AN INFERENCE QUALITY ASSESSMENT FRAMEWORK FOR DEEP NORMALIZING FLOWS

Kaan Güney Keklikçi
Faculty of Engineering and Natural Sciences
Sabanci University
Tuzla, Istanbul 34956
kaanguney@sabanciuniv.edu

October 23, 2021

ABSTRACT

State-of-the-art deep learning practices have successfully acquired the ability to model posterior distributions of publicly available medical data without violating patient data confidentiality. However, modern research requires a quantifiable framework which allows researchers to sample from and evaluate the true posterior distribution of medical data due to data deprivation in the field. Therefore, we propose a modifiable framework of recent novel approaches, an interconnected model of β-VAE and deep normalizing flow to generate and assess medical data that adheres to any domain of inference modeling in the health sector. Our methodology surpasses the barriers of privacy and deprivation by efficient learning of the underlying true posterior distribution while complying with statistical divergence and approximation techniques.

Keywords variational inference · approximation · statistical divergence · deep normalizing flows · β-VAE

1 Introduction

Conventional variational inference methodologies have recently demonstrated exquisite replication of multivariate data proportional to novel deep neural approaches. Although stochastic by nature via reparametrization, dispersion among output samples suggest careful examination of existing framework. In translation, commonly employed inference approaches, for instance, generative adversarial networks (GANs) procure the likelihood but are yet to circumvent an appropriate posterior [10]. Furthermore, existing methodology lacks a framework-like approach where metric evaluation that is intrinsic to target density has remained ineffectual and evaluation of the true posterior has been considered rather rigorous. In this paper, we discuss the significance of state-of-the-art deep neural approaches by proposing mixture framework of density estimation with β-VAEs and normalizing flows. First, we learn a latent distribution from a β-VAE to put implicit pressure Kullback-Leibler divergence between an observation set $\{x^{(t)}\}_{t=1}^{T}$ and the latent distribution $\{z^{(t)}\}_{t=1}^{T}$. Next, we specifically restrict our existing implementation to be interconnected to normalizing flows with deep neural architecture to compare the stability and robustness of vectorized computations across latent samples from the β-VAE. We demonstrate scalability and robustness of our framework on both training and test data of lung cancer patients consisting of FVC (forced vital capacity) measures of respiration volume under Kolmogorov-Smirnov test in a two dimensional setting while abiding by usual significance levels.

2 Related Work

2.1 Evidence Lower Bound (ELBO)

Given an encoder-decoder network, variational autoencoders (VAEs) are optimized thorough stochastic gradient-based maximization of ELBO, i.e. *variational lower bound* [9]. In essence, all variational inference methodologies can be reduced to a single equation as follows:

$$p(x_i) = \int_z p(x_i|z)p(z) \tag{1}$$

Conveniently, target density $p(x_i)$ is achievable with Monte Carlo sampling, however, as the number of dimensions grow, sample size to correctly identify a stochastic mapping from observation set to target density grows exponentially [5]. Note that variational autoencoders are inference mechanisms that are capable of maximizing the marginal likelihood, that is, $p_\theta(x)$. To do so, a parametric model, $q_\phi(z|x)$ is optimized where $q_\phi(z|x)$ is a true approximation of the posterior, $p_\phi(z|x)$ [10]. Therefore, the parametric model rectifies the prior density to approximate the posterior by optimizing ϕ which represents variational parameters of encoder-decoder network such that:

$$q_\phi(z|x) \approx p_\theta(z|x) \tag{2}$$

For any parametric inference model including variational autoencoders, the approximation mentioned in 2 is modeled using the same architecture. Encoder-decoder network is models a joint distribution of latent samples from some space Z and observations,$\{x^{(t)}\}_{t=1}^{T}$, from the data source. Afterwards, joint density is factorized such that:

$$p_\theta(x, z) = p_\theta(z)p_\theta(x|z) \tag{3}$$

Equation 3 relaxes the posterior distribution to a tractable posterior density, $p_\theta(z|x)$ that lays the foundation of ELBO maximization. In any case an appropriate posterior relies on equation 1 where one can minimize the most commonly used f-divergence, Kullback-Leibler (KL) divergence to convert the optimization problem into a tractable one [5]. Using Jensen's inequality, KL divergence can be re-written as:

$$D = E[log\ q(z)] - E[log\ p(z|x)] \tag{4}$$

where $q \in Q$ is an approximation of $p \in P$. Followingly;

$$D = D_{KL}(Q(Z)||P(Z|X)) \tag{5}$$

where Q(Z) is the density representation of arbitrary latent samples, $\{z_1, z_2, \ldots, z_n\}$ over latent space Z and the term D is the explicitly defined KL divergence among the prior (evidence) and the posterior (belief). For further analysis, posterior distribution is split into two components, that is, the joint distribution of observational and latent space in addition to marginalised log-likelihood of target density, that is, evidence:

$$D = E[log\ q(z)] - E[log\ p(z, x)] + E[log\ p(x)] \tag{6}$$

Note that $E[log\ p(x)]$ term in 6 is independent of latent space Z, hence, can be re-written as follows:

$$D = E[log\ q(z)] - E[log\ p(z,x)] + log\ p(x) \tag{7}$$

As it depends on the target density that cannot be modeled by the recognition model [10] in run time, f-divergence cannot be computed in this formulation. An alternative formulation, however, exists and is far more intuitive than minimizing KL divergence by itself [2, 5, 20], that is, ELBO. Rearranging terms in equation 7, a new formulation yields:

$$\begin{aligned} -D + log\ p(x) &= E[log\ p(z,x)] - E[log\ q(z)],, & (8)\\ &= E[log\ p(z,x)] - E[log\ q(z)], & (9)\\ &= ELBO(Q) & (10) \end{aligned}$$

As demonstrated in 8, ELBO objective function maximization is equivalent to minimizing KL divergence and the objective function itself is decomposed into a joint distribution of some $z \in Z$ and $x \in X$ that approximates the true posterior probability along with evidence. However, this version of *variational lower bound* remains rather rigorous and non-intuitive. A more compact, elaborate approach is rewriting ELBO as a combination of marginalised log-likelihood and KL divergence such that:

$$\begin{aligned} ELBO(Q) &= E[log\ p(z,x)] - E[log\ q(z)], & (11)\\ &= E[log\ p(x|z)] + E[log\ p(z) - log\ q(z)], & (12)\\ &= E[log\ p(x|z)] - D_{KL}(Q(Z)||P(Z)) & (13) \end{aligned}$$

Hence, decomposition of ELBO yields two critical components that can be optimized concurrently [9], that is, $E[logp(x|z)]$ and $D_{KL}(Q(Z)||P(Z))$, marginalised log-likelihood and KL divergence between approximate and variational probability density of the prior respectively. Asymptotic behavior of KL divergence is either 0 at minimum or +∞ at maximum. As the encoder network places approximated density in the stochastic space of the prior where prior density is non-existent, the asymptotic behavior is $\frac{0}{P(Z)} \approx +\infty$, therefore, variational lower bound (ELBO) is bound to fit prior density in order to avoid severe punishment of the objective function. As a final remark, a modification of ELBO exists for the conventional variational autoencoder, namely β-VAE [9] which enforces the algorithm to put implicit pressure on constraining KL divergence term of the ELBO.

2.2 Masked Autoencoder for Distribution Estimation (MADE)

Robustness and efficiency of MADE is based on its autoregressive property. Compared to existing neural architectures in the field, primarily generative adversarial nets [7] and deep generative decoders [19], MADE scales faster and preserves the efficiency of a single pass through a vanilla autoencoder [6]. The key feature of MADE is to zero-out some connections in the hidden layers by making sure a valid trajectory along some neurons always exist to preserve autoregressive property. To enforce autoregressiveness in the architecture, weight matrices of form $\mathbf{W}$ are instantiated. Note that, output matrix in the network, in particular, is labeled as $\mathbf{V}$ and the binary masks associated with $\mathbf{W}, \mathbf{V}$ are denoted by $\mathbf{M^W}$ and $\mathbf{M^V}$. Next, a connectivity integer $m(k)$ is sampled from a uniform discrete distribution to apply during a single forward-pass of the network between two consequent dimensions $D-1$ and D. One takeaway is to omit $m(k) = 0$ and $m(k) = D$. In translation, input layer is assumed to produce constant hidden units and final layer, D, always yields a valid trajectory along some path in the network which makes it infeasible to model conditionals of form $p(x_d|x_{<d})$ for some neuron d included in D. Succinctly, autoregessiveness in the network is obtained by the set

of systematic equations below.

$$M^W_{k,d} = 1_{m(k)\geq d} = \begin{cases} 1 & if\ m(k) \geq d \\ 0 & otherwise \end{cases} \tag{14}$$

$$M^V_{d,k} = 1_{d\geq m(k)} = \begin{cases} 1 & if\ d > m(k) \\ 0 & otherwise \end{cases} \tag{15}$$

for $d \in \{1, \ldots, D\}$ and $k \in \{1, \ldots, K\}$.

2.3 Generative Adversarial Networks (GANs)

Generative adversarial networks (GANs) are deep generative models where stochastic mapping of a prior distribution is obtained in a solely unsupervised setting. Although adversarial nets have been widely used in literature, particularly in speech recognition [15, 22], both expensive labelling of the architecture and use of a single generator in the conventional setting hampers the scalability of and decreases efficiency of the network in parallel. Auxiliary classification of labels can enrich discriminator, D, decision making but is yet to hinder the ability of the network to infer specific factors of noise [17].

2.4 Normalizing Flows

Despite achieving remarkable success in image generation of GANs and other novel approaches to use the network as a feature extractor rather than a classifier [22], scalability of GANs has remained fragile. In consequence, normalizing flows emerged as a novel technique that alleviated the burden of scalability of inference networks. Normalizing flows follow a chained series of invertible transformations to approximate a target density $p_\theta(x)$ parametrized by θ [21, 12]. One key component of a normalizing flow is the choice of a differentiable and invertible transformation, f of a base distribution $\pi_y(y)$. Given an observation set of $\{y_1, y_2, \ldots, y_n\}$, a normalizing flow parametrizes a probability distribution function $p(y)$ where y $\sim \pi_y(y)$ [14]. Hence, a normalizing flow can be modeled such that:

$$p(x) = \pi_y(f^{-1}(x))\Big|det\Big(\frac{\partial f^{-1}}{\partial x}\Big)\Big| \tag{16}$$

In a multivariate setting, the flow maximizes the local volume change during maximum likelihood estimation by storing expansions and contractions in volume in the Jacobian matrix, therefore, this technique computes an approximate of target density while maintaining scalability [4]. $\Big|det\Big(\frac{\partial f^{-1}}{\partial x}\Big)\Big|$ is called the Jacobian determinant of function f, that is, $\det J_f$ where the the Jacobian matrix is written such that:

$$J_f = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_K}{\partial x_1} & \frac{\partial f_K}{\partial x_2} & \cdots & \frac{\partial f_K}{\partial x_n} \end{bmatrix} \tag{17}$$

Note that, particularly for normalizing flows, $J_{f^{-1}}$ is used since flow-based models follow the path of a serializable invertible transformations as mentioned in 2.4. This structured approach equips normalizing flows with the ability to perform computations in parallel for multivariate inputs by Graphics Processing Units (GPUs) [14].

3 Motivation

Variational inference took a major leap with introduction of MADE for density estimation as an alternative to Markov Chain Monte Carlo methods given their longevity of execution [1]. Constructing robust estimators subsequently yielded tractable Jacobian determinants composed of a series of invertable, affine transformations where computations were vectorized to output samples equivalent to a single forward pass through a vanilla autoencoder. Other deep normalizing flow approaches, for example, masked autoregressive flow (MAF) [14] and inverse autoregressive flow (IAF) [11] quickly embodied MADE in their methodology and density estimation became a much more feasible task despite apparent challenges of computing a tractable Jacobian and obvious ambiguity of quality assessment of the true posterior distribution. Our motivation relates to latter as far more critical due to sensitive nature of medical data. Although it is possible to infer statistically significant results from deep normalizing flows without a rich, initial latent density, we find that applying a sequence of invertible transformations to a diagonal covariance matrix of a posterior latent's density stabilizes Kullback-Leibler divergence. Formally, we can sample a reparametrized latent variable, z and vectorize distribution parameters as a lower triangular matrix to boost scalability of deep normalizing flows, thereby, achieving higher distribution similarity scores in a unified, statistical approach.

4 Methodology

4.1 Framework

As outlined in 3, our proposed work is intended to serve as an variational inference framework, precisely for medical data. In order to ensure the validity of a reliable surrogate for conventional approaches in 2, we develop a new, parametrizable sampling approach by sequential sampling of a latent variable $z \in Z^n$, that is, the latent space. While conventional approaches model a joint distribution of observational and prior distributions as $p_(z, x)$, the part of our proposed framework in relation to encoder-decoder network maximizes the ELBO to output a multivariate normal distribution with covariance matrix $\mathbf{\Sigma}$, observational space X and mean vector μ. Therefore, output of encoder and decoder in sequential fashion are put both through an implicit batch normalization [8] layer where there exists *internal covariate shift*, i.e. no change in loss despite gradient update of parameters of the network. Systematic equations of the implicit batch normalization defined in the encoder-decoder network yield the following multivariate normal distribution such that:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}, \ \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \dots \\ \mu_n \end{bmatrix}, \sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2n} \\ \dots & \dots & \dots & \dots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_{nn} \end{bmatrix} \tag{18}$$

Reparametrized by the *recognition model*, observational space is transcribed into the multivariate normal distribution defined in 4.1. While reducing internal covariate shift per batch, joint use of the batch normalization layer in the β-VAE enforces an additional constraint on KL divergence. Whether it is appropriate to use a β-VAE or a conventional autoencoder is non-trivial and depends on the optimization problem. Since nearly exact approximation of the posterior probability is crucial in medical data, β-VAE is quite possibly the more sensible alternative of the two approaches. Note that in a conventional variational autoencode, value of β is trivial, that is, $\beta = 1$. However, as the representative power of the normalizing flow is dependent on a nearly exact approximation of $p_\theta(z|x)$, we choose $\beta = 10$, enforcing a very strict punishment on the initial inference mechanism of the framework. Novelty of our framework relies on tractability of Gaussian distributions. Samples of the resulting tractable posterior $q_\phi(z|x)$ out of β-VAE are used to compute μ and

σ of the density across every batch and sample such that:

$$\mu = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \\ \mu_{31} & \mu_{32} \\ \mu_{41} & \mu_{42} \end{bmatrix}, \sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \\ \sigma_{31} & \sigma_{32} \\ \sigma_{41} & \sigma_{42} \end{bmatrix}, \tag{19}$$

$$\mu_\phi = \begin{bmatrix} \frac{\mu_{11}+\mu_{12}}{2} \\ \frac{\mu_{21}+\mu_{22}}{2} \\ \frac{\mu_{31}+\mu_{32}}{2} \\ \frac{\mu_{41}+\mu_{42}}{2} \end{bmatrix}, \sigma_\phi = \begin{bmatrix} \frac{\sigma_{11}+\sigma_{12}}{2} \\ \frac{\sigma_{21}+\sigma_{22}}{2} \\ \frac{\sigma_{31}+\sigma_{32}}{2} \\ \frac{\sigma_{41}+\sigma_{42}}{2} \end{bmatrix} \tag{20}$$

Then, a multivariate normal diagonal matrix σ_ϕ is filled diagonally to be input to a deep normalizing flow for applying a series of invertible transformations. Deep normalizing flows considered in the framework are masked autoregressive flow (MAF), inverse autoregressive flow (IAF), RealNVP [4] and NICE [3]. The framework is solely based on vectorized computations due neural selection of flow-based models. Study demonstrates exceptional approach in mirroring noisy data which is likely an indication of putting implicit pressure on KL divergence during the first layer of the framework.

4.2 Data

Noisy moons: This is a two dimensional toy data generator that is convenient for use of eliciting framework performance against data with inherent noise. Data source is sci-kit learn [16].
Thoracic surgery: The dataset is a UCI dataset [24]. It is commonly for classification of post-operative life expectancy in lung cancer patients. We are particularly interested in this dataset due to two reasons. First, it contains critical, continuous patient data but we are only interested in two features instead of labels since the task at hand is a generative process. In translation, the whole framework operates on a set of forced vital capacity (FVC) and FEV1, that is volume exhaled at the last second of forced respiration. True generative process of these features is decisive in diagnosis of lung disease [13].

5 Experiments

We conducted two sets of experimentation where noisy moons data was used for on-the-fly, fast evaluation of the proposed methodology and thoracic surgery data was used for statistical evaluation of the framework with KL divergence, mean absolute error, and cross entropy . While providing these measures along with transformed densities, confidence intervals given significance level $\alpha = 0.05$ are also constructed for each sequential posterior batch output by the decoder. All experiments have been conducted with 4 autoregressive flows with 2 hidden layers and 256 neurons per flow for 600 epochs while comprising some masked neurons in the setup. Note that this was applied to neurons to zero-out some connections of the network for regularization as previously mentioned in 14.

For fast quality check before any quantifiable assessment, we set a hold-out split ratio of 0.5 and a batch size of 32 for fast execution. Convergence densities per algorithm, namely MAF, IAF and RealNVP showed significant similarities. For NICE, J_ϕ matrix must be ensured to produce a constant J_ϕ determinant for which we did not specifically made any additional checks, therefore, the visualizations that transcribe the localized changes in volume for this particular flow are omitted. Moreover, we deliberately fix a batch size of 64, doubling the previous batch size, in order to see the sampled densities more clearly while applying the framework to facilitate the maximization of $\left| det\left(\frac{\partial f_\phi^{-1}}{\partial x}\right)\right|$, that is, the inverse Jacobian determinant. The results out of the framework for the noisy moons data yield

the following reciprocal contractions and expansions in the inferred samples out of the framework along with target probability density of noisy moons data in Figure 1.

(a) $\mathcal{N}(\mu_\phi, \Sigma_\phi)$ (b) MAF (c) Permutation (d) MAF

(e) Permutation (f) MAF (g) Permutation (h) MAF

Figure 1: Chain of invertible transformations

(a) $p_\phi(x|z)$ (b) $q_\phi(x|z)$

Figure 2: Reciprocal expansions and contractions

As demonstrated in 2, we are able to achieve a remarkable resemblance of $p_\phi(x|z)$ of by $q_\phi(x|z)$, meaning that variational parameters, ϕ, are learned successfully by the framework. However, the main contribution of this paper is to design a generative process that is precise enough to avoid dispersion among samples while conserving analytically proven significance. Therefore, we choose to evaluate the true inferential power of our framework on thoracic surgery data 4.2. For the purpose of analysis, we sample sequentially from the decoder of β-VAE and measure the goodness of fit by a 2 sample Kolmogorov-Smirnov (KS) test [23] while measuring statistical properties mentioned in 5. For every batch, we compute a pooled estimator to ensure the property of unequal variances between two batches as if they are from two different Gaussian populations, a similar approach to [18]. Finally, we provide the density estimation results for each deep normalizing flow that is used in the analysis to finalize experiments.

We have previously outlined the importance of obtaining a near exact copy of target likelihood, that is, $p_\phi(x|z)$ from the decoder of β-VAE to meet expectations of the core problem of inference in 1. After successfully modeling an appropriate posterior, we sequentially sample from the decoder for 100 runs for each deep normalizing flow input and

collect statistical results under $\alpha = 0.5$ for a 95% CI. Statistical metrics gathered in the metric aggregation phase of experimentation helps us measure the stability of the framework per deep normalizing flow.

Table 1: Metric Computation per Deep Normalizing Flow

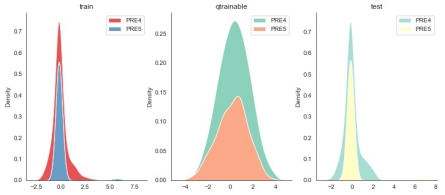
95% Confidence Intervals			
Normalizing Flow	KL Divergence	Cross Entropy	Mean Absolute Error
MAF	(-0.167, 0.065)	(-0.147, 0.179)	(-0.015, 0.003)
IAF	(-0.093, 0.122)	(-0.129, 0.198)	(0.002, 0.022)
RealNVP	(-0.116, 0.114)	(-0.17, 0.131)	(-0.011, 0.009)
NICE	(-0.027, 0.205)	(-0.13, 0.143)	(-0.011, 0.006)

Table 2: Multivariate Kolmogorov-Smirnov Test

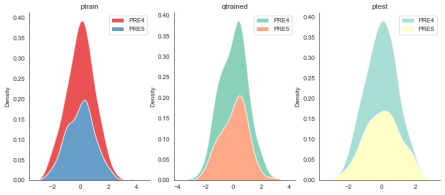
p-values, 100 Runs		
Normalizing Flow	KS (training)	KS (testing)
MAF	0.740	0.145
IAF	0.540	0.477
RealNVP	0.574	0.267
NICE	0.002	0.0001

6 Conclusion

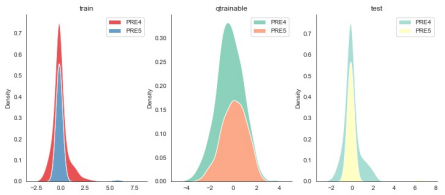
[H] Findings in 1 instantiate an interesting discussion. Without any explicitly defined inversion, masked autoregressive flow is more accurate but also prone to dispersed samples more compared to inverse atuoregressive flow. Many small errors made by each flow are equivalent to a single dispersed, erroneous sample. Due to the absolute minimum value of the confidence in mean absolute error being larger in MAF compared to IAF, our experiment suggests that IAF is a more reliable algorithm. In parallel, the experiment also suggests that RealNVP outweighs the inverse autoregressive flow in terms of expressive power as per KL divergence and cross entropy and is as stable as the masked autoregressive flow. On the other hand, NICE struggles to fit the likelihood, that is, $p_\phi(x|z)$ based on aggregated KL divergence and cross entropy statistics. However, since we did not explicitly check the $J_\phi^{f^{-1}}$ determinant of the flow is constant, i.e. volume-preserving, the results are not to be as reliant on as the rest of the flow models analyzed in the framework. On top of statistical aggregation, we measure KS statistics of both training and testing tests. Fixing a null hypothesis as $\mu_0 = \mu_1$, we test each version of our methodology for goodness-of-fit on training and testing sets. After the test, we find that results are inline with our findings in 1 and null hypothesis is rejected for each implementation except NICE due to the drawback in experimentation setup. It is detrimental to underline that Kolmogorov-Smirnov test discloses the main findings of this paper. The tests involve samples from the prior latent variable z and posterior approximation, $q_\phi(z|x)$ in order to measure inference quality. Statistics regarding 2 encourage that RealNVP is the most significant in testing while MAF surpasses every flow in training. All flows deny the null hypothesis, hence, statistically prove that approximation, $q_\phi(z|x)$ is in fact very significant. Provided we dealt with restrictive data, the generative performance of the framework proves to be a reliable alternative to existing state-of-the-art methodology in 2. We conclude our work by illustrating the generative process per variant of the framework in 3.



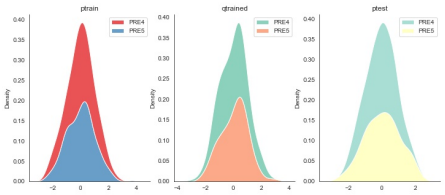
(a) $p_\phi(x|z)$, MAF



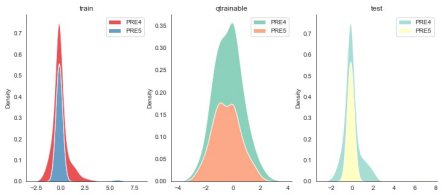
(b) $q_\phi(x|z)$, MAF



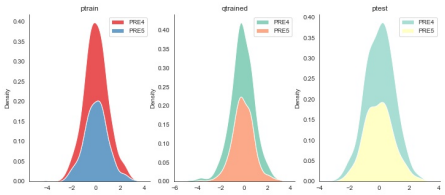
(c) $p_\phi(x|z)$, IAF



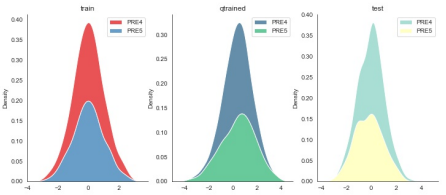
(d) $q_\phi(x|z)$, IAF



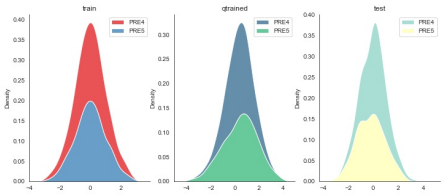
(e) $p_\phi(x|z)$, RealNVP



(f) $q_\phi(x|z)$, RealNVP



(g) $p_\phi(x|z)$, NICE



(h) $q_\phi(x|z)$, NICE

Figure 3: Framework density transformations

References

[1] Backenköhler, Michael, Bortolussi, Luca, and Wolf, Verena. *Variance Reduction in Stochastic Reaction Networks using Control Variates*. 2021. arXiv: 2110.09143 [stat.ME].

[2] Bishop, Christopher M. and Tipping, Michael. *Variational Relevance Vector Machines*. 2013. arXiv: 1301.3838 [cs.LG].

[3] Dinh, Laurent, Krueger, David, and Bengio, Yoshua. *NICE: Non-linear Independent Components Estimation*. 2015. arXiv: 1410.8516 [cs.LG].

[4] Dinh, Laurent, Sohl-Dickstein, Jascha, and Bengio, Samy. *Density estimation using Real NVP*. 2017. arXiv: 1605.08803 [cs.LG].

[5] Ganguly, Ankush and Earp, Samuel W. F. *An Introduction to Variational Inference*. 2021. arXiv: 2108.13083 [cs.LG].

[6] Germain, Mathieu et al. *MADE: Masked Autoencoder for Distribution Estimation*. 2015. arXiv: 1502.03509 [cs.LG].

[7] Goodfellow, Ian J. et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[8] Ioffe, Sergey and Szegedy, Christian. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. arXiv: 1502.03167 [cs.LG].

[9] Kingma, Diederik P and Welling, Max. *Auto-Encoding Variational Bayes*. 2014. arXiv: 1312.6114 [stat.ML].

[10] Kingma, Diederik P. and Welling, Max. "An Introduction to Variational Autoencoders". In: *Foundations and Trends® in Machine Learning* 12.4 (2019), pp. 307–392. ISSN: 1935-8245. DOI: 10.1561/2200000056. URL: http://dx.doi.org/10.1561/2200000056.

[11] Kingma, Diederik P. et al. *Improving Variational Inference with Inverse Autoregressive Flow*. 2017. arXiv: 1606.04934 [cs.LG].

[12] Liao, Huadong, He, Jiawei, and Shu, Kunxian. *Generative Model with Dynamic Linear Flow*. 2019. arXiv: 1905.03239 [cs.LG].

[13] Nazi, Zabir Al et al. *Fibro-CoSANet: Pulmonary Fibrosis Prognosis Prediction using a Convolutional Self Attention Network*. 2021. arXiv: 2104.05889 [eess.IV].

[14] Papamakarios, George, Pavlakou, Theo, and Murray, Iain. *Masked Autoregressive Flow for Density Estimation*. 2018. arXiv: 1705.07057 [stat.ML].

[15] Pascual, Santiago, Serrà, Joan, and Bonafonte, Antonio. *Towards Generalized Speech Enhancement with Generative Adversarial Networks*. 2019. arXiv: 1904.03418 [cs.SD].

[16] Pedregosa, F. et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[17] Phan, Huy et al. "Improving GANs for Speech Enhancement". In: *IEEE Signal Processing Letters* 27 (2020), pp. 1700–1704. ISSN: 1558-2361. DOI: 10.1109/lsp.2020.3025020. URL: http://dx.doi.org/10.1109/LSP.2020.3025020.

[18] Raymaekers, Jakob and Zamar, Ruben H. "Pooled variable scaling for cluster analysis". In: *Bioinformatics* 36.12 (Apr. 2020). Ed. by JonathanEditor Wren, pp. 3849–3855. ISSN: 1460-2059. DOI: 10.1093/bioinformatics/btaa243. URL: http://dx.doi.org/10.1093/bioinformatics/btaa243.

[19] Schuster, Viktoria and Krogh, Anders. *The deep generative decoder: Using MAP estimates of representations*. 2021. arXiv: 2110.06672 [cs.LG].

[20] Sreekumar, Sreejith and Goldfeld, Ziv. *Neural Estimation of Statistical Divergences*. 2021. arXiv: 2110.03652 [math.ST].

[21] Tomczak, Jakub M. *General Invertible Transformations for Flow-based Generative Modeling*. 2021. arXiv: 2011.15056 [cs.LG].

[22] Wang, Ce, Chen, Zhangling, and Shang, Kun. *Label-Removed Generative Adversarial Networks Incorporating with K-Means*. 2019. arXiv: 1902.06938 [cs.LG].

[23] Yamaguchi, Akihiro and Saito, Asaki. *Second-level randomness test based on the Kolmogorov-Smirnov test*. 2021. arXiv: 2110.08023 [stat.ME].

[24] Zikeba, Maciej et al. "Boosted SVM for extracting rules from imbalanced data in application to prediction of the post-operative life expectancy in the lung cancer patients". In: *Applied Soft Computing* (2013). DOI: https://doi.org/10.1016/j.bbr.2011.03.031.