Deep Generative Models
Kingma, D. P. & Welling, M. 2022, Auto-Encoding Variational Bayes
Maaløe, L., Fraccaro, M., Liévin, V., & Winther, O. 2019, BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling
Maaløe, L., Sønderby, C. K., Sønderby, S. K., & Winther, O. 2016, Auxiliary Deep Generative Models
Rezende, D. J., Mohamed, S., & Wierstra, D. 2014, Stochastic Backpropagation and Approximate Inference in Deep Generative Models
Siddharth, N., Paige, B., van de Meent, J.-W., et al. 2017, Learning Disentangled Representations with Semi-Supervised Deep Generative Models
Vahdat, A. & Kautz, J. 2021, NVAE: A Deep Hierarchical Variational Autoencoder