Appendix

A Network Architecture and Training

Network architectures are given in Tab. 3 and largely follow the architecture in Ghosh et al. (2019). For consistency, all the models share the same encoder/decoder structure. The full covariance representation can be realized by predicting n-dimensional standard deviations $\boldsymbol{\sigma}_\phi$ as well as $n(n-1)/2$-dimensional correlation factors $\boldsymbol{r}_\phi$ (followed by a $\tanh$ projection into the valid $[-1, 1]$ range), and building the lower triangular covariance matrix[5] $\boldsymbol{L}_\phi$. This way, one can ensure the symmetry and positive semi-definiteness of the full covariance matrix, $\boldsymbol{\Sigma}_\phi = \boldsymbol{L}_\phi \boldsymbol{L}_\phi^T$.

VAE: $\mathbf{x}_{C\times W\times H} \rightarrow \textsc{Encoder} \rightarrow \{\text{FC}_{1024\times n} : \boldsymbol{\mu}_\phi, \text{FC}_{1024\times n} : \log \boldsymbol{\sigma}_\phi^2\} \rightarrow \mathbf{z} \rightarrow \textsc{Decoder} \rightarrow \hat{\mathbf{x}}$
RAE: $\mathbf{x}_{C\times W\times H} \rightarrow \textsc{Encoder} \rightarrow \{\text{FC}_{1024\times n} : \mathbf{z}_\phi\} \rightarrow \textsc{Decoder} \rightarrow \hat{\mathbf{x}}$
UAE: $\mathbf{x}_{C\times W\times H} \rightarrow \textsc{Encoder} \rightarrow \{\text{FC}_{1024\times n} : \boldsymbol{\mu}_\phi, \text{FC}_{1024\times n} : \log \boldsymbol{\sigma}_\phi^2, \text{FC}_{1024\times n(n+1)/2} : \boldsymbol{r}_\phi\} \rightarrow \mathbf{z} \rightarrow \textsc{Decoder} \rightarrow \hat{\mathbf{x}}$
$\textsc{Encoder}$: $\textsc{Conv}_{32\times 64} \rightarrow \textsc{Conv}_{64\times 128} \rightarrow \textsc{Conv}_{128\times 256} \rightarrow \textsc{Conv}_{256\times 512} \rightarrow \textsc{Conv}_{512\times 1024} \rightarrow \textsc{Flatten}$
$\textsc{Decoder}$: $\text{FC}_{n\times 1024\cdot 8\cdot 8} \rightarrow \text{T}\textsc{Conv}_{1024\times 512} \rightarrow \text{T}\textsc{Conv}_{512\times 256}[\rightarrow \text{T}\textsc{Conv}_{256\times 128}]^{\text{CelebA}} \rightarrow \text{T}\textsc{Conv}_{256\,\text{or}\,128\times C}$
MNIST: $C = 1, W = H = 32, n = 16$ CIFAR10: $C = 3, W = H = 32, n = 128$ CelebA: $C = 3, W = H = 64, n = 64$

Table 3: Network architectures of the implemented VAE, RAE, and UAE models. Batch dimensions omitted for clarity. All the encoder 2D convolution blocks contain 3×3 kernels, stride 2, and padding 1, followed by a 2D batch normalization and a Leaky-ReLU activation. The decoder transpose convolutions share the same parameters as the encoder convolutions apart from using a 4×4 kernel. The last transpose convolution (mapping to channel dimension) however has a 3×3 kernel and is followed by a $\tanh$ activation instead (without batch normalization).

The dataset preprocessing procedure is the following. The Fashion-MNIST images are scaled from 28×28 to 32×32. For the training dataset, we use 50k out of the 60k provided examples, leaving the remaining 10k for the validation dataset. For the test dataset, we use the provided examples. In CIFAR10, we perform a size-4 padding, a random 32×32 crop, and a random horizontal flip on the training data, followed by a normalization for all the dataset subsets. We use the same training/validation/test split method as in Fashion-MNIST. In CelebA, we perform a 148×148 center crop and resize the images to 64×64. We use the provided training/validation/testing subsets.

All the models are implemented in PyTorch (Paszke et al., 2019) (source code available upon request) and use the library provided in Seitzer (2020) for FID computation. The models are trained for 100 epochs, starting with a 0.005 learning rate, cut in half after every five epochs without improvement. The weights used in the loss functions are the following: KL-divergence (or the Wasserstein metric) terms are weighted with $\beta = 2.5e^{-4}$ in the case of VAE and UAE and $\beta = 1e^{-4}$ for the RAE. The decoder regularization terms are weighted with $\gamma = 1e^{-6}$ for both RAE and UAE. We performed minimal hyperparameter search over the weights.

In computing the FID scores, we follow the same procedure as in Ghosh et al. (2019). In the three cases of reconstruction, sampling, and interpolation, we evaluate the FID to the test set image reconstructions as the ground-truth. In the reconstruction metric, we use the validation image reconstructions. In sampling, we fit the training dataset latent features to a GMM (see Sec. 5.1) and sample and reconstruct the same number of elements as in the test set. In interpolation, we apply mid-point spherical interpolation between a random pair of validation set embeddings and use the reconstructions of the same number of samples as in the validation set.

The network architectures largely follow the structure adopted by Ghosh et al. (2019), with the difference of the added first two encoder layers. Nevertheless, in Tab. 2, we did not manage to reproduce the FID values reported in Ghosh et al. (2019) on CelebA and CIFAR10, even observing that removing the first two encoder layers reduces the overall performance. We suspect that it is

[5] In the 3-dimensional case: $\boldsymbol{L}_\phi = [\sigma_1 \quad 0 \quad 0; \quad r_1\sigma_2\sigma_1 \quad \sigma_2 \quad 0; \quad r_2\sigma_3\sigma_1 \quad r_3\sigma_3\sigma_2 \quad \sigma_3]$

due to the differing Tensorflow and PyTorch model implementations as well as the FID computation libraries. However, in most cases, our implementation of the RAE attains a larger performance gain over the VAE than reported in Ghosh et al. (2019).

B Gradient Variance

In the following, we present the trivial argument that a VAE-like model with deterministic posterior sampling (such as the unscented transform in the UAE) achieves lower variance in training than a random-sampling VAE model. For simplicity, we observe the single-dimensional case that can be generalized to multiple dimensions. We compare on the one hand, a VAE reconstruction loss with random sampling from the standard normal prior $\mathcal{N}$ and the reconstruction loss of a VAE-like model where we sample from a discrete uniform distribution χ of K points $\{\chi_k\}_{k=1}^{K}$ (for example the sigma points in Eq. (5)) instead. The losses are given as

$$\mathcal{L}_\chi = \|D_\theta(\mu + \sigma\epsilon) - x\|^2, \quad \epsilon \sim \chi \tag{20}$$

$$\mathcal{L}_\mathcal{N} = \|D_\theta(\mu + \sigma\epsilon) - x\|^2, \quad \epsilon \sim \mathcal{N}\ . \tag{21}$$

For a single sample ϵ_j (from the $\mathcal{N}$ or χ) and a single training example denoted by x_i, the gradients of the reconstruction losses w.r.t. the parameterization θ are given as

$$\nabla_\theta \mathcal{L}_\chi^{i,j} = \nabla_\theta \|D_\theta(\mu_i + \sigma_i\epsilon_j|x_i) - x_i\|^2, \quad \epsilon_j \sim \chi \tag{22}$$

$$\nabla_\theta \mathcal{L}_\mathcal{N}^{i,j} = \nabla_\theta \|D_\theta(\mu_i + \sigma_i\epsilon_j|x_i) - x_i\|^2, \quad \epsilon_j \sim \mathcal{N}. \tag{23}$$

The variance of the given gradient can be computed under the expectation of the data distribution

$$\mathrm{Var}(\nabla_\theta \mathcal{L}_\chi^{j}) = E_{x_i}\|\nabla_\theta \mathcal{L}_\chi^{i,j} - \nabla_\theta \mathcal{L}_\chi^{j}\|_2^2 \tag{24}$$

$$\mathrm{Var}(\nabla_\theta \mathcal{L}_\mathcal{N}^{j}) = E_{x_i}\|\nabla_\theta \mathcal{L}_\mathcal{N}^{i,j} - \nabla_\theta \mathcal{L}_\mathcal{N}^{j}\|_2^2\ , \tag{25}$$

for a given sample ϵ_j, where $\nabla_\theta \mathcal{L}_\chi^j$ and $\nabla_\theta \mathcal{L}_\mathcal{N}^j$ are the 'true' gradients. For clarity, we denote $\nabla_\theta \mathcal{L}_\chi = \Delta_\chi$ and $\nabla_\theta \mathcal{L}_\mathcal{N} = \Delta_\mathcal{N}$ and compare the given gradients while assuming that the variance of Δ_χ^j is lower

$$\mathrm{Var}(\Delta_\chi^j) \leq \mathrm{Var}(\Delta_\mathcal{N}^j) \tag{26}$$

The variances are random variables w.r.t. ϵ_j and can be generalized into an expectation

$$E_{\epsilon\sim\chi}[\mathrm{Var}(\Delta_\chi)] \leq E_{\epsilon\sim\mathcal{N}}[\mathrm{Var}(\Delta_\mathcal{N})] \tag{27}$$

Here, the discrete distribution χ can be approximated by a mixture of Diracs or very narrow Gaussian kernels. Therefore, an importance sampling substitution can be used to replace the χ sampling with the normal distribution

$$E_{\epsilon\sim\mathcal{N}}[\frac{\chi(\epsilon)}{\mathcal{N}(\epsilon)}\mathrm{Var}(\Delta_\mathcal{N})] \leq E_{\epsilon\sim\mathcal{N}}[\mathrm{Var}(\Delta_\mathcal{N})]\ . \tag{28}$$

This is well-defined since the approximated χ distribution is contained within the support of $\mathcal{N}$. Then, the common sampling distribution allows to confirm the initial assumption through the following relationship

$$E_{\epsilon\sim\mathcal{N}}[(\frac{\chi(\epsilon)}{\mathcal{N}(\epsilon)} - 1)\mathrm{Var}(\Delta_\mathcal{N})] \leq 0\ . \tag{29}$$

In practice, the expectation is approximated by a single or few samples from $\mathcal{N}$. Thus, the density $\chi(\epsilon)$ is zero (in the case of a Dirac approximation, or near-zero in the narrow kernel case), rendering the variance-weighting term negative and consequently the entire left side of the equation non-positive (due to the by-definition non-negativity of the variance). This allows to conclude that the variance of a deterministic sampling reconstruction loss gradient is lower than the random sampling gradient.

We provide additional, empirical reasoning to the lower-gradient-variance argument. Fig. 3 shows the isolated effects of the unscented transform on the VAE training (not considering the full UAE model) through the infinity norm of the training gradient. Apart from observing fewer peaks in the values, the lower norm of the training gradient in practice contributes to an overall lower gradient variance, simply through the lower values.

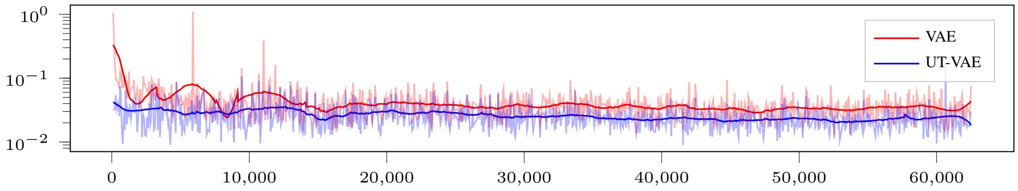

Figure 3: Comparison of the infinity norm of the training gradients (logarithm scale) for the VAE and UT-VAE across approx. 60k training steps (100 epochs) on the CIFAR10 dataset. See Tab. 1 for the loss function definitions. In both instances, a single sample or sigma point is taken. The UT-VAE incorporates lower-variance posterior sampling than the reparameterization trick; thus, simply sampling at the sigma points contributes to lower values and fewer peaks of the gradient norm.

C ELBO CONSTRAINT DERIVATION

In this section, we complete the derivation of the constraint in Eq. (13) to the reformulated version in Eq. (14). The constraint in Eq. (13) can be bounded by the maximum of the decoder output in a single dimension i, multiplied by the number of dimensions

$$\|D_\theta(\mathbf{z}_1) - D_\theta(\mathbf{z}_2)\|_p \leq \dim(\mathbf{x}) \cdot \sup_i\{\|d_i(\mathbf{z}_1) - d_i(\mathbf{z}_2)\|_p\} < \epsilon \; . \tag{30}$$

Using the mean value theorem, the term $\sup_i\{\|d_i(\mathbf{z}_1) - d_i(\mathbf{z}_2)\|_p\}$ can be reduced to

$$\sup_i\{\|\nabla_t d_i((1-t)\mathbf{z}_1 + t\mathbf{z}_2)\|_p \cdot \|\mathbf{z}_1 - \mathbf{z}_2\|_p\} < \epsilon \; , \tag{31}$$

Since $\mathbf{z}_1$ and $\mathbf{z}_2$ are arbitrary, the first part can be simplified and generalized over all dimensions while separating the overall product using the Cauchy-Schwarz inequality

$$\sup_i\{\|\nabla_{\mathbf{z}} d_i(\mathbf{z})\|_p \cdot \|\mathbf{z}_1 - \mathbf{z}_2\|_p\} < \epsilon \tag{32}$$

$$\sup\{\|\nabla_{\mathbf{z}} D_\theta(\mathbf{z})\|_p\} \cdot \sup\{\|\mathbf{z}_1 - \mathbf{z}_2\|_p\} < \epsilon \; , \tag{33}$$

obtaining the form in Eq. (14).

D MULTI-SIGMA TRAINING

The UAE loss function defined in Tab. 1 assumes a single sigma point sample in the reconstruction term and the decoder gradient. Similarly to the multi-sample case in the VAE, multiple sigma points (up to $2n+1$) chosen from the set $\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi, \boldsymbol{\Sigma}_\phi)\}_{i=0}^{2n}$ can be incorporated. One strategy is expanding the reconstruction term into an expectation $\mathbb{E}_{\mathbf{z}_i \sim \{\boldsymbol{\chi}_i\}}\|\mathbf{x} - D_\theta(\mathbf{z}_i)\|_2^2$ and computing the decoder gradient term for the sampled sigma points. An alternative is training on a single randomly chosen sigma point but replicating the training examples. Multiple training examples were used in Burda et al. (2016), where importance-weighted posterior samples (obtained via the reparameterization trick) yield a tighter lower bound.

Instead of choosing the points randomly and with equal probability, different sampling strategies can be utilized. For example, only pairs of sigma points along an axis can be chosen, conveying the width of the posterior distribution in the given dimension. This is illustrated in Fig. 1b for an ellipsoid with three pairs and seven sigma points in total.

Sampling sigma points is trivially lower variance than the random sampling of the VAE reparameterization trick, see Appendix B. However, it is biased and yields non-independent samples. Nevertheless, in practice we have observed that the multi-sigma UAE models clearly outperform the single-sigma one, evidenced by the results in Tab. 4.

	Fashion-MNIST			CIFAR10			CelebA		
	Rec.	Sample	Interp.	Rec.	Sample	Interp.	Rec.	Sample	Interp.
$^{1x}\sim$VAE	47.38	51.74	66.04	160.05	173.45	170.33	65.86	67.66	68.08
$^{2x}\sim$VAE	43.36	48.47	62.57	151.13	165.60	162.60	63.46	65.46	65.98
$^{4x}\sim$VAE	42.30	49.50	64.09	146.21	161.17	158.19	62.19	64.36	64.58
$^{8x}\sim$VAE	39.00	46.81	62.16	141.84	158.14	154.65	60.49	62.89	62.93
$^{1x}\sim$VAE*	31.90	39.61	55.60	136.90	156.83	151.41	45.15	50.29	53.23
$^{2x}\sim$VAE*	30.43	44.43	60.22	119.46	146.18	139.28	43.52	48.21	54.29
$^{4x}\sim$VAE*	28.89	43.16	65.13	115.04	141.73	134.61	38.75	44.47	54.24
$^{8x}\sim$VAE*	28.39	44.82	72.56	101.76	136.73	127.44	38.02	44.90	55.69
$^{1x}\sim$UAE	33.30	40.81	60.13	120.95	147.07	137.39	37.93	44.59	46.45
$^{2x}\sim$UAE	32.15	40.48	59.68	113.46	141.34	130.99	37.55	44.35	48.41
$^{4x}\sim$UAE	29.31	40.35	63.57	107.42	137.41	127.12	35.45	42.33	46.79
$^{8x}\sim$UAE	27.44	42.36	74.42	94.17	132.62	116.68	34.36	41.22	45.04

Table 4: Comparison of multi-sample models: in all the cases, training examples are replicated, which results in multiple prior samples or sigma points to be utilized on the same training example.

E Ablation Study of the loss components

This section provides an additional ablation study of the loss components used in the UAE model. The loss functions considered are provided in Tab 5 and the obtained results are in Tab. 6. There are four dimensions along which the results can be interpreted: Wasserstein metric, unscented transform, full covariance representation, and the decoder regularization (gradient penalty).

Tab. 6 is divided into two parts: the top part models use the analytical form of the KL divergence (Eq. 9) while the bottom part use the Frobenius norm mismatch derived from the Wasserstein metric (Eq. 10). It is clearly visible that the latter models strongly outperform the former, in all datasets and configurations. The loss function allows for a sharper posterior and thus larger expressiveness of the model (see Appendix F).

Considering the unscented transform models, it is interesting to note that applying it in the case of a diagonal posterior and standard KL divergence loss (UT-VAE row) can lead to large regressions in the sampling and interpolation metrics (as seen in Fashion-MNIST and CelebA while CIFAR10 seems to be more robust). It seems to have a detrimental effect on the structure of the latent space under assumptions of orthogonality. However, the VAE* can benefit from the unscented transform sampling, evidenced by the CIFAR10 metrics.

Considering the full covariance models, interesting interplays can be noticed. Overall, modeling the full covariance posterior in the context of the analytical KL divergence loss is not beneficial, as evidenced in the VAE-full $\mathbf{\Sigma}_\phi$ row. However, in the VAE* case, it appears that the Wasserstein metric enables the model to utilize correlations in a stable way (VAE*-full $\mathbf{\Sigma}_\phi$ row). This is evidenced by the improvements on CIFAR10 and CelebA while Fashion-MNIST generally appears to not benefit from the full covariance. As mentioned in Sec. 6, we assume the lower-dimensional input space to be the cause. Then, applying the unscented transform in conjunction with the full covariance model brings performance improvements on CIFAR10, both in the case of UT-VAE-full $\mathbf{\Sigma}_\phi$ vs. VAE-full $\mathbf{\Sigma}_\phi$ as well as UT-VAE*-full $\mathbf{\Sigma}_\phi$ vs. VAE*-full $\mathbf{\Sigma}_\phi$, with mixed results on the other datasets.

Finally, it appears that the decoder regularization helps the unscented transform models achieve a much more structured latent space, both in the KL divergence and Wasserstein metric cases, and both for the diagonal and full covariance posteriors. Without the unscented transform, the decoder regularization is beneficial in the VAE* case but not in the KL divergence case.

F Aggregated Posterior Visualization

In Fig. 4 we present detailed plots on the posterior distributions of VAE and VAE* for the first 16 dimensions. The VAE clearly shows signs of posterior collapse; we have observed that more than

	Loss function	Posterior sampling
$\mathcal{L}_{\text{VAE}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\sum_i\sigma_{\phi,i}^2-2\log\sigma_{\phi,i}$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{\sigma}_\phi\odot\boldsymbol{\epsilon},\ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{UT-VAE}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\sum_i\sigma_{\phi,i}^2-2\log\sigma_{\phi,i}$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\text{diag}(\boldsymbol{\sigma}_\phi^2))\}_{i=0}^{2n}$
$\mathcal{L}_{\text{UT-VAE-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\sum_i\sigma^2\phi{,}i-2\log\sigma_{\phi,i}+\max(\boldsymbol{\sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\text{diag}(\boldsymbol{\sigma}_\phi^2))\}_{i=0}^{2n}$
$\mathcal{L}_{\text{VAE-full}\,\boldsymbol{\Sigma}_\phi}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\text{tr}(\boldsymbol{\Sigma}_\phi)-2\text{tr}(\log\boldsymbol{L}_\phi)$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{L}_\phi\boldsymbol{\epsilon},\ \ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{VAE-full}\,\boldsymbol{\Sigma}_\phi\text{-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\text{tr}(\boldsymbol{\Sigma}_\phi)-2\text{tr}(\log\boldsymbol{L}_\phi)+\lambda_{\max}(\boldsymbol{\Sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{L}_\phi\boldsymbol{\epsilon},\ \ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{UT-VAE-full}\,\boldsymbol{\Sigma}_\phi}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\text{tr}(\boldsymbol{\Sigma}_\phi)-2\text{tr}(\log\boldsymbol{L}_\phi)$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\boldsymbol{\Sigma}_\phi)\}_{i=0}^{2n}$
$\mathcal{L}_{\text{UT-VAE-full}\,\boldsymbol{\Sigma}_\phi\text{-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\text{tr}(\boldsymbol{\Sigma}_\phi)-2\text{tr}(\log\boldsymbol{L}_\phi)+\lambda_{\max}(\boldsymbol{\Sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\boldsymbol{\Sigma}_\phi)\}_{i=0}^{2n}$
$\mathcal{L}_{\text{VAE*}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\text{diag}(\boldsymbol{\sigma}_\phi^2)-\mathbf{I}\rVert_F^2$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{\sigma}_\phi\odot\boldsymbol{\epsilon},\ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{UT-VAE*}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\text{diag}(\boldsymbol{\sigma}_\phi^2)-\mathbf{I}\rVert_F^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\text{diag}(\boldsymbol{\sigma}_\phi^2))\}_{i=0}^{2n}$
$\mathcal{L}_{\text{UT-VAE*-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\text{diag}(\boldsymbol{\sigma}_\phi^2)-\mathbf{I}\rVert_F^2+\max(\boldsymbol{\sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\text{diag}(\boldsymbol{\sigma}_\phi^2))\}_{i=0}^{2n}$
$\mathcal{L}_{\text{VAE*-full}\,\boldsymbol{\Sigma}_\phi}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\boldsymbol{L}_\phi-\mathbf{I}\rVert_F^2$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{L}_\phi\boldsymbol{\epsilon},\ \ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{VAE*-full}\,\boldsymbol{\Sigma}_\phi\text{-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\boldsymbol{L}_\phi-\mathbf{I}\rVert_F^2+\lambda_{\max}(\boldsymbol{\Sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}=\boldsymbol{\mu}_\phi+\boldsymbol{L}_\phi\boldsymbol{\epsilon},\ \ \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$
$\mathcal{L}_{\text{UT-VAE*-full}\,\boldsymbol{\Sigma}_\phi}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\boldsymbol{L}_\phi-\mathbf{I}\rVert_F^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\boldsymbol{\Sigma}_\phi)\}_{i=0}^{2n}$
$\mathcal{L}_{\text{UT-VAE*-full}\,\boldsymbol{\Sigma}_\phi\text{-GP}}$	$\lVert\mathbf{x}-D_\theta(\mathbf{z})\rVert_2^2+\lVert\boldsymbol{\mu}_\phi\rVert_2^2+\lVert\boldsymbol{L}_\phi-\mathbf{I}\rVert_F^2+\lambda_{\max}(\boldsymbol{\Sigma}_\phi)\lVert\nabla_{\mathbf{z}}D_\theta(\mathbf{z})\rVert_2^2$	$\mathbf{z}\sim\{\boldsymbol{\chi}_i(\boldsymbol{\mu}_\phi,\boldsymbol{\Sigma}_\phi)\}_{i=0}^{2n}$

Table 5: The loss functions used for the models in Tab. 6.

	Fashion-MNIST			CIFAR10			CelebA		
	Rec.	Sample	Interp.	Rec.	Sample	Interp.	Rec.	Sample	Interp.
VAE	47.38	51.74	66.04	160.05	173.45	170.33	65.86	67.66	68.08
UT-VAE	47.48	74.67	69.99	155.72	175.89	166.37	58.09	250.2	122.5
UT-VAE-GP	50.98	74.57	71.19	153.23	172.40	163.82	54.68	215.6	92.81
VAE-full $\boldsymbol{\Sigma}_\phi$	59.21	63.73	72.97	181.56	189.49	184.59	87.38	88.09	87.32
VAE-full $\boldsymbol{\Sigma}_\phi$-GP	60.51	64.76	73.54	189.27	199.04	196.23	117.9	189.8	166.0
UT-VAE-full $\boldsymbol{\Sigma}_\phi$	82.40	134.7	139.7	174.11	187.37	179.07	92.39	267.4	169.1
UT-VAE-full $\boldsymbol{\Sigma}_\phi$-GP	70.61	107.0	96.36	153.62	179.43	173.14	98.50	250.1	154.4
VAE*	33.72	40.39	59.76	136.90	156.83	151.41	45.15	50.29	53.23
UT-VAE*	33.57	40.31	57.21	134.21	153.09	147.32	48.29	56.14	54.14
UT-VAE* -GP	32.46	39.88	59.65	131.28	151.29	145.11	40.48	47.63	50.93
VAE*-full $\boldsymbol{\Sigma}_\phi$	36.23	42.73	62.95	133.98	153.15	148.04	42.26	49.12	62.58
VAE*-full $\boldsymbol{\Sigma}_\phi$-GP	35.41	41.97	61.10	125.65	148.35	139.16	40.08	47.40	50.58
UT-VAE*-full $\boldsymbol{\Sigma}_\phi$	35.78	42.97	66.75	126.00	149.52	141.67	43.00	53.39	51.22
UT-VAE*-full $\boldsymbol{\Sigma}_\phi$-GP	33.30	40.81	60.13	120.95	147.07	137.39	37.93	44.59	46.45

Table 6: Full ablation study of the models between the VAE and UAE (in the UT-VAE*-full $\boldsymbol{\Sigma}_\phi$-GP row), using the Wasserstein metric denoted by *, unscented transform (UT), full covariance matrix, and the decoder gradient penalty (GP) components.

half of the 128 dimensions are nearly equal to the prior. This considerably hurts the generative power of the VAE model. In contrast, the VAE* model has very low variance in all dimensions, which reflects a nearly deterministic encoder at the end of the training.

G Qualitative Results

Qualitative results on Fashion-MNIST and CIFAR10 are provided in Fig. 5 and Fig. 6. The same setup as in Fig. 2 is employed. It can be seen that the CIFAR10 images appear considerably richer and sharper, consistent with the results in Tab. 2 and Tab. 4.

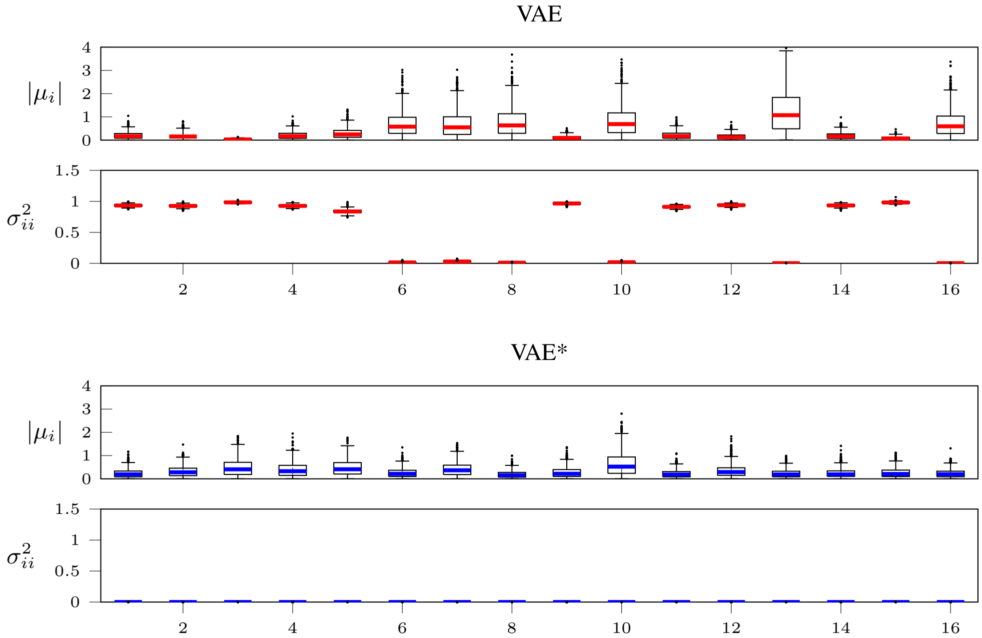

Figure 4: Comparison of the distribution of absolute means and variances of 1000 posterior samples for the VAE and the VAE* models on the CIFAR10 dataset. Top rows show the absolute means and the lower rows the variances of the first 16 dimensions. For the VAE* all the means differ from zero while the variances are close to zero, whereas for the VAE, 10 of 16 dimensions are effectively deactivated.

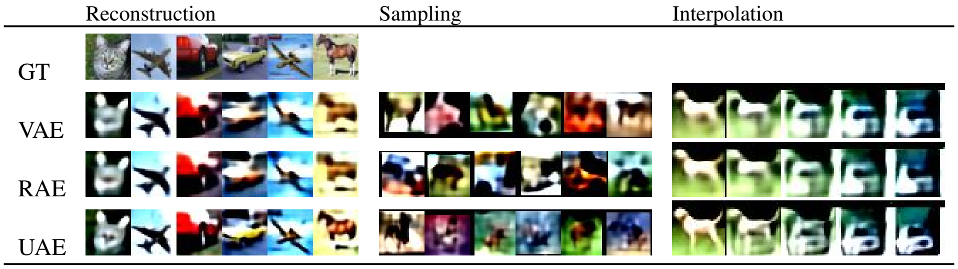

Figure 5: Qualitative results on CIFAR10 dataset.

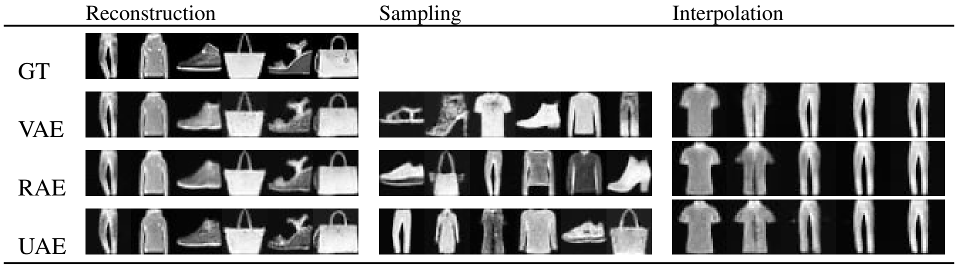

Figure 6: Qualitative results on Fashion-MNIST dataset.