Journal of Machine Learning Research 22 (2021) 1-41 Submitted 1/21; Revised 5/21; Published 9/21

A Generalised Linear Model Framework for β-Variational Autoencoders based on Exponential Dispersion Families

Robert Sicks * ROBERT.SICKS@ITWM.FRAUNHOFER.DE
Department of Financial Mathematics
Fraunhofer ITWM
Kaiserslautern

Ralf Korn KORN@MATHEMATIK.UNI-KL.DE
Department of Financial Mathematics
TU Kaiserslautern
Kaiserslautern

Stefanie Schwaar STEFANIE.SCHWAAR@ITWM.FRAUNHOFER.DE
Department of Financial Mathematics
Fraunhofer ITWM
Kaiserslautern

Editor: Shakir Mohamed

Abstract

Although variational autoencoders (VAE) are successfully used to obtain meaningful low-dimensional representations for high-dimensional data, the characterization of critical points of the loss function for general observation models is not fully understood. We introduce a theoretical framework that is based on a connection between β-VAE and generalized linear models (GLM). The equality between the activation function of a β-VAE and the inverse of the link function of a GLM enables us to provide a systematic generalization of the loss analysis for β-VAE based on the assumption that the observation model distribution belongs to an exponential dispersion family (EDF). As a result, we can initialize β-VAE nets by maximum likelihood estimates (MLE) that enhance the training performance on both synthetic and real world data sets. As a further consequence, we analytically describe the auto-pruning property inherent in the β-VAE objective and reason for posterior collapse.

Keywords: Variational autoencoders, ML-estimation, EDF observation models, loss analysis, posterior collapse

1. Introduction

Variational autoencoders (VAE) (Kingma and Welling (2014) and Rezende et al. (2014)) are described by Goodfellow et al. (2016) as an "excellent manifold learning algorithm" due to the fact that the model is forced "to learn a predictable coordinate system that the encoder can capture". VAE do so by using a regularization term in order to get to low energy regions. According to LeCun (2020), regularization like in the VAE case helps to keep the energy

∗. Corresponding author

function smooth, which is desirable for the model to learn meaningful dependencies (e.g. to fill blanks). In contrast, maximum likelihood approaches push down the energy surface only at training sample regions. Therefore, their inherent objective is "to make the data manifold an infinitely deep and infinitely narrow canyon" (see LeCun 2020).

Learning meaningful dependencies is a desirable concept for advancing deep learning. Hence, there exists an interest in understanding and developing VAE. Recent work aims at explaining and overcoming well-known pitfalls of VAE, such as spurious global optima (see Dai and Wipf 2019), posterior collapse (see Lucas et al. 2019) or prior posterior mismatch (see Dai and Wipf 2019 and Ghosh et al. 2020). In these works, a Gaussian observation model distribution is assumed.

In this work, we answer the following research question:

> *Is there a way to generalize the loss analysis of β-VAE based on the observation model distribution?*

For this, we establish a connection between β-VAE and generalized linear models (GLM) and provide a framework for analysing β-VAE based on the observation model distribution. By doing so, we generalize works of Dai et al. (2018), Lucas et al. (2019) and Sicks et al. (2020). We provide an approximation to the evidence lower bound (ELBO), which is exact in the Gaussian distribution case (see also Dai et al. 2018 and Lucas et al. 2019) and a lower bound for the Bernoulli distribution case (see also Sicks et al. 2020). Further, we analyse the maximum likelihood estimates (MLE) of this approximation. Given the MLE, we

- propose a MLE based initialization and show that the training performance of a VAE net can be enhanced.
- find an analytical description of the auto-pruning property of β-VAE, a reason for posterior collapse.
- analytically calculate a statistic used for predicting the number of inactive units in a yet to be trained VAE net and show its practical applicability.

As GLM are based on exponential dispersion families (EDF), the analysis is based on the distribution assumption for the observation model. This is favourable as VAE, based on EDF, are applied in various different fields (with different distribution assumptions), as e.g.: anomaly detection using Gaussian distribution (see Xu et al. 2018), molecules representation using Bernoulli distribution (see Blaschke et al. 2018), image compression using Bernoulli distribution (see Duan et al. 2019) or multivariate spatial point processes using Poisson distribution (see Yuan et al. 2020).

This work is structured as follows: In Section 2, we give a motivation as well as an overview of related work. In Section 3, we present the theoretical background and our results that are a consequence of connecting VAE and GLM. Afterwards in Section 4, we provide simulations validating our theoretical results. In Section 5, we summarize our contributions and point out future research directions.

2. Motivation and Related Work

Our main contribution is to interpret the decoder of a β-VAE as a GLM (see Section 3.2). This will allow us to identify well-known activation functions as the reciprocal of link functions of GLM. Therefore, we are able to provide a systematic generalization of the loss analysis for VAE based on the assumption that the observation model belongs to an EDF (see Section 3.3).

Given an affine transformation for a part of the decoder, we derive MLE for an approximation to the β-VAE objective. Even though the decoder architecture is arguably simple, analysing the critical points and the loss landscapes of VAE helps to understand these models better. For example Lucas et al. (2019) consider this approach for their analysis with a Gaussian observation model.

Given the derived MLE, we derive weight and bias initializations (see Section 4.1 and Appendix B.3), analyse the auto-pruning of β-VAE (see Section 3.4) and analytically calculate a statistic used for predicting the number of inactive units (see Section 3.4 and Section 4.2).

By auto-pruning, we mean that during training the net sets nodes for the latent space inactive (i.e. to zero) and is presumably not able to activate them again due to local minima or saddle points in the loss surface. On the one hand, this property of β-VAE is desirable, as the model focusses on useful representations. On the other hand, it is considered a problem when too many units become inactive before learning a useful representation.

To weaken the effect of auto-pruning, different approaches are considered in the literature. Bowman et al. (2015), Sønderby et al. (2016) and Huang et al. (2018) use annealing of the parameter β during training. Our results below suggest that the annealing does not influence the final amount of active units, if training is conducted long enough and the loss surface is smooth. Lucas et al. (2019) make the same observation for the Gaussian observation model case.

Other approaches to tackle posterior collapse are to adjust the training objective. Kingma et al. (2016) propose an alternative objective, in which the gradient is ignored if the KL-Divergence is below a pre-determined threshold. In Razavi et al. (2019) a variational posterior is chosen so that the posterior collapse cannot happen by design. Hence, an implicit threshold is chosen. To reduce posterior collapse, Yeung et al. (2017) propose to use masking of groups in the latent dimension during training in the fashion of dropout layers. He et al. (2019) find empirically that the variational approximation lags behind the true model posterior in the initial stages of training. They propose to train the inference network separately to alleviate posterior collapse. Burda et al. (2015) consider importance weighting and show in their experiments that their proposed method yields less inactive units. For this, they calculate an activity statistic for each net after training. In this work, we provide a closed form for this statistic that can be calculated without training.

As our work focuses on analysing critical points of β-VAE, in the following, we give an overview of literature for analysing Autoencoders and VAE as well as on GLM used in the context of neural nets.

Various authors have analysed optimal points of the loss surface for squared loss (i.e. a Gaussian observation model) autoencoders. For autoencoders with linearised activations, Bourlard and Kamp (1988) show that the optimal solution is given by the solution of a singular value decomposition (SVD). Baldi and Hornik (1989) extend these results and analyse

the squared error loss of autoencoders for all critical points. Zhou and Liang (2018) provide analytical forms for critical points and characterize the values of the corresponding loss functions as well as properties of the loss landscape for one-hidden layer ReLU autoencoders. Kunin et al. (2019) consider regularizations on linear autoencoders and analyse the loss-landscape for different regularizations. They show that regularized linear autoencoders are capable of learning the principal directions and have connections to pPCA.

Variational Autoencoders with Gaussian observation models have been considered in Dai et al. (2018), Dai and Wipf (2019), Lucas et al. (2019) and Dai et al. (2020). Dai et al. (2018) analyse Gaussian VAE and show connections to pPCA and robust PCA as well as smoothing effects for local optima of the loss landscape. Dai and Wipf (2019) analyse deep Gaussian VAE. Assuming the existence of an invertible and differentiable mapping between low-rank manifolds in the sample space and the latent space, they show that spurious global optima exist, which do not reflect the data-generating manifold appropriately. Lucas et al. (2019) extend results of Dai et al. (2018) to analyse posterior collapse. They do this by analysing the difference from the true marginal to the ELBO which is possible under Gaussian assumptions, as $P_\theta(Z|X)$ becomes tractable in their setting. Furthermore, they provide experimental results on the posterior collapse for deep non-linear cases. For Gaussian observation models, Dai et al. (2020) introduce a taxonomy for different types of posterior collapse. Furthermore, they show that apart from the KL-Divergence, bad local minima, inherent in deep autoencoders, can lead to posterior collapse.

In this work, we use a linearisation of the decoder. Rolínek et al. (2019) analyse β-VAE and show that local orthogonality is promoted on the decoder. Kumar and Poole (2020) generalize the work of Rolínek et al. (2019) to different observation models and show that diagonal covariances of the variational distribution naturally encourages orthogonal columns of the Jacobian of the decoder. We extend their work as we provide an alternative formulation as well as critical points and error bounds for their approximation. Sicks et al. (2020) formulate a lower bound for the ELBO of a Bernoulli observation model using the linearisation of the decoder. They use the MLE to derive an initialization scheme and empirically compare it on synthetic data.

Wüthrich (2020) describes connections between GLM and neural network regression models, by interpreting the last layer of a neural net as GLM. With this, he is able to use a L^1 regularized neural net to learn representative features to improve a standard GLM. Furthermore, favourable properties (as for an actuarial context, the "balance property") are achieved by a proposed hybrid model.

3. Analysing the β-VAE objective

For realizations $x^{(1)}, \dots, x^{(N)}$ of a random variable (r.v.) X, we consider a β-VAE as in Higgins et al. (2017) with the objective $\mathcal{L}$, given by

$$\mathcal{L}(\phi, \theta) := \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{Z \sim q_\phi(\cdot|x^{(i)})} \left[\log P_\theta(x^{(i)}|Z) \right] - \beta D_{KL} \left(q_\phi(Z|x^{(i)}) || P_\theta(Z) \right), \tag{1}$$

where $\beta \geq 0$. Interpreting the expectation in this expression as autoencoder yields the encoder $q_\phi(Z|x^{(i)})$ and the decoder $P_\theta(x^{(i)}|Z)$. We make the usual assumptions (see Kingma

and Welling 2014) $P_\theta(Z) \sim \mathcal{N}(\vec{\mathbf{0}}, \boldsymbol{I})$ and $P_\theta(Z|X)$ is approximated by the recognition model with variational distribution

$$q_\phi(z|X) \sim \mathcal{N}(\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z).$$

We further assume for the encoder parameters $\boldsymbol{\mu}_z^{(i)} := f_1(x^{(i)}, \phi)$ and $0 \prec \boldsymbol{\Sigma}_z^{(i)} := \boldsymbol{S}_z^{(i)} \boldsymbol{S}_z^{(i)^T}$, with $\boldsymbol{S}_z^{(i)} := f_2(x^{(i)}, \phi)$, where f_1 and f_2 are arbitrary functions including affine transformations.

In the following, we first provide results for the special case of a Gaussian observation model. Then, we provide the theoretical background on EDF and show how the decoder can be interpreted as GLM. Finally, given this new perspective, we present an approximation to the objective in (1), MLE for this approximation and use these MLE to describe the auto-pruning of β-VAE.

3.1 Behaviour of the β-VAE objective for a Gaussian observation model

In this introductory section, we motivate the novel derivations for the EDF distribution families (see Section 3.3), by recapitulating known results for the Gaussian observation model case. We mainly follow the line of argument given in Kumar and Poole (2020).

Consider a Gaussian observation model with independent marginals $P_\theta(x|z_0) \sim \mathcal{N}\left(\boldsymbol{\vartheta}, \sigma^2 \boldsymbol{I}\right)$. The deterministic component[1] $\boldsymbol{\vartheta} : Z \to \mathbb{R}^d$ of the decoder maps to the location parameter (in this case the mean) of the observation model $\log P_\theta(x|z)$. To derive their "Gaussian Regularized Autoencoder" Kumar and Poole (2020) use a second order Taylor series expansion of the decoder $f_x(z) = \log P_\theta(x|z)$ in a $z_0 \in \mathbb{R}^\kappa$

$$f_x(z) \approx \log P_\theta(x|z_0) + J_{f_x}(z_0)(z - z_0) + \frac{1}{2}(z - z_0)^T H_{f_x}(z_0)(z - z_0), \tag{2}$$

where $J_{f_x} \in \mathbb{R}^{1\times\kappa}$ denotes the Jacobian and $H_{f_x} \in \mathbb{R}^{\kappa\times\kappa}$ the Hessian of f_x evaluated at z_0. The benefit of this approximation is that we can analytically calculate the expectation term (1). Given an analytically solvable KL-Divergence, the target of the ELBO becomes deterministic, which is beneficial for the analysis of β-VAE.

Kumar and Poole (2020) show that if piecewise linear activations are considered for the deterministic component[2] ϑ, we get

$$H_{f_x}(z) = J_\vartheta^T \left(\nabla_\vartheta^2 \log P_\theta(x|z)\right) J_\vartheta, \tag{3}$$

where $J_\vartheta \in \mathbb{R}^{d\times\kappa}$ is the Jacobian of $\boldsymbol{\vartheta}$. By considering piecewise linear functions, we allow for the decoder architecture to have an arbitrary amount of layers with prominent activations like the ReLU (see Nair and Hinton 2010) and alternations of this. Kumar and Poole (2020) choose $z_0^{(i)} = \mathbb{E}_{q_\phi\left(\cdot|x^{(i)}\right)}(Z)$ to remove the first order term in (2). This is a reasonable choice, but for the sake of later results, we will stick with general Taylor expansion points $z_0^{(i)} \in \mathbb{R}^\kappa$. Using (2) in (1), yields the deterministic objective

1. For notational reasons (see Section 3.2), we use $\boldsymbol{\vartheta}$ instead of the commonly used $\boldsymbol{\mu}$.
2. Kumar and Poole (2020) denote this component as g.

$$\mathcal{L}(\phi,\theta) \approx \widehat{\mathcal{L}}(\phi,\theta) := \frac{1}{N}\sum_{i=1}^{N} \log P_\theta(x|z_0^{(i)}) + J_{f_x}(z_0^{(i)})(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}) \tag{4}$$
$$+ \frac{1}{2}\mathrm{tr}\left(J_\vartheta^T(z_0^{(i)})\left(\nabla_\vartheta^2 \log P_\theta(x|z_0^{(i)})\right) J_\vartheta(z_0^{(i)})\boldsymbol{\Sigma}_z^{(i)}\right)$$
$$- \beta D_{KL}\left(q_\phi(Z|x^{(i)})||P_\theta(Z)\right).$$

Kumar and Poole (2020) argue that for the deterministic approximation to be accurate either higher central moments of the variational distribution or the higher order derivates $(\nabla_z^n \log P_\theta(x|z), n \geq 3)$ need to be small. Based on the EDF representation, we give bounds for this approximation error (see Corollary 1). For the Gaussian case, the approximation error vanishes, regardless of the choice for $\boldsymbol{\vartheta}$. Hence, the Taylor expansion point is arbitrary in this case.

Maximizing w.r.t. $\boldsymbol{\Sigma}_z^{(i)}$ and $\boldsymbol{\mu}_z^{(i)}$ yields

$$\hat{\boldsymbol{\Sigma}}_z^{(i)} = \left(I - \frac{1}{\beta} H_{f_x}(z_0^{(i)})\right)^{-1} \tag{5}$$

and

$$\hat{\boldsymbol{\mu}}_z^{(i)} = \frac{1}{\beta}\hat{\boldsymbol{\Sigma}}_z^{(i)}\left(J_{f_x}(z_0^{(i)})^T - H_{f_x}(z_0^{(i)}) \cdot z_0^{(i)}\right). \tag{6}$$

For now[3] considering the Gaussian case and $\boldsymbol{\vartheta}(z) = Wz + b$, with $W \in \mathbb{R}^{d\times\kappa}$ and $b \in \mathbb{R}^d$, we get by substituting the expressions (5) and (6)

$$\mathcal{L}(\theta,\hat{\phi}) = \frac{-1}{2N}\sum_{i=1}^{N}\left[\left(x^{(i)} - b\right)^T C^{-1}\left(x^{(i)} - b\right) + \beta \log|C| + d\log(2\pi\sigma^{2(1-\beta)})\right], \tag{7}$$

where $C := \left(\sigma^2 \boldsymbol{I} + \beta^{-1} W W^T\right)$. The derivation of this objective is the same as in the proof of Proposition 1, which can be found in Appendix A.2.

The objective in (7) is equivalent to the objective in (1) and reveals the connections of the VAE objective with $\beta = 1$ to probabilistic PCA. As stated in Dai et al. (2018), a solution for W and b can be derived analytically as given in Tipping and Bishop (1999). Solutions for the general EDF observation model can be found in Section 3.4.

In the next section, we introduce the EDF, to which the Gaussian distribution belongs, and GLM. Further, we state assumptions in order to generalize the approach from this section.

3.2 The EDF and the decoder of a VAE as GLM

Nelder and Wedderburn (1972) introduce GLM, providing a generalization of linear statistical models and thus of well-known statistical tools, such as analysis of variance (ANOVA), deviance statistics and MLE (see McCullagh and Nelder 1989). GLM consist of three parts: A random component X with a distribution belonging to the EDF, a systematic component given as affine mapping of features Z used to estimate $\mathbb{E}(X|Z)$, and a link function connecting these two components. The EDF is defined by the structure of the density.

3. The piecewise linear case is considered in section 3.3.

Definition 1 *We say the distribution of X given Z belongs to the exponential dispersion family (EDF), if the (conditional) density can be written as*

$$\log P_{\vartheta,\varphi}(X|Z) = \frac{X \cdot \vartheta(Z) - F(\vartheta(Z))}{\varphi} + K(X, \varphi), \tag{8}$$

where F and K are one-dimensional functions. $F : \mathbb{R} \to \mathbb{R}$ is called the log-normalizer. It ensures that integration w.r.t. the density in (8) over the support of X is equal to one. $\vartheta(Z) \in \Theta$ is the location parameter. Θ is an open, convex space with

$$\Theta = \left\{ \vartheta \in \mathbb{R} : \int_x \exp\left(\frac{x\vartheta}{\varphi} + K(x, \varphi) \right) dx < \infty \right\}.$$

$\varphi > 0$ is called the dispersion parameter and is independent of Z.

The EDF is a subset of the more general Exponential Family and differs by the fact that we can identify the dispersion parameter φ. Several well-known distributions, like the Gaussian, Bernoulli and Poisson distribution belong to this family. See Table 1 for the respective representations.

Table 1: An overview of well-known distributions that can be written as EDF distribution. The functions for the representation as exponential family member as well as ϑ and φ in terms of the natural parameters are displayed.

Dist. of X	$F(\vartheta)$	$K(x, \varphi)$	ϑ	φ
$Bin(n, p)$, with n fixed	$n \log(1 + \exp(\vartheta))$	$\log\binom{n}{x}$	$\log\left(\frac{p}{1-p}\right)$	1
$Bern(p)$ $= Bin(1, p)$	$\log(1 + \exp(\vartheta))$	0	$\log\left(\frac{p}{1-p}\right)$	1
$\mathcal{N}(\mu, \sigma^2)$, with σ^2 fixed	$\frac{\vartheta^2}{2}$	$-\frac{x^2}{2\varphi} - \frac{\log(2\pi\varphi)}{2}$	μ	σ^2
$Pois(\lambda)$	$\exp(\vartheta)$	$-\log(x!)$	$\log(\lambda)$	1

The EDF is studied in Barndorff-Nielsen (2014), Jorgensen (1986) and Jorgensen (1987). For an EDF distribution, the expectation as well as the variance can easily be computed. Further, the log-normalizer F provides explicit forms of the conditional expectation and variance and has further desirable properties, as can be seen in the following Lemma.

Lemma 1 *Let the distribution of a one-dimensional r.v. $X \sim P_{\vartheta,\varphi}(X|Z)$ given Z belong to the EDF. Then, it holds $\mathbb{E}(X|Z) = F'(\vartheta(Z))$ and $\mathbb{V}ar(X|Z) = \frac{1}{\varphi}F''(\vartheta(Z))$. Furthermore, the log-normalizer function F is convex and possesses all derivatives.*

The proof for the unconditional case is performed in Theorem 7.1, Corollary 7.1 and Theorem 8.1 in Barndorff-Nielsen (2014). The statement for the conditional case follows analogously.

We interpret the decoder $P_\theta(x^{(i)}|Z)$ as GLM. Therefore, we assume that the independent identical marginal distributions of X given Z belong to an EDF, where they share the same φ. With $Z \sim q_\phi\left(\cdot|x^{(i)}\right)$ from the encoder, the parameters of the decoder $P_\theta(x^{(i)}|Z)$ are given by $\theta = \{\boldsymbol{\vartheta}, \varphi\}$ and we have

$$\boldsymbol{\vartheta}(Z) = (\vartheta_1(Z), \ldots, \vartheta_d(Z))^T.$$

In order for the neural net implementation $\boldsymbol{m} \circ \boldsymbol{\vartheta} : \mathbb{R}^\kappa \to \mathbb{R}^d$ of a decoder to be reasonable for the log-likelihood part in (1), the decoder should approximate the expectation of $x^{(i)}$ given Z. According to Lemma 1, the last activation $\boldsymbol{m} : \boldsymbol{\vartheta}(Z) \to \mathbb{R}^d$ has to be $\boldsymbol{m} = F'$ (applied element wise) to get

$$\boldsymbol{m}(\boldsymbol{\vartheta}(Z)) = F'(\boldsymbol{\vartheta}(Z)) = \mathbb{E}_{\boldsymbol{\vartheta},\varphi}\left(x^{(i)}|Z\right). \tag{9}$$

We call the choice of $\boldsymbol{m}$ in (9) "canonical activation". This name originates from the "canonical link function". As mentioned before, for GLM a link function g connecting the systematic component of the model $\boldsymbol{\vartheta}(z)$ to the random component $\mathbb{E}_{\boldsymbol{\vartheta}}(X|z)$ is used. This function is called canonical if $g = (F')^{-1}$. Hence, the canonical activation is the inverse of the canonical link. In practice various different link functions, or in our case activations, are considered.

We want to emphasize that common neural net implementations depend on the choice of the last activation function to properly map to the natural parameters of the distribution, as the choice of loss function is strongly connected to this:

- If we use a Mean-Squared-Error loss and therefore implicitly[4] assume a Gaussian ($\mathcal{N}(\boldsymbol{\mu}, \sigma^2\boldsymbol{I})$) log-likelihood, the last activation has to be linear. In Section 3.1, we have implicitly assumed the last activation $\boldsymbol{m}$ to be the identity to ensure

$$\boldsymbol{m}(\boldsymbol{\vartheta}) = id(\boldsymbol{\vartheta}) = F'(\boldsymbol{\vartheta}) = \boldsymbol{\mu}.$$

- For the Binary Cross-Entropy loss, we implicitly assume a Bernoulli distribution $Bern(p)$. Hence, the last activation should be the sigmoid function to ensure

$$\boldsymbol{m}(\boldsymbol{\vartheta}) = \frac{1}{1 + \exp(-\boldsymbol{\vartheta})} = F'(\boldsymbol{\vartheta}) = p.$$

Actually, all activations that are equivalent to the choice in (9) up to a scalar $\rho \in \mathbb{R} \setminus \{0\}$, i.e.

$$\boldsymbol{m}(\boldsymbol{\vartheta}) = F'(\rho \cdot \boldsymbol{\vartheta}), \tag{10}$$

are legitimate choices. We call such activations "linearly canonical activation" and for canonical activations we have $\rho = 1$. The following example shows that the tanh activation can be used as a "linearly canonical activation".

4. Furthermore, σ^2 is implicitly set to $1/2$ which can result in unwanted consequences (see Section 3.4).

Example 1 (Bernoulli distribution - tanh activation) *Assume* $X \sim Bern\left(p\left(\boldsymbol{\vartheta}\right)\right)$ *and set the activation as*

$$\boldsymbol{m}(\boldsymbol{\vartheta}) = 1/2 \cdot \tanh(\boldsymbol{\vartheta}) + 1/2.$$

As in the example before, $F(\boldsymbol{\vartheta}) = \log\left(1 + \exp\left(\boldsymbol{\vartheta}\right)\right)$ *and it can be shown that*

$$\boldsymbol{m}(\boldsymbol{\vartheta}) = F'(2 \cdot \boldsymbol{\vartheta}).$$

Our theory presented in this paper applies for any linearly canonical activation. As we consider piecewise linear functions for $\boldsymbol{\vartheta}$, we can substitute $\hat{\boldsymbol{\vartheta}} := \rho \cdot \boldsymbol{\vartheta}$ and calculate

$$\boldsymbol{m}(\boldsymbol{\vartheta}(Z)) = F'(\hat{\boldsymbol{\vartheta}}(Z)) = \mathbb{E}_{\hat{\boldsymbol{\vartheta}},\varphi}\left(x^{(i)}|Z\right).$$

Therefore, for notational ease we will stick with the canonical activations. During our simulations, settings with either sigmoid or tanh activation were indistinguishable.

3.3 Local Behaviour of the β-VAE objective for EDF observation models

In this section, we derive an approximation to the β-VAE objective in (1) similar to the way in Section 3.1, but for a more general case by considering distributions from the EDF.

In their work, Kumar and Poole (2020) consider distributions with finite first and second moments, which is even more general. Using the more restrictive class of EDF distributions, we provide an error characterization for the approximation in (2). Further, given Proposition 1, we derive MLE for the affine decoder case (see Section 3.4). Given these, we produce weight and bias initializations (see Section 4.1 and Appendix B.3) and analyse the auto-pruning of β-VAE (see Section 3.4 and 4.2).

Apart from the assumptions in Section 3, for the Taylor Series expansion based on the decoder $\boldsymbol{m} \circ \boldsymbol{\vartheta}$ as in Section 3.1, we further assume

- $\boldsymbol{\vartheta}$ to be piecewise linear,
- a canonical activation function $\boldsymbol{m}$ and
- the Taylor expansion points $z_0^{(i)}$ from (2) belong to the null space (kernel) of $\boldsymbol{\vartheta}$: $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$.

Proposition 1 *Assume that the independent identical marginals of* X *given* Z *belong to the same EDF distribution with functions* F *and* K *as in* (8)*. Under the assumptions stated in the beginning of this section, there exists an approximative representation for the VAE objective in* (1)*,*

$$\mathcal{L}(\theta, \phi) \approx \widehat{\mathcal{L}}(\theta, \phi), \tag{11}$$

that admits optimal solutions $\hat{\phi} = \left\{\hat{\boldsymbol{\mu}}_z^{(i)}, \hat{\boldsymbol{\Sigma}}_z^{(i)}; i = 1, \ldots, N\right\}$ *(given in* (5) *and* (6)*), such that it can be written as*

$$\widehat{\mathcal{L}}(\theta) := \widehat{\mathcal{L}}(\theta, \hat{\phi}) = \frac{-1}{2N} \sum_{i=1}^{N} \Bigg[\left(F''(0)^{-1}(x^{(i)} - F'(0)) + J_\vartheta(z_0^{(i)}) z_0^{(i)}\right)^T C(z_0^{(i)})^{-1} \left(F''(0)^{-1}(x^{(i)} - F'(0)) + J_\vartheta(z_0^{(i)}) z_0^{(i)}\right) + \beta \log \left| C(z_0^{(i)}) \right| + \beta \cdot d \log \left(\varphi^{-1} F''(0)\right) + D(\varphi) \Bigg], \tag{12}$$

where $C(z_0^{(i)}) := F''(0)^{-1}\varphi \boldsymbol{I}_d + \beta^{-1} J_\vartheta(z_0^{(i)}) J_\vartheta(z_0^{(i)})^T$ *and* $J_\vartheta(z_0^{(i)}) \in \mathbb{R}^{d \times \kappa}$ *is the Jacobian of* $\boldsymbol{\vartheta}$*. The definition of* $D(\varphi)$ *can be found in equation* (34) *of the appendix.*

The proof can be found in Appendix A.2.

By choosing a common Taylor expansion point $z_0^{(i)} = z_0$ for all $i = 1, \ldots, N$ (i.e. for all observations), Proposition 1 shows that the local approximation of the β-VAE objective for different EDF admits a pPCA fashioned representation. Furthermore, this representation belongs to the matrix perspective functions class in Won (2020), which can be optimized using proximity operators.

See Table 2 for different EDF distribution associated parameters. Unfortunately, this approximation is not possible for all EDF distributions. As an example the canonical activation of the Gamma distribution (which also belongs to the EDF) is given by $-1/x$, with support in $\mathbb{R}^-$. Hence, we cannot choose $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$.

Table 2: EDF distribution associated parameters in Proposition 1.

Dist. of $X\|Z$	$F(0)$	$F'(0)$	$F''(0)$	$\beta \cdot d \log\left(\varphi^{-1} F''(0)\right) + D(\varphi)$
Bern(p)	$\log(2)$	$1/2$	$1/4$	$(1-\beta)\, d \log(4) - 4N^{-1} \sum_{i=1}^{N} \left\| x^{(i)} - 1/2 \right\|_2^2$
$\mathcal{N}(\mu, \sigma^2)$	0	0	1	$d \log(2\pi\sigma^{2(1-\beta)})$
$Pois(\lambda)$	1	1	1	$2d + N^{-1} \sum_{i=1}^{N} \left[- \left\| x^{(i)} - 1 \right\|_2^2 + 2 \log\left(\prod_{j=1}^{d} x_j^{(i)}!\right)\right]$

In the following Corollary, we quantify the introduced Taylor remainder in (2) for different distributions.

Corollary 1 *Let the assumptions of Proposition 1 be given. We introduce the remainder of a second order Taylor term* $T(z; z_0^{(i)})$ *in* (2)*, by*

$$f_x(z) - T(z; z_0^{(i)}) = R_2(z; z_0^{(i)}).$$

- *For a Gaussian observation model, we have* $R_2(z; z_0^{(i)}) = 0 \quad \forall z_0^{(i)} \in Z$ *and hence in* (11)

$$\mathcal{L}(\theta, \phi) = \widehat{\mathcal{L}}(\theta, \phi).$$

- *For a Binomial observation model, we obtain*

$$R_2(z; z_0^{(i)}) = \frac{n}{8 \cdot 4!} \left\| J_\vartheta(\xi)(z - z_0^{(i)}) \right\|_4^4 \cdot M,$$

with $M \in \left[\frac{-1}{3}, 1\right]$ *and* $\xi = z_0^{(i)} + c\left(z - z_0^{(i)}\right)$*, where* $c \in [0,1]$*. Further, if we assume* $\boldsymbol{\vartheta}$ *to be affine on the convex set spanned by* z *and* $z_0^{(i)}$*, we have* $M \in [0,1]$ *and hence*

$$\mathcal{L}(\theta, \phi) \geq \widehat{\mathcal{L}}(\theta, \phi). \tag{13}$$

- *For a Poisson observation model, if we assume* $\boldsymbol{\vartheta}$ *to be affine on the convex set spanned by* z *and* $z_0^{(i)}$*, it can be shown that we have*

$$\sum_{j=1}^{d} -\boldsymbol{\vartheta}_j(z)^3 \cdot \exp(\boldsymbol{\vartheta}_j(z))/6 \leq R_2(z, z_0^{(i)}) \leq \sum_{j=1}^{d} -\boldsymbol{\vartheta}_j(z)^3/6.$$

See Appendix A.3 for the proof.

If we choose $\beta = 1$, the objective in (1) becomes the ELBO. For $\boldsymbol{\vartheta}(z) = Wz + b$, with $W \in \mathbb{R}^{d \times \kappa}$ and $b \in \mathbb{R}^d$, Corollary 1 highlights how our theory generalizes the works of Dai et al. (2018), Lucas et al. (2019) and Sicks et al. (2020). Under the Gaussian assumption $\widehat{\mathcal{L}}$ is exact. Then, Proposition 1 yields the objective in (7) also given by Dai et al. (2018) and analysed by Lucas et al. (2019). For the Bernoulli distribution, according to (13) we approximate the ELBO in (1) from below. The result is the same lower bound as reported in Sicks et al. (2020). Thus, the ELBO is bounded from both sides as naturally its values have to be smaller than zero. Further, as we will see in the simulations, the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})] := \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{q_{\hat{\phi}}}[R_2(z^{(i)}; z_0^{(i)})]$ evaluated at $M = 1$ and $\theta = \hat{\theta}$ serves as an indicator for us on what to expect from training.

3.4 MLE for the affine transformation case

In this section, we derive analytical solutions for the objective in (12), when the location parameter is given by an affine transformation $\boldsymbol{\vartheta}(z) = Wz + b$. We analyse the optimal values of $W \in \mathbb{R}^{d \times \kappa}$ and $b \in \mathbb{R}^d$ and highlight interesting implications.

First, we rewrite the objective in (12). With $\boldsymbol{\vartheta}(z) = Wz + b$, we obtain a similar representation as Tipping and Bishop (1999) for pPCA, given by

$$\widehat{\mathcal{L}}(W, b) = \frac{-1}{2}\left(\operatorname{tr}\left(C^{-1}S\right) + \beta \log|C| + \beta \cdot d \log\left(\varphi^{-1}F''(0)\right) + D\left(\varphi\right)\right), \tag{14}$$

where $C := \left(F''(0)^{-1}\varphi I_d + \beta^{-1}WW^T\right)$ and

$$S := \frac{1}{N}\sum_{i=1}^{N}\left(F''(0)^{-1}\left(x^{(i)} - F'(0)\right) - b\right)\left(F''(0)^{-1}\left(x^{(i)} - F'(0)\right) - b\right)^T.$$

According to Tipping and Bishop (1999), the MLE for $\hat{b}$ is given by the sample mean

$$\hat{b} = \frac{1}{N}\sum_{i=1}^{N} F''(0)^{-1}\left(x^{(i)} - F'(0)\right). \tag{15}$$

Therefore, S with $\hat{b}$ becomes the sample covariance, which we denote as $\hat{S}$. With $\lambda_1, \ldots, \lambda_d$ we denote the (ordered) eigenvalues of the matrix $\hat{S}$ and in a similar way to Tipping and Bishop (1999), for $F''(0) \leq 1$, we can derive the MLE of W as

$$\hat{W} = U_\kappa \left(K_\kappa - \beta F''(0)^{-1} \varphi I_\kappa \right)^{1/2} R =: U_\kappa L R, \tag{16}$$

where $U_\kappa \in \mathbb{R}^{d \times \kappa}$ is composed of κ eigenvectors of the matrix $\hat{S}$. The eigenvectors are associated with the κ biggest eigenvalues $\lambda_1, \ldots, \lambda_\kappa$. $K_\kappa \in \mathbb{R}^{\kappa \times \kappa}$ is a diagonal matrix with entries

$$k_j = \begin{cases} \lambda_j, & \lambda_j \geq \beta F''(0)^{-1} \varphi \\ \beta F''(0)^{-1} \varphi, & \text{else.} \end{cases} \tag{17}$$

$R \in \mathbb{R}^{\kappa \times \kappa}$ is an arbitrary rotation matrix, which implies that our optimal solution is invariant to rotations. Dai et al. (2018) show this as well as invariance to permutations in their Theorem 2.

Further for the Gaussian case, the MLE for $\varphi = \sigma^2$ is given by

$$\hat{\sigma}^2 = \frac{1}{(d - \beta\kappa)} \sum_{i=\kappa+1}^{d} \lambda_i, \tag{18}$$

which can be interpreted as the variance lost due to the dimension reduction by the autoencoder. This expression is only well-defined for $\beta \in [0, d/\kappa)$ and we have $\hat{\sigma}^2 > 0$ if $rank(S) > \kappa$. Further, the estimator $\hat{\sigma}^2$ is increasing in β. Hence the VAE performs optimal in view of reconstruction (has the lowest variance lost), when $\beta = 0$.

This observation agrees with the definition of the objective (1): lower β emphasize the reconstruction part. As pointed out by an anonymous reviewer, our observation is in line with the analysis by Alemi et al. (2018) as well as Rezende and Viola (2018). While the results therein are originated from a different point of view, the interpretations on β are consistent.

It is possible to have $\text{rank}(\hat{W}) < \kappa$ as the "cut-off" term

$$\beta F''(0)^{-1} \varphi \tag{19}$$

controls how much columns in the matrix $\hat{W} R^T$ are zero. We interpret this as a consequence of the auto-pruning property of VAE. If the data signal is not strong enough, it is pruned away. In common VAE implementations σ^2 is often implicitly assumed to be equal to $1/2$ (i.e. when using MSE loss without any scaling). Lucas et al. (2019) show how the stability of the estimator for W is influenced by the choice of σ^2. If σ^2 (and hence the cut-off value) is chosen too large, principal components cannot be captured by the model. We agree with their conclusion that learning σ^2 is necessary for gaining a full latent representation.

For the parameter estimates of the variational distribution, we get

$$\hat{\boldsymbol{\Sigma}}_z = \frac{\beta \varphi}{F''(0)} R^T K_\kappa^{-1} R \tag{20}$$

and

$$\hat{\boldsymbol{\mu}}_z^{(i)} = \frac{1}{\beta \varphi} \hat{\boldsymbol{\Sigma}}_z \hat{W}^T \left(x^{(i)} - \bar{x} \right) = R^T K_\kappa^{-1} L U_\kappa^T \frac{1}{F''(0)} \left(x^{(i)} - \bar{x} \right). \tag{21}$$

When a diagonal covariance structure is imposed, the decoder Jacobian columns are forced to be orthogonal. In (20) a diagonal covariance matrix means $R = I_\kappa$. As a result, we have orthogonal columns in the matrix $\hat{W}$. This result supports the findings of Kumar and Poole (2020). They show an implicit regularization in the local behaviour of the VAE objective (1) for a diagonal covariance assumption, without presenting analytical solutions as the one in (20).

Next, we analyse how the parameter β influences the optimal variational parameters and as a consequence the auto-pruning of β-VAE.

- For β high enough, we get $\hat{W} = \mathbf{0}$ and $\mathbf{\Sigma}_z = \boldsymbol{I}_\kappa$ and hence $\boldsymbol{\mu}_z^{(i)} = 0$ independent of the input $x^{(i)}$. Therefore, the Kullback-Leibler Divergence part in (1) is amplified enough such that the variational distribution generates independent noise. The posterior collapses.

- For smaller β values, more and more eigenvalue dimensions covered by U_κ^T are used and scaled appropriately with $K_\kappa^{-1}L$. Therefore, in $\boldsymbol{\mu}_z^{(i)}$ the inputs $F''(0)^{-1}\left(x^{(i)} - \bar{x}\right)$ are transformed better and better to the latent space to guarantee a proper reconstruction.

We can further analytically compute statistics used to detect active latent dimensions in β-VAE. Burda et al. (2015) propose the statistic $A_{z_j} = Cov_x\left(\mathbb{E}_{q_\phi(z_j|x)}[z_j]\right)$. They define the dimension z_j to be active if $A_{z_j} > 0.01$. Using the sample covariance to approximate this value with the given data points and using (21), we get

$$A_{z_j} \approx \frac{\left(k_j - \beta F''(0)^{-1}\varphi\right)\lambda_j}{k_j^2} = \begin{cases} \dfrac{\left(\lambda_j - \beta F''(0)^{-1}\varphi\right)}{\lambda_j}, & \lambda_j \geq \beta F''(0)^{-1}\varphi \\ 0, & \text{else.} \end{cases} \tag{22}$$

So the value is either 0 or equals the (positive) relative distance of the eigenvalue λ_j to the cut-off value $\beta F''(0)^{-1}\varphi$. The effect on how β controls the activity of the latent space dimensions becomes apparent. The bigger β the less latent dimensions remain non-zero.

This result yields the ineffectiveness of annealing the β parameter during training. If training is conducted long enough and the loss surface is smooth enough, the MLE will be achieved by optimization. Hence, the active units are determined by (22) for the last β value during annealing.

4. Simulation results

In this section, we provide simulation results to illustrate our theoretical results from Section 3. We consider two applications.

1. We show that the use of the MLE derived in section 3 as initialization for VAE implementations yields a faster training convergence.

2. We compare the analytical calculations of the activity statistics in (22) with the resulting activities of β-VAE implementations. The analytical values serve as good indicator on how much latent dimensions become inactive during training.

4.1 MLE-based initialization

We focus on the Bernoulli case, popular for image data and set $\beta = 1$. According to Corollary 1, $\widehat{\mathcal{L}} = \widehat{\mathcal{L}}(\theta)$ from Proposition 1 becomes a lower bound yielding (13). Therefore, we expect the ELBO of VAE with an according architecture to lie above $\widehat{\mathcal{L}}$. The essential messages of the simulations are the following:

- It is reasonable to use $\widehat{\mathcal{L}}$ to analyse the training performance on real life data sets.
- The statement above is also valid for ReLU-Net decoders.
- The MLE points, from Section 3.4, used as initialization enhance the training performance.

For training of the nets, we use the Adam optimizer by Kingma and Ba (2015) with learning rate 0.0001 and a batch size of 100. Training was done for a total of $25,000$ batch evaluations. The simulations ran on a dual Intel Xeon E5-2670 with 16 CPU @ 2.6 GHz. The longest setup took about one hour of computing time.

By varying the following hyper parameters, we conduct a total of 18 different simulation setups:

- Architecture: "Affine" or "ReLU-Net" decoder.
- Latent dimension κ: 2, 5 or 20.
- Data: "synthetic", "frey" or "mnist".

We compare our initialization scheme ("MLE-B") to a benchmark ("Bench") given by He et al. (2015). The initialization schemes and different hyper parameters are explained in detail in Appendix B. Figure 1 shows the result of training the two different initialized VAE on the frey data set with $\kappa = 2$. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

In Figure 1, the bound $\widehat{\mathcal{L}}$ is reasonable and both architectures do not perform significantly better. The results of all simulation schemes can be found in Appendix B.6. Comparing these simulation results and considering Figure 1, we observe the following:

- For the affine decoder architecture, the initialization MLE-B converges directly, whereas the Benchmark takes much more time. The end values are comparable. For the ReLU-Net decoder architecture, the performance of the two initialization methods mostly shows a small initial advantage of MLE-B which, however, is not as clear as for the affine architecture.
- In no simulation setup a net was over-fitting, not even for large values of κ with synthetic data, where a much smaller κ is needed. This is a consequence of the auto-pruning.
- For the MLE values, based on Corollary 1 we know that $\mathcal{L}(\hat{\theta}, \hat{\phi})$ lies above $\widehat{\mathcal{L}}(\hat{\theta}, \hat{\phi})$. It seems that MLE-B needs a very short burn-in period to perform according to Corollary 1. We believe that the offset at the beginning originates from not readily initialized hidden layers (mainly for the encoder). We can observe a performance

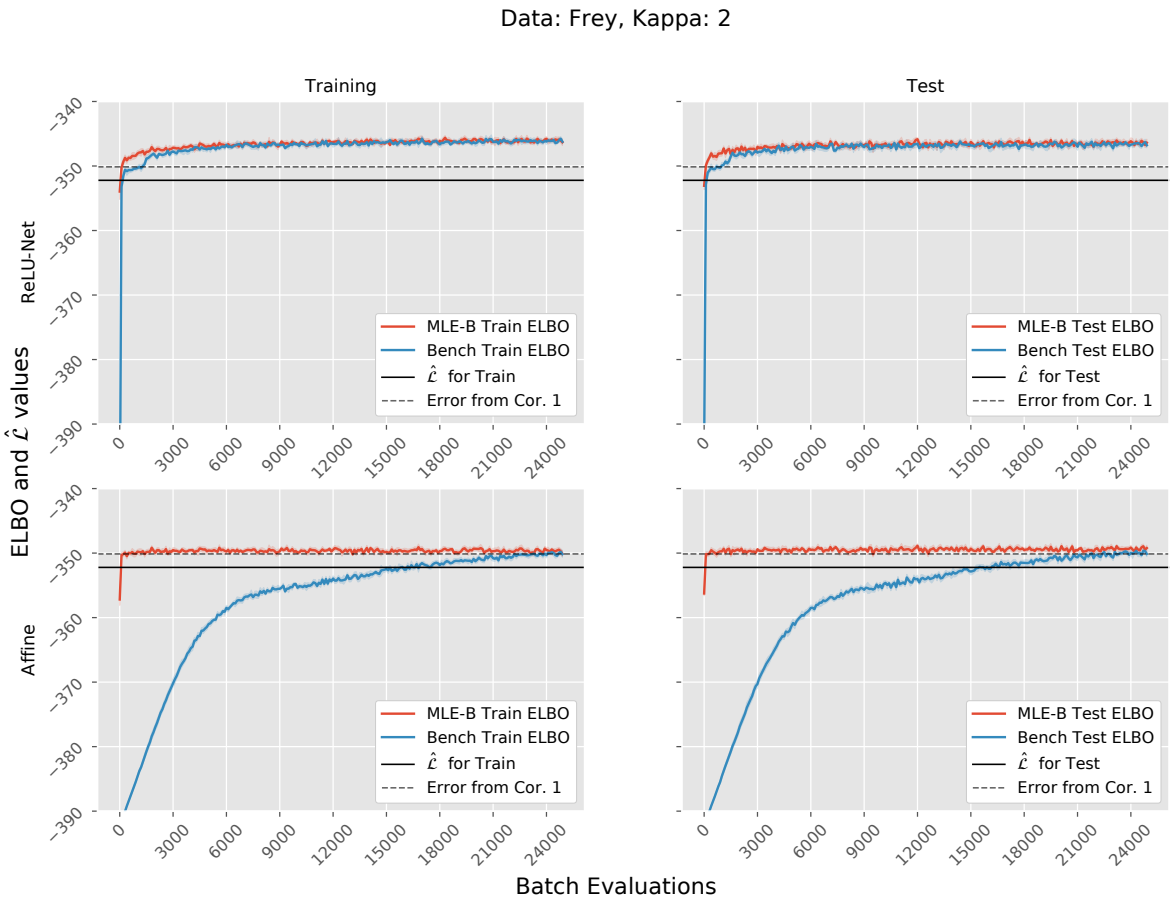

Figure 1: The figure shows two different setups ReLU-Net and Affine with frey data, $\kappa = 2$ and sigmoid activation. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

above the expected error. Possible reasons for this are that the sampled values during optimization differ from the analytically calculated expectation and different realized values for θ and ϕ.

4.2 Latent dimension activities

In this section, we show that the analytical activities in (22) serve as a good indicator for the amount of active nodes without conducting training. This statement also holds for ReLU-Net decoder and not just for the affine case. We consider the mnist data set. For the Gaussian observation model, Figure 2 shows histograms of the activity statistics A_{z_j}, proposed by Burda et al. (2015), for the analytical case and an Affine / ReLU-Net decoder (Details on the architectures can be found in Appendix B.1) after the training. Displayed

are the values for different β. Table 3 displays the calculated distances to the analytical calculations based on 10 simulations.

The corresponding figure and table for the Bernoulli observation model can be found in Appendix B.5.

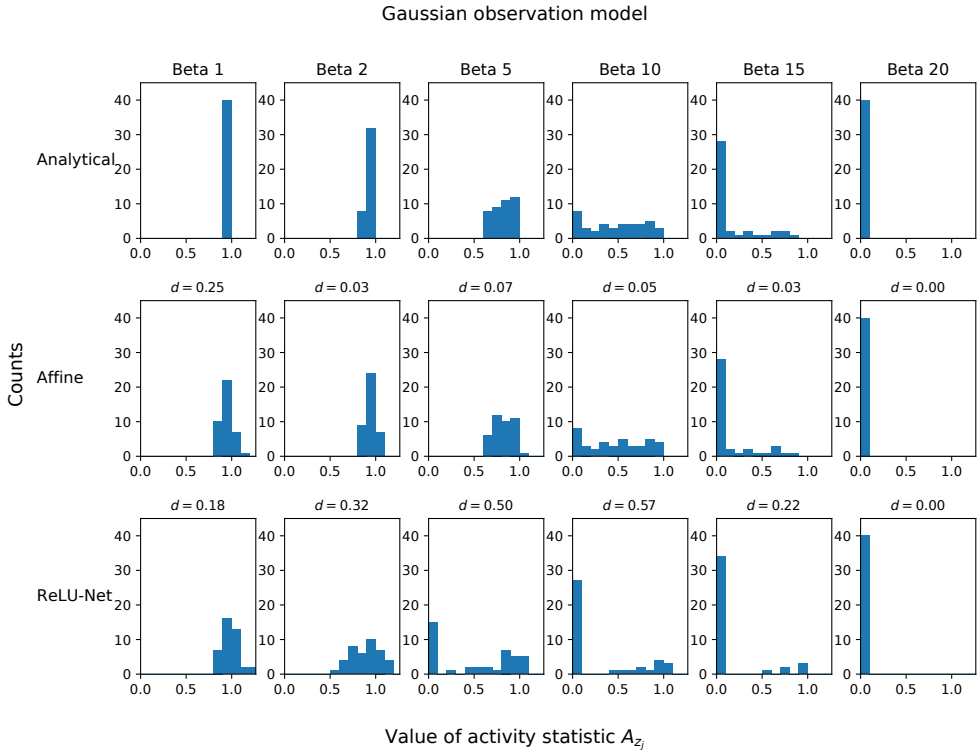

Figure 2: The figure shows histograms of the activities for 40 latent dimensions for our analytical calculation and an Affine/ ReLU-Net decoder (as described in appendix B.1) after training. We have considered the mnist data set with a Gaussian observation model. Above each Affine and ReLU-Net histogram plot, we show the distance ($\in [0, 1]$, lower is better) as defined in (38) to the analytical histogram.

Table 3: We display the distances ($\in [0, 1]$, lower is better) of the histograms to the corresponding analytical calculation for the Gaussian observation model. Displayed are the results of 10 simulations as "mean±std".

	Beta 1	Beta 2	Beta 5	Beta 10	Beta 15	Beta 20
Affine	0.24 (±0.06)	0.04 (±0.02)	0.04 (±0.02)	0.05 (±0.01)	0.03 (±0.02)	0 (±0)
ReLU	0.18 (±0.03)	0.3 (±0.02)	0.49 (±0.04)	0.55 (±0.03)	0.23 (±0.01)	0 (±0)

Since the analytical calculations are based on the affine decoder architecture, the first two rows look similar. Given a value of β we can make trustworthy predictions how much latent dimensions become inactive during training.

The ReLU-Net decoder behaves differently. It seems to be that the deeper structure and piecewise linear functions allow the model to use less latent dimensions to properly model the data distribution and hence more latent dimensions can become inactive. Given this point of view, we can use the analytical calculation of the statistics as a lower estimate of how much latent dimensions will turn out to be inactive after training.

Since the analytical calculation of the statistic in (22) is low cost, we recommend to use it in either case.

5. Conclusion

We have established a new framework for β-VAE, by interpreting the decoder of a β-VAE as a GLM. Given this framework, we derive and analyse an approximation to the β-VAE objective based on the EDF observation model.

We derive MLE for this approximation in the affine transformation setting. Furthermore, we present simulation results validating the theory on real world data sets, like the frey and mnist data set. The results here generalize previous work in this field.

Further, we provide an analytical description of the auto-pruning of β-VAE . We show that the parameter MLEs are directly influenced by the cut-off term in (19), which yields the dependence on the parameter β for the affine decoder setting. Furthermore, the amount of active units is directly affected by this term. Our simulation results suggest that the implications can be used for ReLU-Net decoders.

A possible extension is to integrate distributions like the Gamma distribution which belongs to the EDF.

Acknowledgments

We would like to thank the editor, Shakir Mohamed, as well as the anonymous reviewers for their thoughtful assessment and comments which helped us to improve the quality of the paper. Robert Sicks gratefully acknowledges the financial support via a PhD grant from the Fraunhofer Institute for Industrial Mathematics ITWM.

Appendix A. Appendix: Proofs

A.1 Auxiliary results

Lemma 2 *Let $B, \Gamma \in \mathbb{R}^{\kappa \times \kappa}$ be symmetric positive definite matrices. Then it holds*

$$B = \arg\min_{\Gamma \succ 0} \operatorname{tr}(B\Gamma^{-1}) + \log|\Gamma|$$

and hence

$$\kappa + \log|B| = \min_{\Gamma \succ 0} \operatorname{tr}(B\Gamma^{-1}) + \log|\Gamma|.$$

Proof Define the two distributions $\mathcal{N}_0(\boldsymbol{\mu}, B)$ and $\mathcal{N}_1(\boldsymbol{\mu}, \Gamma)$. We have

$$\begin{aligned} &2 \cdot D_{KL}\left(\mathcal{N}_0(\boldsymbol{\mu}, B) || \mathcal{N}_1(\boldsymbol{\mu}, \Gamma)\right) \\ &= \operatorname{tr}(B\Gamma^{-1}) + \log|\Gamma| - \kappa - \log|B|. \end{aligned} \tag{23}$$

Now, consider that for the Kullback-Leibler-Divergence with probability distributions P and Q it holds:

- $D_{KL}(P||Q) \geq 0$ for all inputs.
- $D_{KL}(P||Q) = 0$ if and only if $P = Q$ almost everywhere.

Hence, we conclude $B = \Gamma$ in the minimum. ■

A.2 Proof of Proposition 1

Proof To proof Proposition 1, we change the perspective. Instead of maximizing, we want to minimize the negative expression given by

$$\begin{aligned} -\mathcal{L}(\phi,\theta) :=& \frac{1}{N}\sum_{i=1}^{N} \beta \cdot D_{KL}\left(q_\phi(Z|x^{(i)})||P(Z)\right) \\ & - \mathbb{E}_{Z\sim q_\phi(\cdot|x^{(i)})}\left[\log P_{\vartheta,\varphi}(x^{(i)}|Z)\right]. \end{aligned} \tag{24}$$

Looking at (24), we see two terms. For the KL-Divergence we have that

$$\begin{aligned} & 2 \cdot D_{KL}\left(q_\phi(Z|x^{(i)})||P(Z)\right) \\ & = \operatorname{tr}[\boldsymbol{\Sigma}_z^{(i)}] - \log|\boldsymbol{\Sigma}_z^{(i)}| + ||\boldsymbol{\mu}_z^{(i)}||_2^2 - \kappa \end{aligned} \tag{25}$$

and for the second term (with q_ϕ as abbreviation for $q_\phi(\cdot|x^{(i)})$) we get with a second-order Taylor Expansion as in (2) and

$$\mathbb{E}\left[XX^T\right] = \operatorname{Cov}(X) + \boldsymbol{\mu}\boldsymbol{\mu}^T$$

that

$$\begin{aligned} & -\mathbb{E}_{q_\phi}\left[\log P_{\vartheta,\varphi}(x^{(i)}|Z)\right] \\ & \approx -\log P_\theta(x|z_0^{(i)}) - \mathbb{E}_{q_\phi}\left[J_{f_x}(z_0^{(i)})(z-z_0^{(i)}) + \frac{1}{2}(z-z_0^{(i)})^T H_{f_x}(z_0^{(i)})(z-z_0^{(i)})\right] \\ & = -\log P_\theta(x|z_0^{(i)}) - J_{f_x}(z_0^{(i)})(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}) \\ & - \frac{1}{2}\operatorname{tr}\left(H_{f_x}(z_0^{(i)})\boldsymbol{\Sigma}_z^{(i)}\right) - \frac{1}{2}\operatorname{tr}\left(H_{f_x}(z_0^{(i)})\left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)\left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)^T\right). \end{aligned} \tag{26}$$

Putting the two terms together, for our target function (24), it follows that it is approximated by

$$\begin{aligned} -\widehat{\mathcal{L}}(\phi,\theta) :=& \frac{1}{N}\sum_{i=1}^{N}\left[\frac{\beta\operatorname{tr}[\boldsymbol{\Sigma}_z^{(i)}]}{2} - \frac{\beta\log|\boldsymbol{\Sigma}_z^{(i)}|}{2} + \frac{\beta||\boldsymbol{\mu}_z^{(i)}||_2^2}{2} - \frac{\beta\kappa}{2} - \frac{1}{2}\operatorname{tr}\left(H_{f_x}(z_0^{(i)})\boldsymbol{\Sigma}_z^{(i)}\right)\right. \\ & - \log P_\theta(x|z_0^{(i)}) - J_{f_x}(z_0^{(i)})(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}) \\ & \left. - \frac{1}{2}\operatorname{tr}\left(H_{f_x}(z_0^{(i)})\left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)\left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)^T\right)\right]. \end{aligned}$$

All potential minima w.r.t. $\mathbf{\Sigma}_z^{(i)}$ have to conform to

$$\hat{\mathbf{\Sigma}}_z^{(i)} = \left(I_\kappa - \frac{1}{\beta} H_{f_x}(z_0^{(i)})\right)^{-1}, \tag{27}$$

independent of $x^{(i)}$. Note that this expression is well-defined as per assumption, we have $\beta > 0$ and further it can be shown that $-H_{f_x}(z_0^{(i)})$ is positive semi definite with X having an EDF distribution. To see that these are minima and not maxima, consider Lemma 2,

$$\kappa + \log|AA^T| = \min_{\Gamma \succ 0} \operatorname{tr}(AA^T\Gamma^{-1}) + \log|\Gamma|,$$

where we set $\Gamma^{-1} = \hat{\mathbf{\Sigma}}_z^{(i)}$ and $AA^T = I_\kappa - \frac{1}{\beta} H_{f_x}(z_0^{(i)})$.

Given the fact, that $\boldsymbol{\vartheta}$ is a piecewise linear function, we have

$$H_{f_x}(z) = J_\vartheta(z_0^{(i)})^T \left(\nabla_\vartheta^2 \log P_\theta\left(x|z\right)\right) J_\vartheta(z_0^{(i)}), \tag{28}$$

as in (3). Further, since we assume a $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$, it can be shown that $\nabla_\vartheta^2 \log P_\theta\left(x|z\right)$ is independent of the explicit choice of $z_0^{(i)}$, so we can use

$$HH^T = \Gamma := -\left(\nabla_\vartheta^2 \log P_\theta\left(x|z_0^{(i)}\right)\right), \tag{29}$$

which is positive definite when the distribution of X given Z belongs to the EDF. If follows

$$\hat{\mathbf{\Sigma}}_z^{(i)} = \left(I_\kappa + \frac{1}{\beta} J_\vartheta(z_0^{(i)})^T \Gamma J_\vartheta(z_0^{(i)})\right)^{-1}. \tag{30}$$

Given $\hat{\mathbf{\Sigma}}_z^{(i)}$ the minimal target function evaluated at this point becomes

$$\begin{aligned} -\widehat{\mathcal{L}}(\phi \setminus \{\mathbf{\Sigma}_z\}, \theta) =& \frac{1}{N}\sum_{i=1}^{N}\Bigg[\frac{\beta||\boldsymbol{\mu}_z^{(i)}||_2^2}{2} - \log P_\theta(x|z_0^{(i)}) - J_{f_x}(z_0^{(i)})(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}) \\ &- \frac{1}{2}\operatorname{tr}\left(H_{f_x}(z_0^{(i)}) \cdot \left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)\left(\boldsymbol{\mu}_z^{(i)} - z_0^{(i)}\right)^T\right) \\ &+ \frac{1}{2}\log\left|\hat{\mathbf{\Sigma}}_z^{(i)^{-1}}\right|\Bigg]. \end{aligned}$$

For $\boldsymbol{\mu}_z^{(i)}$, we get as minimal points

$$\hat{\boldsymbol{\mu}}_z^{(i)} = \frac{1}{\beta}\hat{\mathbf{\Sigma}}_z^{(i)}\left(J_{f_x}(z_0^{(i)})^T - H_{f_x}(z_0^{(i)}) \cdot z_0^{(i)}\right). \tag{31}$$

The candidates for an optimal $\hat{\boldsymbol{\mu}}_z^{(i)}$ are minima since the second derivative is a positive constant times $\hat{\mathbf{\Sigma}}_z^{(i)^{-1}}$, which is positive definite. Given the optimal $\boldsymbol{\mu}_z^{(i)}$ and $\mathbf{\Sigma}_z^{(i)}$ our target function is only dependent on the parameters θ. We get

$$\begin{aligned} -\widehat{\mathcal{L}}(\theta) =& \frac{1}{N}\sum_{i=1}^{N}\Bigg[\left(J_{f_x}(z_0^{(i)})^T - H_{f_x}(z_0^{(i)}) \cdot z_0^{(i)}\right)^T E\left(J_{f_x}(z_0^{(i)})^T - H_{f_x}(z_0^{(i)}) \cdot z_0^{(i)}\right) \\ &- \log P_\theta(x|z_0^{(i)}) - \frac{1}{2} z_0^{(i)^T} H_{f_x}(z_0^{(i)}) z_0^{(i)} + J_{fx}(z_0^{(i)}) z_0^{(i)} + \frac{\beta}{2}\log\left|\hat{\mathbf{\Sigma}}_z^{(i)^{-1}}\right|\Bigg], \end{aligned} \tag{32}$$

where $E := \frac{1}{2\beta}\hat{\mathbf{\Sigma}}_z^{(i)^2} - \frac{1}{2\beta^2}\hat{\mathbf{\Sigma}}_z^{(i)} H_{f_x}(z_0^{(i)})\hat{\mathbf{\Sigma}}_z^{(i)} - \frac{1}{\beta}\hat{\mathbf{\Sigma}}_z^{(i)}$. Consider a Singular Value Decomposition of $H^T J_\vartheta(z_0^{(i)}) = U\widetilde{D}V^T$ with $U \in \mathbb{R}^{d\times d}$ and $V \in \mathbb{R}^{\kappa\times\kappa}$ are unitary matrices and

$$\widetilde{D} = \begin{bmatrix} \delta_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \delta_\kappa \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{d\times\kappa}.$$

We have

$$\widetilde{D}^T\widetilde{D} = \begin{bmatrix} \delta_1^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \delta_\kappa^2 \end{bmatrix}$$

and can write

$$\begin{aligned} \hat{\mathbf{\Sigma}}_z^{(i)} =& \left(V(I_\kappa + \frac{1}{\beta}\widetilde{D}^T\widetilde{D})V^T\right)^{-1} \\ =& V\widehat{D}V^T, \end{aligned}$$

with $\widehat{D} := \operatorname{diag}\left(\frac{1}{1+\beta^{-1}\delta_1^2}, \dots, \frac{1}{1+\beta^{-1}\delta_\kappa^2}\right)$. For E it follows that

$$\begin{aligned} &\frac{1}{2\beta}\left[\hat{\mathbf{\Sigma}}_z^{(i)^2} - \frac{1}{\beta}\hat{\mathbf{\Sigma}}_z^{(i)} H_{f_x}(z_0^{(i)})\hat{\mathbf{\Sigma}}_z^{(i)} - 2\hat{\mathbf{\Sigma}}_z^{(i)}\right] \\ &= \frac{1}{2\beta}V\left[\widehat{D}^2 + \frac{1}{\beta}\widehat{D}\widetilde{D}^T\widetilde{D}\widehat{D} - 2\widehat{D}\right]V^T \\ &= \frac{-1}{2\beta}V\widehat{D}V^T. \end{aligned} \tag{33}$$

The justification of the last equation becomes apparent, when we consider one respective diagonal element $\delta_\cdot$ of the diagonal matrices in the equation. We have

$$\begin{aligned} &\frac{1}{(1+\beta^{-1}\delta_\cdot^2)^2} + \frac{\beta^{-1}\delta_\cdot^2}{(1+\beta^{-1}\delta_\cdot^2)^2} - \frac{2}{1+\beta^{-1}\delta_\cdot^2} \\ &= \frac{1+\beta^{-1}\delta_\cdot^2 - 2(1+\beta^{-1}\delta_\cdot^2)}{2(1+\beta^{-1}\delta_\cdot^2)^2} \\ &= \frac{-1}{(1+\beta^{-1}\delta_\cdot^2)}. \end{aligned}$$

We can further rephrase a part of (32), as we can use (28),(29) and have

$$J_{fx}(z_0^{(i)}) = \left(\nabla_\vartheta \log P_\theta\left(x^{(i)}|z_0^{(i)}\right)\right) J_\vartheta(z_0^{(i)}),$$

with $y^{(i)} := \left(\nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)^T + \Gamma J_\vartheta(z_0^{(i)}) z_0^{(i)}\right)$ we have

$$\begin{aligned}
&-\frac{1}{2} {z_0^{(i)}}^T H_{f_x}(z_0^{(i)}) z_0^{(i)} + J_{fx}(z_0^{(i)}) z_0^{(i)} \\
&= \frac{1}{2} {z_0^{(i)}}^T J_\vartheta(z_0^{(i)})^T H H^T J_\vartheta(z_0^{(i)}) z_0^{(i)} + \left(\nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)\right) H^{-T} H^T J_\vartheta(z_0^{(i)}) z_0^{(i)} \\
&= \frac{1}{2} {y^{(i)}}^T \Gamma^{-1} y^{(i)} - \frac{1}{2} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right) \Gamma^{-1} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)^T.
\end{aligned}$$

Using this and (33), for (32) we get

$$\begin{aligned}
-\widehat{\mathcal{L}}(\theta) =& \frac{1}{N} \sum_{i=1}^{N} \Bigg[{y^{(i)}}^T \left(\frac{-1}{2\beta} J_\vartheta(z_0^{(i)}) V \widehat{D} V^T J_\vartheta(z_0^{(i)})^T + \frac{1}{2} \Gamma^{-1} \right) y^{(i)} - \log P_\theta(x|z_0^{(i)}) \\
&- \frac{1}{2} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right) \Gamma^{-1} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)^T + \frac{\beta}{2} \log \left| \hat{\mathbf{\Sigma}}_z^{(i)^{-1}} \right| \Bigg].
\end{aligned}$$

Since $\Gamma = HH^T$, we can further rewrite

$$\begin{aligned}
&\frac{-1}{2\beta} J_\vartheta(z_0^{(i)}) V \widehat{D} V^T J_\vartheta(z_0^{(i)})^T + \frac{1}{2} \Gamma^{-1} \\
&= \frac{1}{2} H^{-T} U \left[-\beta^{-1} \widetilde{D} \widehat{D} \widetilde{D}^T + I \right] U^T H^{-1} \\
&= \frac{1}{2} H^{-T} U \left[\beta^{-1} \widetilde{D} \widetilde{D}^T + I \right]^{-1} U^T H^{-1} \\
&= \frac{1}{2} \Gamma^{-1} \left[\Gamma^{-1} + \beta^{-1} J_\vartheta(z_0^{(i)}) J_\vartheta(z_0^{(i)})^T \right]^{-1} \Gamma^{-1}
\end{aligned}$$

and together with

$$\begin{aligned}
\log \left| \hat{\mathbf{\Sigma}}_z^{(i)^{-1}} \right| &= \log \left| I_\kappa + \beta^{-1} \widetilde{D}^T \widetilde{D} \right| = \log \left| I_d + \beta^{-1} \widetilde{D} \widetilde{D}^T \right| \\
&= \log \left| \Gamma^{-1} + \beta^{-1} J_\vartheta(z_0^{(i)}) J_\vartheta(z_0^{(i)})^T \right| + \log |\Gamma|,
\end{aligned}$$

for $C(z_0^{(i)}) := \Gamma^{-1} + \beta^{-1} J_\vartheta(z_0^{(i)}) J_\vartheta(z_0^{(i)})^T$ we get for our target function

$$\begin{aligned}
-\widehat{\mathcal{L}}(\theta) =& \frac{1}{N} \sum_{i=1}^{N} \Bigg[\frac{1}{2} {y^{(i)}}^T \Gamma^{-1} C(z_0^{(i)})^{-1} \Gamma^{-1} y^{(i)} - \log P_\theta(x|z_0^{(i)}) \\
&- \frac{1}{2} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right) \Gamma^{-1} \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)^T \\
&+ \frac{\beta}{2} \log \left| C(z_0^{(i)}) \right| \Bigg] + \frac{\beta}{2} \log |\Gamma|.
\end{aligned}$$

Based on the assumption, that the distribution of X given Z belongs to the EDF and $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$, we have

$$\begin{aligned} -\log P_\theta(x|z_0^{(i)}) =& d/\varphi F(0) - \sum_{j=1}^{d} K(x_j^{(i)}, \varphi), \\ \nabla_\vartheta \log P_\theta \left(x^{(i)}|z_0^{(i)}\right)^T =& \varphi^{-1}\left(x^{(i)} - F'(0)\right) \end{aligned}$$

$$\Gamma^{-1} = F''(0)^{-1}\varphi \boldsymbol{I}_d$$

and hence

$$\Gamma^{-1} y^{(i)} = F''(0)^{-1}\left(x^{(i)} - F'(0)\right) + J_\vartheta(z_0^{(i)}) z_0^{(i)}.$$

Therefore, we get

$$\begin{aligned} -\widehat{\mathcal{L}}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\Bigg[& \frac{1}{2}\left(F''(0)^{-1}(x^{(i)} - F'(0)) + J_\vartheta(z_0^{(i)}) z_0^{(i)}\right)^T C(z_0^{(i)})^{-1} \\ & \left(F''(0)^{-1}(x^{(i)} - F'(0)) + J_\vartheta(z_0^{(i)}) z_0^{(i)}\right) \\ & + \frac{\beta}{2}\log\left|C(z_0^{(i)})\right| + \frac{\beta \cdot d}{2}\log\left(\varphi^{-1}F''(0)\right) + \frac{1}{2}D(\varphi)\Bigg], \end{aligned}$$

with

$$C(z_0^{(i)}) = F''(0)^{-1}\varphi \boldsymbol{I}_d + \beta^{-1} J_\vartheta(z_0^{(i)}) J_\vartheta(z_0^{(i)})^T$$

and

$$D(\varphi) := \frac{2d}{\varphi}F(0) - \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{F''(0)\varphi}\left\|x^{(i)} - F'(0)\right\|_2^2 + 2\sum_{j=1}^{d} K(x_j^{(i)}, \varphi)\right]. \tag{34}$$

■

A.3 Proof of Corollary 1

Proof We look at the Gaussian, Binomial and Poisson cases separately.

- Gaussian case: As for the EDF representation of a Gaussian model we have $F(\vartheta) = \frac{\vartheta^2}{2}$. As $\boldsymbol{\vartheta}$ is assumed to be a piecewise linear function, it follows directly that all third or higher derivatives of $f_x(z)$ vanish and we are done.

- Binomial case: Given $\boldsymbol{\vartheta}$ is a p.w. linear function we have for $u, v, m, n \in \{1, \ldots, \kappa\}$

$$\frac{\partial^3 f_x(z)}{\partial z_u \partial z_v \partial z_m} = \sum_{j=1}^{d} -F^{(3)}(\boldsymbol{\vartheta}_j(z))\frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_u}\frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_v}\frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_m} \tag{35}$$

and

$$\frac{\partial^4 f_x(z)}{\partial z_u \partial z_v \partial z_m \partial z_n} = \sum_{j=1}^{d} -F^{(4)}(\boldsymbol{\vartheta}_j(z)) \frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_u} \frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_v} \frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_m} \frac{\partial \boldsymbol{\vartheta}_j(z)}{\partial z_n}. \tag{36}$$

For $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$, we have $F^{(3)}(\boldsymbol{\vartheta}_j(z)) = 0$ and can conclude that for these points a second order Taylor approximation is the same as a third order approximation. The remainder is given in the Lagrange form by

$$R_2(z, z_0^{(i)}) = R_3(z, z_0^{(i)}) = \sum_{j=1}^{d} \frac{-F^{(4)}(\boldsymbol{\vartheta}_j(\xi))}{4!} \left(\sum_{u=1}^{\kappa} \frac{\partial \boldsymbol{\vartheta}_j(\xi)}{\partial z_u} (z - z_0^{(i)})_u \right)^4,$$

with $\xi = z_0^{(i)} + c \cdot (z - z_0^{(i)})$, where $c \in [0, 1]$. We can rewrite $\sum_{u=1}^{\kappa} \frac{\partial \boldsymbol{\vartheta}_j(\xi)}{\partial z_u} (z - z_0^{(i)})_u = J_{\vartheta j}(\xi)(z - z_0^{(i)})$ and we can show that

$$-F^{(4)}(\boldsymbol{\vartheta}_j(\xi)) \in [-n/24, n/8].$$

Using this, the first statement follows for the Binomial case follows.

For the second statement we write the remainder in the Lagrange form for a second order-approximation and get

$$\begin{aligned} R_2(z, z_0^{(i)}) &= \sum_{j=1}^{d} \frac{-F^{(3)}(\boldsymbol{\vartheta}_j(\xi))}{3!} \left(\sum_{u=1}^{\kappa} \frac{\partial \boldsymbol{\vartheta}_j(\xi)}{\partial z_u} (z - z_0^{(i)})_u \right)^3 \\ &= \sum_{j=1}^{d} \frac{-F^{(3)}(\boldsymbol{\vartheta}_j(\xi))}{3!} \left(J_{\vartheta j}(\xi)(z - z_0^{(i)}) \right)^3, \end{aligned}$$

If we assume $\boldsymbol{\vartheta}$ to be an affine transformation on the convex set spanned by z and $z_0^{(i)}$ we have

$$\boldsymbol{\vartheta}(\xi) = W\xi + b$$

for some $W \in \mathbb{R}^{d \times \kappa}$ and $b \in \mathbb{R}^d$. Hence we can rewrite the equation above as

$$R_2(z, z_0^{(i)}) = \sum_{j=1}^{d} \frac{-F^{(3)}(W_j z_0^{(i)} + b_j + c \cdot W_j(z - z_0^{(i)}))}{3!} \left(W_j(\xi)(z - z_0^{(i)}) \right)^3$$

Since $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$, we can write

$$W_j z_0^{(i)} + b_j + c \cdot W_j(z - z_0^{(i)}) = c\boldsymbol{\vartheta}_j(z).$$

Notice that

$$F(\vartheta)^{(3)} = -n \frac{e^{\vartheta} \left(e^{\vartheta} - 1 \right)}{\left(e^{\vartheta} + 1 \right)^3}$$

is point symmetric to zero, with negative values if $\vartheta > 0$ and positive values if $\vartheta < 0$. Therefore, we can write

$$R_2(z, z_0^{(i)}) = \sum_{j=1}^{d} \frac{-|c\boldsymbol{\vartheta}_j(z)|}{3!} |\boldsymbol{\vartheta}_j(z)|^3 \geq 0.$$

This yields the statement.

- Poisson case: We have $F(\vartheta) = \exp(\vartheta)$ and can write the remainder in Lagrange form as

$$R_2(z, z_0^{(i)}) = \sum_{j=1}^{d} \frac{-\exp(\boldsymbol{\vartheta}_j(\xi))}{6} \left(J_{\vartheta j}(\xi)(z - z_0^{(i)})\right)^3,$$

with $\xi = z_0^{(i)} + c \cdot (z - z_0^{(i)})$, where $c \in [0, 1]$. Since we assume $\boldsymbol{\vartheta}$ to be an affine transformation on the convex set spanned by z and $z_0^{(i)}$ and $z_0^{(i)} \in ker(\boldsymbol{\vartheta})$, like in the Binomial case we have

$$R_2(z, z_0^{(i)}) = \sum_{j=1}^{d} \frac{-\exp(c\boldsymbol{\vartheta}_j(z))}{6} (\boldsymbol{\vartheta}_j(z))^3,$$

for some $W \in \mathbb{R}^{d \times \kappa}$ and $b \in \mathbb{R}^d$.

Consider the two cases $\boldsymbol{\vartheta}_j(z) < 0$ and $\boldsymbol{\vartheta}_j(z) \geq 0$. For the case $\boldsymbol{\vartheta}_j(z) < 0$ we have

$$\begin{aligned}
&\min_{c\in[0,1]} \left(\frac{\exp(-c|\boldsymbol{\vartheta}_j(z)|)}{6} |\boldsymbol{\vartheta}_j(z)|^3 \right) \\
&= \min_{c\in[0,1]} \left(\exp(-c|\boldsymbol{\vartheta}_j(z)|)\right) |\boldsymbol{\vartheta}_j(z)|^3 /6 \\
&= \min_{c\in[0,1]} \left(\frac{1}{\exp(c|\boldsymbol{\vartheta}_j(z)|)} \right) |\boldsymbol{\vartheta}_j(z)|^3 /6 \\
&= \left(\frac{1}{\max_{c\in[0,1]} \exp(c|\boldsymbol{\vartheta}_j(z)|)} \right) |\boldsymbol{\vartheta}_j(z)|^3 /6 \\
&= \left(\frac{1}{\exp(|\boldsymbol{\vartheta}_j(z)|)} \right) |\boldsymbol{\vartheta}_j(z)|^3 /6
\end{aligned}$$

and

$$\begin{aligned}
&\max_{c\in[0,1]} \left(\frac{\exp(-c|\boldsymbol{\vartheta}_j(z)|)}{6} |\boldsymbol{\vartheta}_j(z)|^3 \right) \\
&= \left(\frac{1}{\min_{c\in[0,1]} \exp(c|\boldsymbol{\vartheta}_j(z)|)} \right) |\boldsymbol{\vartheta}_j(z)|^3 /6 \\
&= |\boldsymbol{\vartheta}_j(z)|^3 /6
\end{aligned}$$

and hence

$$\sum_{j=1}^{d} -\boldsymbol{\vartheta}_j(z)^3 \cdot \exp(\boldsymbol{\vartheta}_j(z)/6 \leq R_2(z, z_0^{(i)}) \leq \sum_{j=1}^{d} -\boldsymbol{\vartheta}_j(z)^3/6. \tag{37}$$

For the case $\boldsymbol{\vartheta}_j(z) \geq 0$ we have

$$\begin{aligned}
&\min_{c\in[0,1]} \left(\frac{-\exp(c|\boldsymbol{\vartheta}_j(z)|)}{6} \left|\boldsymbol{\vartheta}_j(z)\right|^3 \right) \\
&= \max_{c\in[0,1]} \left(\exp(c|\boldsymbol{\vartheta}_j(z)|)\right) \cdot - \left|\boldsymbol{\vartheta}_j(z)\right|^3/6 \\
&= \exp(|\boldsymbol{\vartheta}_j(z)|) \cdot - \left|\boldsymbol{\vartheta}_j(z)\right|^3/6
\end{aligned}$$

as well as

$$\begin{aligned}
&\max_{c\in[0,1]} \left(\frac{-\exp(c|\boldsymbol{\vartheta}_j(z)|)}{6} \left|\boldsymbol{\vartheta}_j(z)\right|^3 \right) \\
&= \min_{c\in[0,1]} \left(\exp(c|\boldsymbol{\vartheta}_j(z)|)\right) \cdot - \left|\boldsymbol{\vartheta}_j(z)\right|^3/6 \\
&= - \left|\boldsymbol{\vartheta}_j(z)\right|^3/6
\end{aligned}$$

and it follows again (37).

■

Appendix B. Simulation

B.1 Architecture, latent dimension κ and the last decoder activation

For the architecture, we look at two different versions, which we denote as "deep" and "canonical". The deep architecture is given as in Dai et al. (2018), by

$$\begin{array}{c}
x(d) \to E_1(2000) \to E_2(1000) \to \ \boldsymbol{\mu}_z(\kappa) \ \to D_1(1000) \to D_2(2000) \to \hat{x}(d), \\
\qquad\qquad\qquad\qquad \searrow \log \boldsymbol{\sigma}_z^2(\kappa) \ \nearrow
\end{array}$$

where E/D denote encoder/decoder layers and the values in the brackets indicate the dimension of the layer. So, κ is the dimension of the latent space and we use the values $2, 5$ and 20 for different simulation setups. The covariance of the variational distribution is set as diagonal matrix.

The canonical architecture is given by

$$\begin{array}{c}
x(d) \to E_1(2000) \to E_2(d) \to \ \boldsymbol{\mu}_z(\kappa) \ \to \ \hat{x}(d). \\
\qquad\qquad\qquad\qquad \searrow \log \boldsymbol{\sigma}_z^2(\kappa) \ \nearrow
\end{array}$$

The canonical architecture conforms to the assumptions of Proposition 1.

The hidden layers of the encoder and the decoder are implemented with ReLU-activation (see Nair and Hinton 2010), which is known to be highly expressive. The "$\boldsymbol{\mu}_z$"- and "$\log \boldsymbol{\sigma}_z^2$"-layer have linear activations. The last layer "$\hat{x}$" has either a linear (Gaussian observation model) or a sigmoid (Bernoulli observation model) activation as reported in Table 2.

Apart from the fact that we expect both architectures to provide a better loss than provided by our theoretical bound, it should be able to represent the optimal $\hat{\boldsymbol{\mu}}_z^{(i)}$ and diagonal entries of the optimal $\hat{\boldsymbol{\Sigma}}_z^{(i)}$ in (20) and (21) if necessary.

B.2 Data

We consider three data sets: a synthetic data set (we describe the construction at the end of this chapter), the mnist data set (see LeCun et al. 2010) and the frey data set.[5] Each set is transformed to only have values in between 0 and 1. As training/test split, we have

- synthetic: 6700 / 3300
- mnist: 60000 / 10000
- frey: 1316 / 649

The synthetic data is constructed in the fashion of Lee et al. (2010). For $k = 2$, $N = 10000$ and $d = 200$, we generate two matrices $A \in \mathbb{R}^{N\times k}$ and $B \in \mathbb{R}^{d\times k}$. A is identifiable with principal components and B with (sparse) loading vectors of a PCA. The two-dimensional principal components $a^{(i)}(i = 1, \dots, N)$ of A are drawn from normal distributions, so that $a_1^{(i)} \sim \mathcal{N}(0, 0.09)$ and $a_2^{(i)} \sim \mathcal{N}(0, 0.25)$. The sparse loading vectors are constructed by setting B to zero except for $b_{j,1} = 1, j = 1, \dots, 20$ and $b_{j,2} = 1, j = 21, \dots, 40$.

Given A and B we calculate

$$\boldsymbol{\Xi} := A \cdot B^T$$

and the probability matrix Π, with

$$\Pi = \sigma(\boldsymbol{\Xi}),$$

where we apply the sigmoid function $\sigma(\cdot)$ element-wise. We then use the probabilities $\Pi_j^{(i)}$ to independently draw samples

$$x_j^{(i)} \sim Bern(\Pi_j^{(i)}).$$

With the data $X_{Data} := (x_j^{(i)})_{i=1,\dots,N;j=1,\dots,d}$, we conduct the simulation.

B.3 Initialization

In each simulation, we compare two competing VAE with the same architecture and different initialization. We call these initializations "Bench" (benchmark) and "MLE-B" (MLE - based).

- For the benchmark, all network weights and biases are initialized as proposed by He et al. (2015). This initialization particularly considers rectifier non-linearities.

5. Taken from `https://cs.nyu.edu/~roweis/data.html`

- For MLE-B we use the same initialization as for the benchmark except for the weights and biases of the "$\boldsymbol{\mu}_z$"-, "$\log \boldsymbol{\sigma}_z^2$"- and "$\hat{x}$"-layer. The MLE-B initialization is cheap to calculate as only a singular value decomposition of the training data is needed for the maximum likelihood estimates.

  For the "$\hat{x}$"-layer we use $\hat{W}$ as weights and $\hat{b}$ as bias. In case of over-parametrized nets, more edges lead into the affected layers than we need for MLE-B. This problem only concerns the weights and not the biases. We solve this by initializing not needed dimensions of the weights with zero.

  For the "$\boldsymbol{\mu}_z$"-layer we use $W_e := \frac{1}{\hat{\varphi}\beta}\hat{\boldsymbol{\Sigma}}_z\hat{W}^T$ as weights and $b_e := \frac{1}{\hat{\varphi}\beta}\hat{\boldsymbol{\Sigma}}_z\hat{W}^T\bar{x}$ as bias.

  For the "$\log \boldsymbol{\sigma}_z^2$"-layer we use the log of a diagonal (20) as bias. The weights a initialized according to He et al. (2015).

B.4 Latent space activity: Histogram Distance

To compare the resulting Activity statistics after training, we calculate a distance between two histograms. If we choose the amount of bins to be $b \in \mathbb{N}$, the histograms can be represented as b-dimensional, integer valued vectors $x, y \in \mathbb{N}^b$, with $\sum_i x_i = \sum_i y_i = \kappa$ and $x_i, y_i \geq 0$ for all i, where κ denotes the latent dimension. We define the distance between two histograms as

$$d(x, y) := 1 - \frac{\sum_{i=1}^{b} \min\{x_i, y_i\}}{\kappa}. \tag{38}$$

This distance fulfils all properties of a metric on the defined set of histograms. Further, if two histogram representations are equal, we have the minimal value $d(x, y) = 0$. The maximum distance of $d(x, y) = 1$ is achieved, if bars never overlap.

For our experiments, to calculate the distance we choose 10 bins, given by

$$[0, 0.1), [0.1, 0.2), \ldots, [0.9, \infty).$$

B.5 Latent space activity: Bernoulli Case

Table 4: We display the distances ($\in [0, 1]$, lower is better) of the histograms to the corresponding analytical calculation for the Bernoulli observation model. Displayed are the results of 10 simulations as "mean±std".

	Beta 1	Beta 2	Beta 5	Beta 10	Beta 15	Beta 20
Affine	0.48 (±0.03)	0.38 (±0.03)	0.15 (±0.03)	0.08 (±0.01)	0.04 (±0.02)	0 (±0.01)
ReLU	0.52 (±0.03)	0.35 (±0.03)	0.16 (±0.02)	0.09 (±0.01)	0.07 (±0)	0.03 (±0)

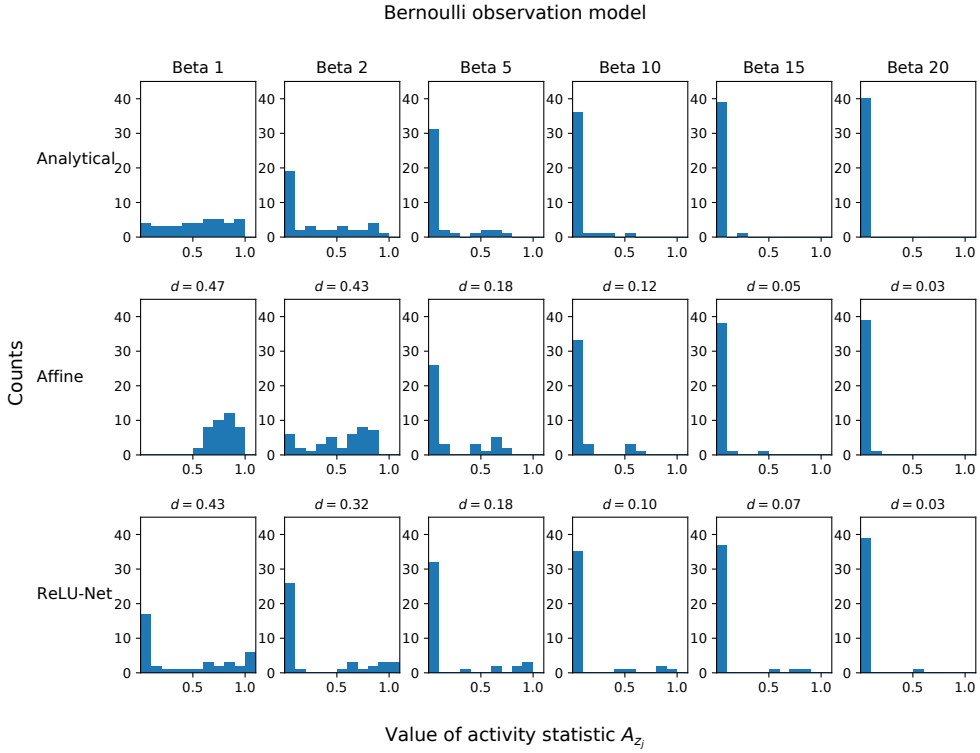

Figure 3: The figure shows histograms of the activities for 40 latent dimensions for our analytical calculation and an Affine/ ReLU-Net decoder (as described in appendix B.1) after training. We have considered the mnist data set with a Bernoulli observation model. Above each Affine and ReLU-Net histogram plot, we show the distance ($\in [0, 1]$, lower is better) as defined in (38) to the analytical histogram.

B.6 Simulation results

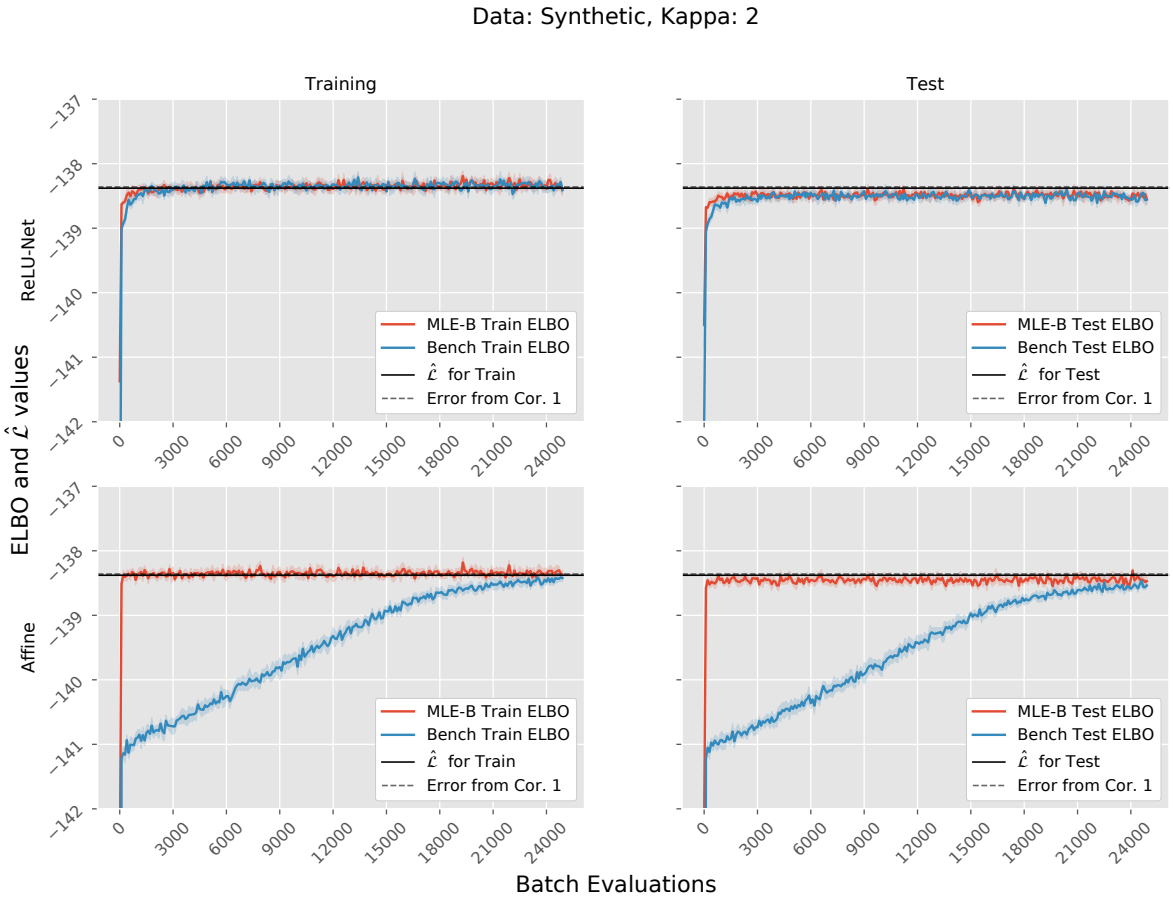

Figure 4: The pictures show the setups ReLU-Net and Affine with synthetic data, $\kappa = 2$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

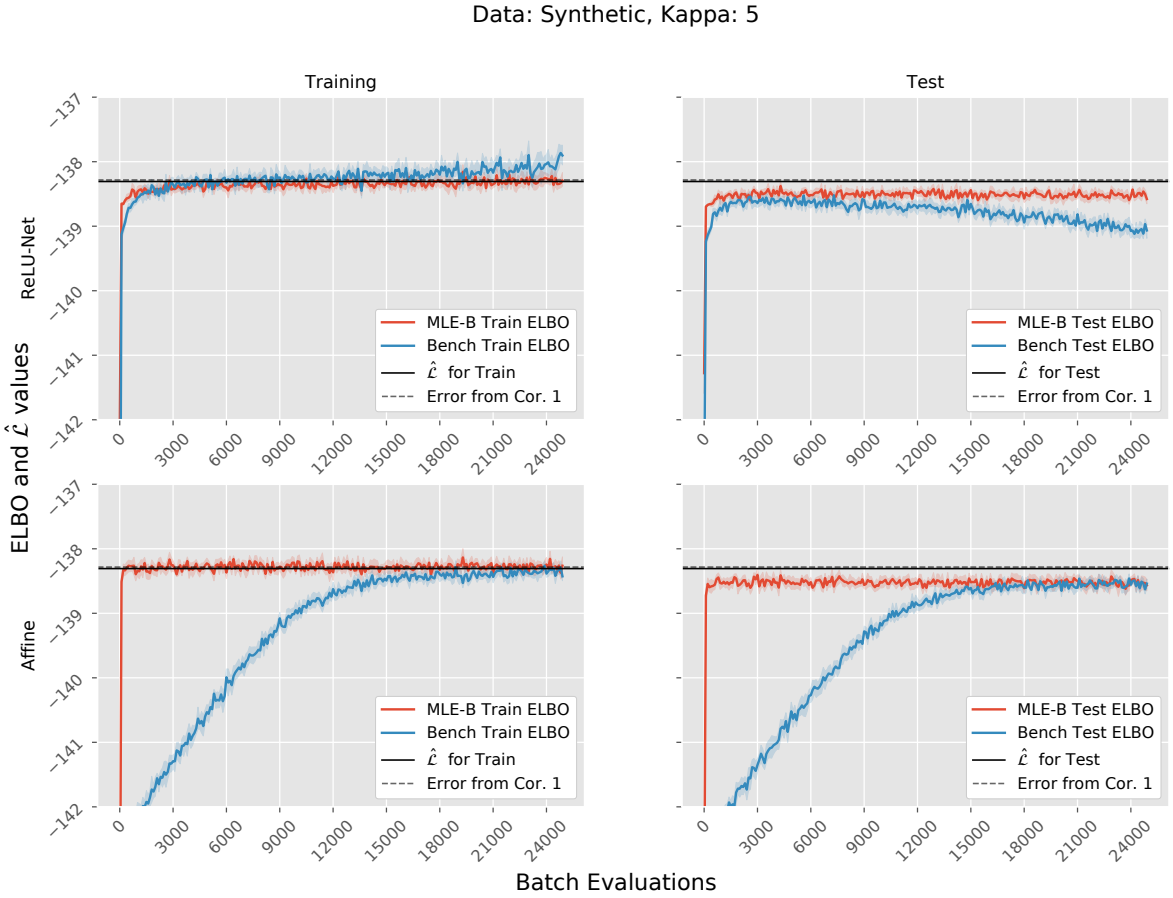

Figure 5: The pictures show the setups ReLU-Net and Affine with synthetic data, $\kappa = 5$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

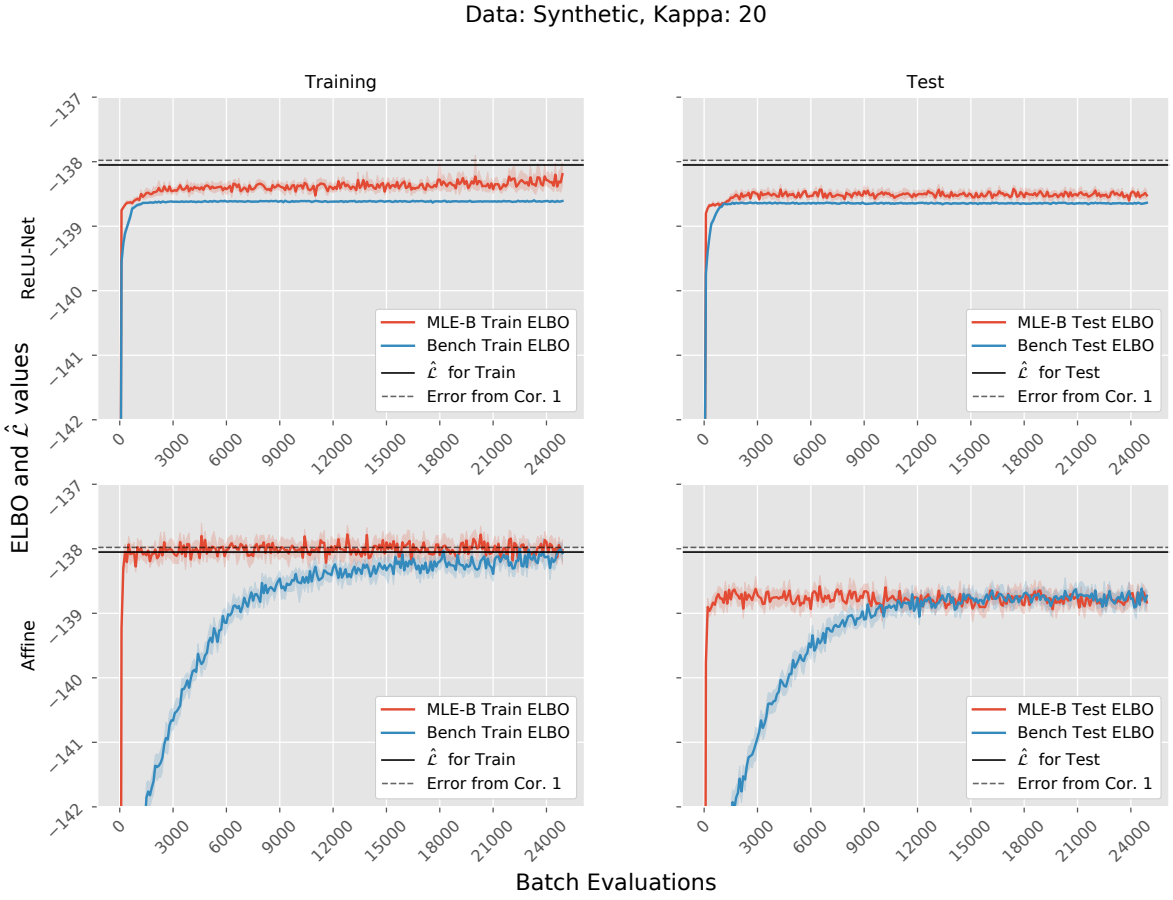

Figure 6: The pictures show the setups ReLU-Net and Affine with synthetic data, $\kappa = 20$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

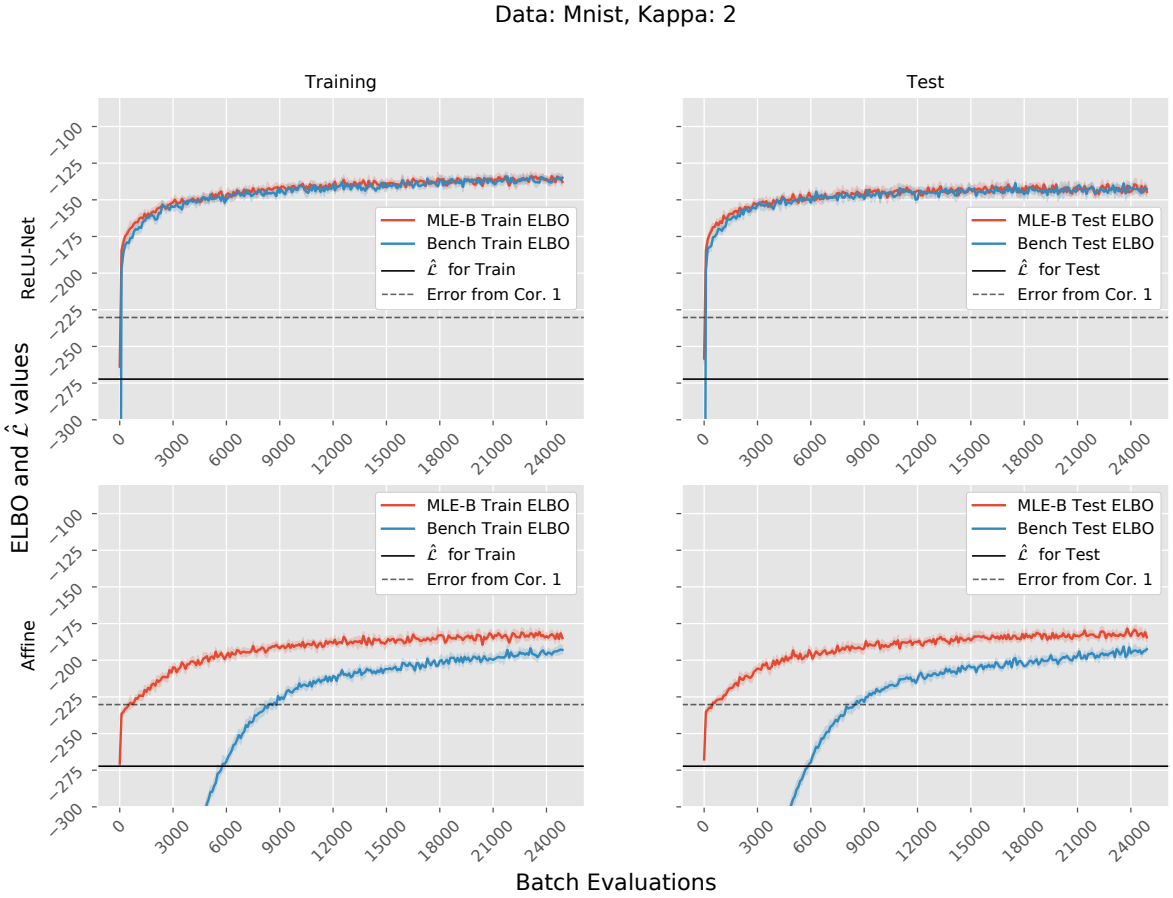

Figure 7: The pictures show the setups ReLU-Net and Affine with mnist data, $\kappa = 2$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

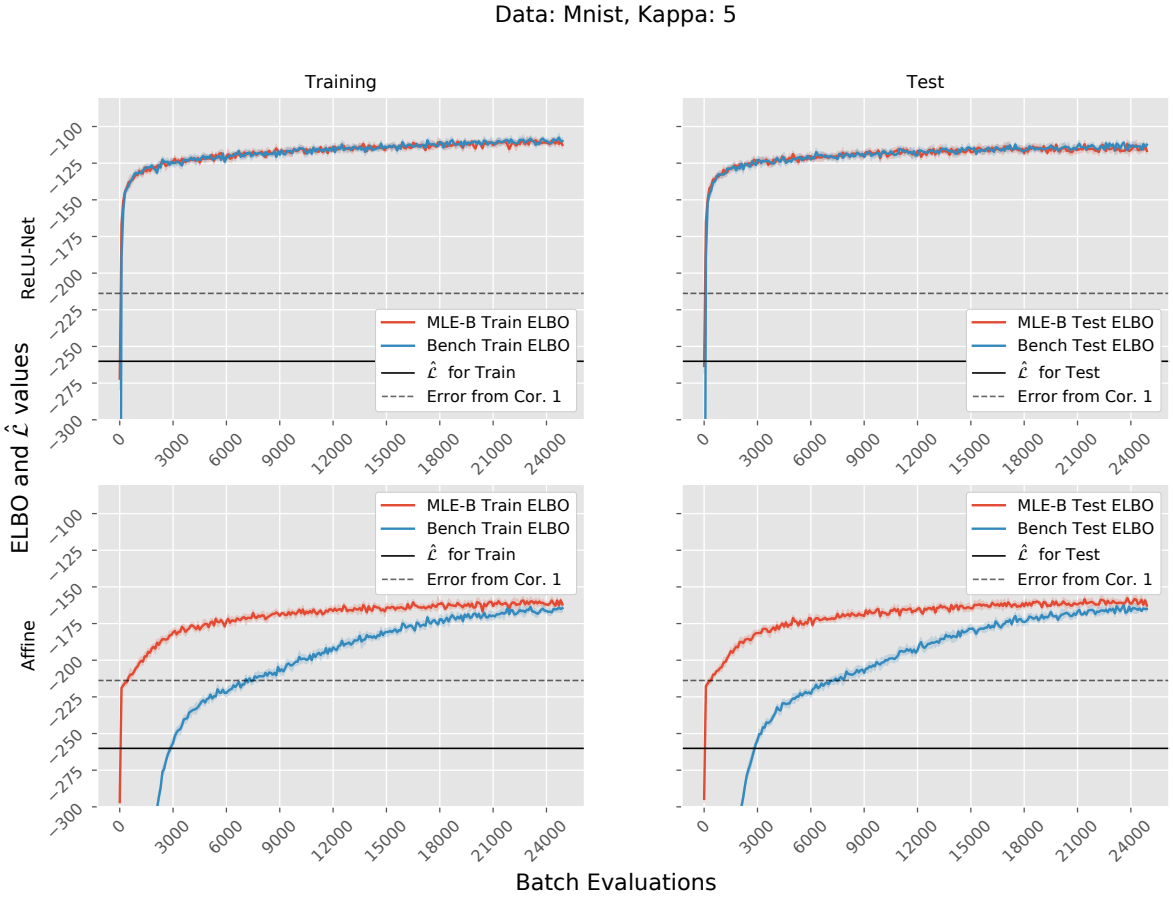

Figure 8: The pictures show the setups ReLU-Net and Affine with mnist data, $\kappa = 5$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

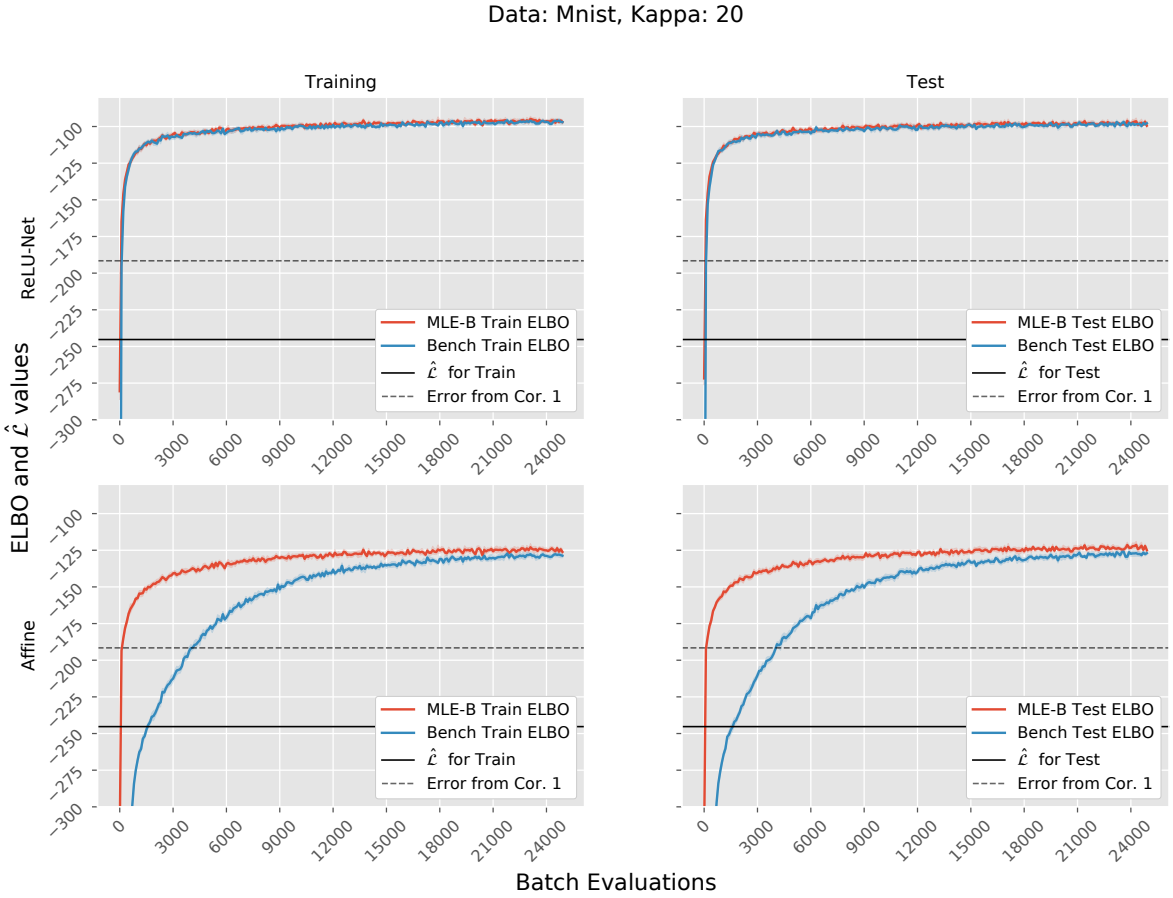

Figure 9: The pictures show the setups ReLU-Net and Affine with mnist data, $\kappa = 20$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

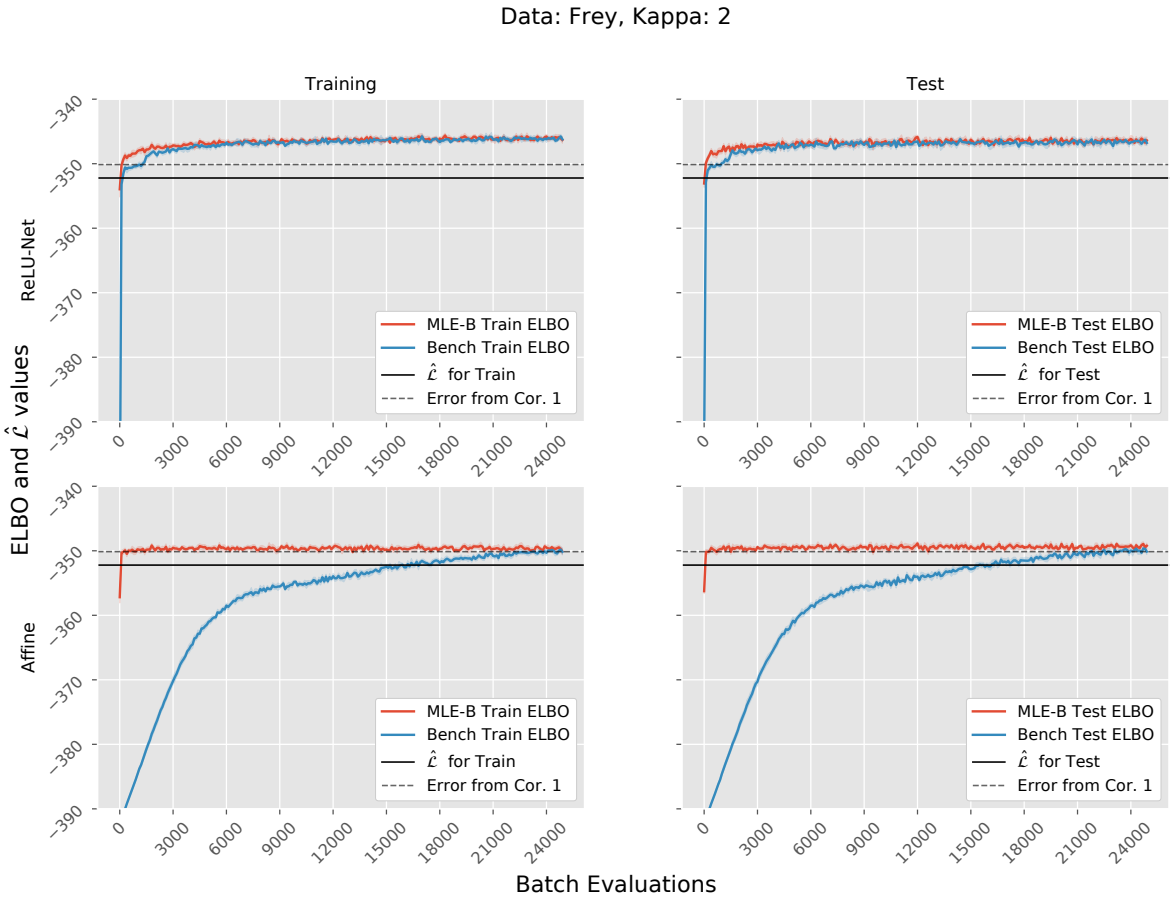

Figure 10: The pictures show the setups ReLU-Net and Affine with frey data, $\kappa = 2$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

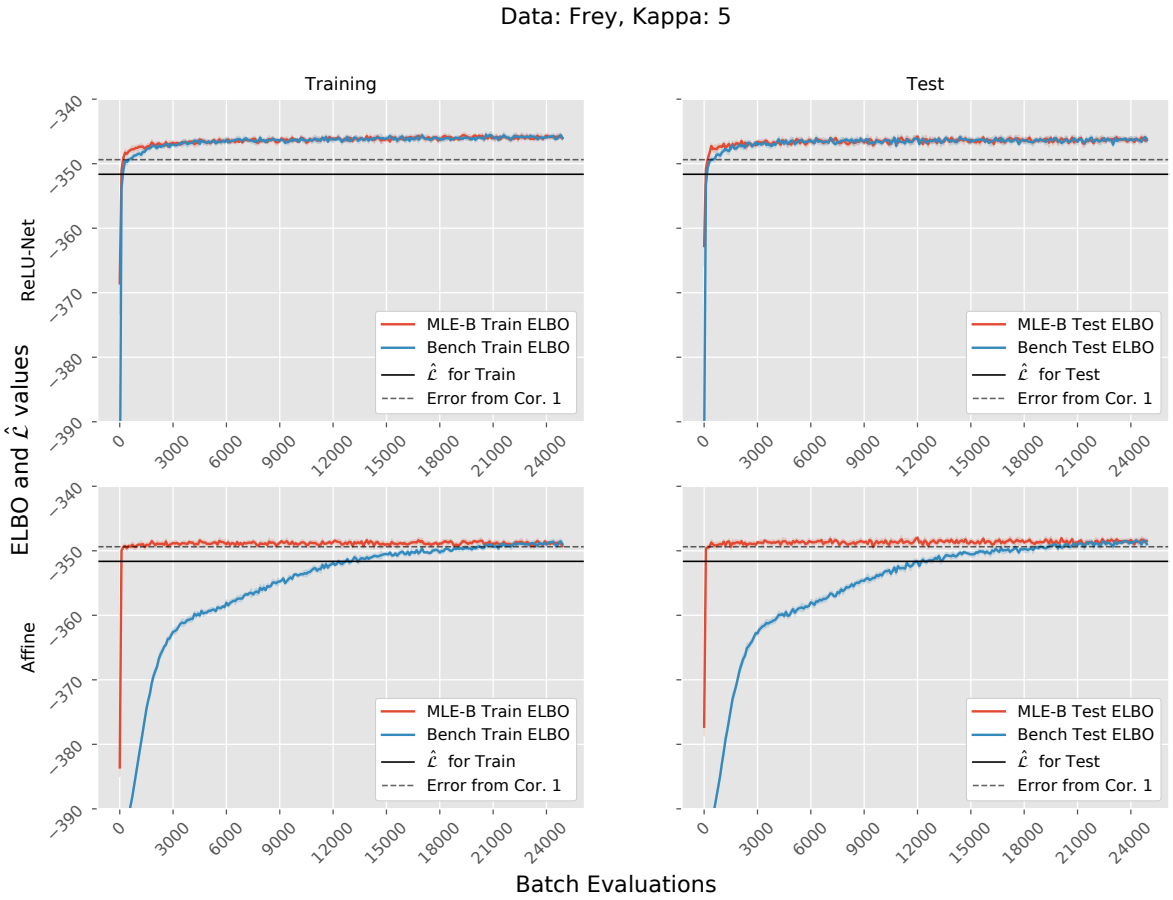

Figure 11: The pictures show the setups ReLU-Net and Affine with frey data, $\kappa = 5$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

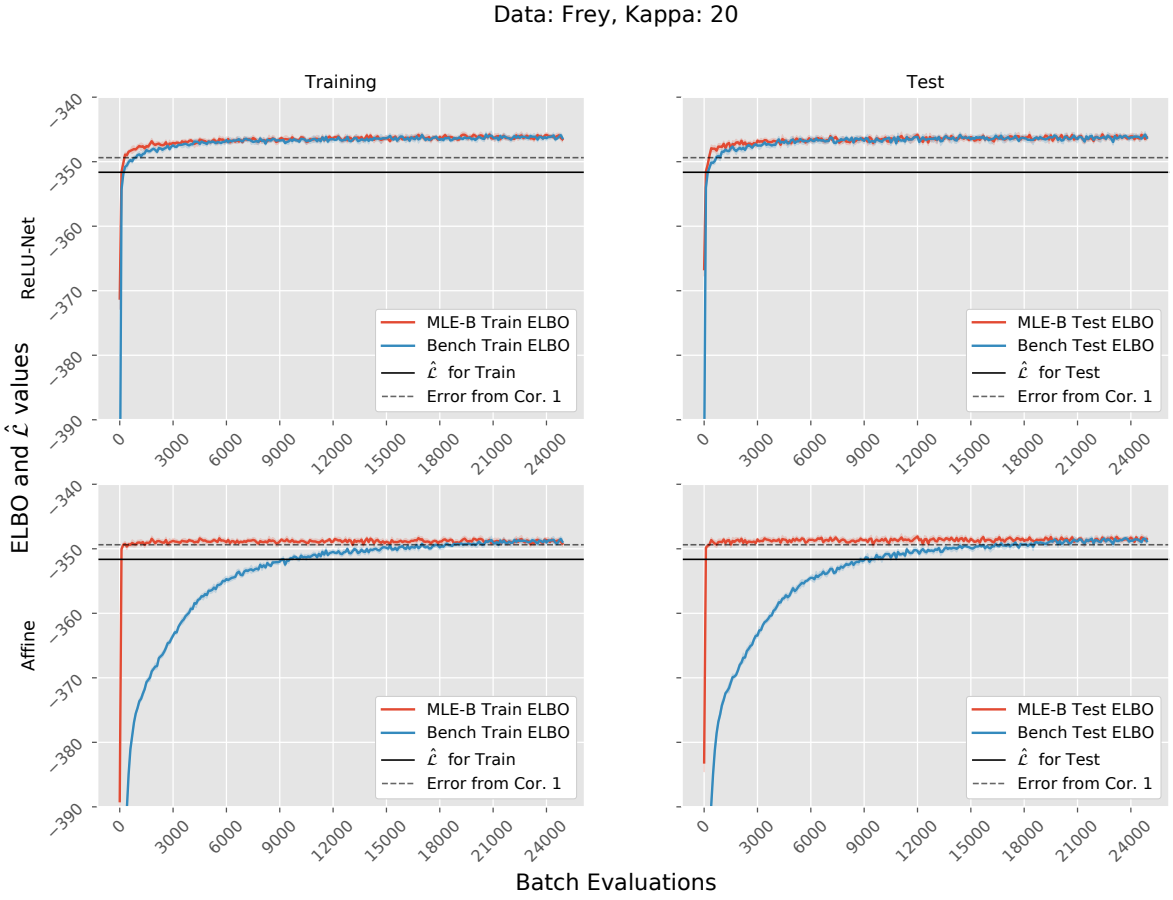

Figure 12: The pictures show the setups ReLU-Net and Affine with frey data, $\kappa = 20$. Displayed are the ELBOs of both initialisations MLE-B and Bench as well as the lower bound $\widehat{\mathcal{L}}$ and the expected error $\mathbb{E}_{q_{\hat{\phi}}}[R_2(\hat{\theta})]$ as provided by Corollary 1, calculated based on MLE. On the left the ELBO values are calculated with the training data and on the right with test data. The curves are based on simulations with 10 different seeds. We display the average training performances with pointwise 0.95 confidence intervals.

References

Alexander A Alemi, Ben Poole, Ian Fischer, Joshua V Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken ELBO. In *International Conference on Machine Learning(ICML)*, 2018.

Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989. ISSN 08936080. doi: 10.1016/0893-6080(89)90014-2.

Ole Barndorff-Nielsen. *Information and Exponential Families in Statistical Theory*. John Wiley & Sons, 2014.

Thomas Blaschke, Marcus Olivecrona, Ola Engkvist, Jürgen Bajorath, and Hongming Chen. Application of generative autoencoder in de novo molecular design. *Molecular Informatics*, 37(1-2), jan 2018. doi: 10.1002/minf.201700123. URL http://doi.wiley.com/10.1002/minf.201700123.

Hervé Bourlard and Yves Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biol. Cybern*, 59:291–294, 1988. URL https://link.springer.com/content/pdf/10.1007%2FBF00332918.pdf.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. In *CoNLL 2016 - 20th SIGNLL Conference on Computational Natural Language Learning, Proceedings*, pages 10–21. Association for Computational Linguistics (ACL), nov 2015. URL http://arxiv.org/abs/1511.06349.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In *International Conference on Learning Representations(ICLR)*, sep 2015. URL http://arxiv.org/abs/1509.00519.

Bin Dai and David Wipf. Diagnosing and enhancing VAE models. In *International Conference on Learning Representations(ICLR)*, 2019. ISBN 1903.05789v2. URL arxiv:1903.05789v2.

Bin Dai, Yu Wang, John Aston, and David Wipf. Connections with robust PCA and the role of emergent dparsity in variational autoencoder models. *Journal of Machine Learning Research*, 19: 1–42, 2018. URL http://jmlr.org/papers/v19/17-704.html.

Bin Dai, Ziyu Wang, and David Wipf. The usual suspects? Reassessing blame for VAE posterior collapse. In *International Conference on Machine Learning(ICML)*, 2020.

Xintao Duan, Jingjing Liu, and En Zhang. Efficient image encryption and compression based on a VAE generative model. *Journal of Real-Time Image Processing*, 16(3):765–773, jun 2019. ISSN 1861-8200. doi: 10.1007/s11554-018-0826-4. URL http://link.springer.com/10.1007/s11554-018-0826-4.

Partha Ghosh, Mehdi S. M. Sajjadi, Antonio Vergari, Michael Black, and Bernhard Schölkopf. From Variational to deterministic Autoencoders. In *International Conference on Learning Representations(ICLR)*, 2020.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. 2016.

Junxian He, Daniel Spokoyny, Graham Neubig, and Taylor Berg-Kirkpatrick. Lagging inference networks and posterior collapse in variational autoencoders. In *International Conference on Learning Representations(ICLR)*, pages 1–15, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. URL https://www.cv-foundation.org/openaccess/content_iccv_2015/html/He_Delving_Deep_into_ICCV_2015_paper.html.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, Alexander Lerchner, and Google Deepmind. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations(ICLR)*, nov 2017.

Chin-Wei Huang, Shawn Tan, Alexandre Lacoste, and Aaron Courville. Improving explorability in variational inference with annealed variational objectives. Technical report, 2018.

Bent Jorgensen. Some properties of exponential dispersion models. *Scandinavian Journal of Statistics*, 13(3):187–197, 1986.

Bent Jorgensen. Exponential dispersion models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 49(2):127–162, 1987.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations(ICLR)*, 2015.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations(ICLR)*, 2014. URL https://arxiv.org/pdf/1312.6114.pdf.

Diederik P. Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improving variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems(NeurIPS)*, pages 4743–4751, jun 2016. URL http://arxiv.org/abs/1606.04934.

Abhishek Kumar and Ben Poole. On implicit regularization in β-VAEs. In *International Conference on Machine Learning(ICML)*, jan 2020. URL http://arxiv.org/abs/2002.00041.

Daniel Kunin, Jonathan M. Bloom, Aleksandrina Goeva, and Cotton Seed. Loss landscapes of regularized linear autoencoders. In *International Conference on Machine Learning(ICML)*, 2019.

Yann LeCun. Keynote: The future is self-supervised. In *International Conference on Learning Representations(ICLR)*, 2020.

Yann LeCun, Corinna Cortes, and Christopher C.J. Burges. MNIST handwritten digit database. *ATT Labs [Online]*, 2, 2010. URL http://yann.lecun.com/exdb/mnist.

Seokho Lee, Jianhua Z. Huang, and Jianhua Hu. Sparse logistic principal components analysis for binary data. *The annals of applied statistics*, 4(3):1579–1601, sep 2010. ISSN 1932-6157. doi: 10.1214/10-AOAS327SUPP. URL http://www.ncbi.nlm.nih.gov/pubmed/21116451http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2992445.

James Lucas, George Tucker, Roger Grosse, and Mohammad Norouzi. Don't blame the ELBO! A linear VAE perspective on posterior collapse. In *Advances in Neural Information Processing Systems(NeurIPS)*, 2019. URL https://sites.google.com/view/dont-blame-the-elbo.

Peter McCullagh and John A. Nelder. *Generalized Linear Models*. Chapman and Hall, London, 2nd edition, 1989.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *International Conference on Machine Learning(ICML)*, pages 807–814, 2010.

John A. Nelder and Robert W. M. Wedderburn. Generalized linear models. *Source: Journal of the Royal Statistical Society. Series A (General)*, 135(3):370–384, 1972.

Ali Razavi, Aaron van den Oord, Ben Poole, and Oriol Vinyals. Preventing posterior collapse with delta-VAEs. In *International Conference on Learning Representations(ICLR)*, 2019.

Danilo Jimenez Rezende and Fabio Viola. Taming VAEs. 2018. URL `http://arxiv.org/abs/1810.00597`.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning(ICML)*, pages 1278–1286. PMLR, jun 2014. URL `http://proceedings.mlr.press/v32/rezende14.html`.

Michal Rolínek, Dominik Zietlow, and Georg Martius. Variational autoencoders pursue PCA directions (by accident). In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Robert Sicks, Ralf Korn, and Stefanie Schwaar. A lower bound for the ELBO of the Bernoulli variational autoencoder. Technical report, mar 2020. URL `http://arxiv.org/abs/2003.11830`.

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems(NeurIPS)*, volume 29, pages 3738–3746, 2016.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, aug 1999. ISSN 1369-7412. doi: 10.1111/1467-9868.00196. URL `http://doi.wiley.com/10.1111/1467-9868.00196`.

Joong-ho Won. Proximity operator of the matrix perspective function and its applications. In *Advances in Neural Information Processing Systems(NeurIPS)*, 2020.

Mario V Wüthrich. From generalized linear models to neural networks, and back. Technical report, 2020. URL `https://ssrn.com/abstract=3491790`.

Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, Jie Chen, Zhaogang Wang, and Honglin Qiao. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In *World Wide Web Conference*, page 10. ACM, 2018. ISBN 9781450356398. doi: 10.1145/3178876.3185996. URL `https://doi.org/10.1145/3178876.3185996`.

Serena Yeung, Anitha Kannan, Yann Dauphin, and Li Fei-Fei. Tackling over-pruning in variational autoencoders. jun 2017. URL `http://arxiv.org/abs/1706.03643`.

Baichuan Yuan, Xiaowei Wang, Jianxin Ma, Chang Zhou, Andrea L. Bertozzi, and Hongxia Yang. Variational autoencoders for highly multivariate spatial point processes intensities. In *International Conference on Learning Representations(ICLR)*, 2020.

Yi Zhou and Yingbin Liang. Critical points of neural networks: Analytical forms and landscape properties. In *International Conference on Learning Representations(ICLR)*, 2018.