Conditional Flow Variational Autoencoders for Structured Sequence Prediction

Apratim Bhattacharyya[1], **Michael Hanselmann**[2], **Mario Fritz**[3], **Bernt Schiele**[1], and **Christoph-Nikolas Straehle**[2]

[1]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[2]Bosch Center for Artificial Intelligence, Renningen, Germany
[3]CISPA Helmholtz Center for Information Security, Saarland Informatics Campus, Saarbrücken, Germany

Abstract

Prediction of future states of the environment and interacting agents is a key competence required for autonomous agents to operate successfully in the real world. Prior work for structured sequence prediction based on latent variable models tend to imposes simple and often uni-modal standard Gaussian prior on the latent variables. This induces a strong model bias which makes it challenging to fully capture the multi-modality of the distribution of the future states. In this work, we introduce *Conditional Flow Variational Autoencoders (CF-VAE)* using our novel conditional normalizing flow based prior and demonstrate state of the art results on two multi-modal structured sequence prediction tasks.

1 Introduction

Anticipating future states of the environment is a key competence necessary for the success of autonomous agents. In complex real world environments, the future is highly uncertain. Therefore, structured predictions, one to many mappings (Sohn et al., 2015; Bhattacharyya et al., 2018) of the likely future states of the world, are important. In many scenarios, these tasks can be cast as sequence prediction problems. Particularly, conditional variational autoencoders (CVAE) (Sohn et al., 2015) have been successful for such problems (Lee et al., 2017; Bhattacharyya et al., 2018; Pajouheshgar and Lampert, 2018; Babaeizadeh et al., 2018). CVAEs model diverse futures by factorizing the distribution of future states using a set of latent variables which are mapped to likely future states. However, CVAEs assume a standard Gaussian prior on the latent variables which induces a strong model bias (Hoffman and Johnson, 2016; Tomczak and Welling, 2018) and makes it difficult for the model to capture multi-modal distributions.

Recent work (Tomczak and Welling, 2018; Wang et al., 2017; Gu et al., 2018) has therefore focused on more expressive Gaussian mixture based priors. However, Gaussian mixtures still have limited expressivity and optimization suffers from complications e.g. determining the number of mixture components. In contrast, normalizing flows are more expressive and enable the modelling of complex multi-modal priors. Recent work on flow based priors (Chen et al., 2017; Ziegler and Rush, 2019), have focused only on the unconditional (plain VAE) case. However, this not sufficient for CVAEs because in the conditional case the complexity of the distributions are highly dependent on the condition. In this work, in order to model complex multi-modal conditional distributions, we propose *Conditional Flow Variational Autoencoders (CF-VAE)* with novel conditional normalizing flow based priors. Furthermore, we propose a novel regularization scheme that stabilizes training and prevents degenerate solutions during optimization of the evidence lower bound. Finally, we show that our regularized CF-VAE outperforms the state of the art on two important structured sequence prediction tasks – handwriting stroke prediction on MNIST and traffic participant prediction on Stanford Drone.

4th workshop on Bayesian Deep Learning (NeurIPS 2019), Vancouver, Canada.

2 Conditional Flow Variational Autoencoder

Our Conditional Flow Variational Autoencoder is based on the conditional variational autoencoder (Sohn et al., 2015) which models conditional data distributions $p_\theta(\mathrm{y}|\mathrm{x})$ with a prior latent variable distribution $p(\mathbf{z}|\mathrm{x})$. The posterior distribution of latent variables $q_\phi(\mathrm{z}|\mathrm{x})$ is learnt using amortized variational inference.The ELBO is maximized, given by,

$$\log(p_\theta(\mathrm{y}|\mathrm{x})) \geq \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})} \log(p_\theta(\mathrm{y}|\mathrm{z},\mathrm{x})) - D_{\mathrm{KL}}(q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})||p(\mathrm{z}|\mathrm{x})). \tag{1}$$

In practice, a simple unconditional standard Gaussian prior $\mathcal{N}(0, I)$ is used (Sohn et al., 2015). Although in theory a strong enough encoder/decoder pair should be able to perfectly encode and decode from a unit Gaussian, in practice this is difficult to achieve. On complex conditional multi-modal data, simple Gaussian priors have been shown to induce strong model bias resulting in missing modes (Tomczak and Welling, 2018; Ziegler and Rush, 2019). Moreover, the ground truth conditional distribution $p(\mathrm{y}|\mathrm{x})$ can differ considerably depending upon the data point x – a unconditional prior $p(\mathrm{z})$ leaves the burden of learning the dependence on the condition completely on the decoder.

Conditional Normalizing Flows. Recently, normalizing flow Tabak et al. (2010); Dinh et al. (2015) based priors for VAEs have been proposed (Chen et al., 2017; Ziegler and Rush, 2019). However, these flow based priors are unconditional. Here we propose conditional priors, through the use of conditional normalizing flows. Our conditional normalizing flow based prior starts with a simple base distribution $p(\epsilon|\mathrm{x})$, which is then transformed by n layers of invertible normalizing flows f_i, with parameters ψ, to a more complex prior distribution (dependent on number of layers n) on the latent variables $p_\psi(\mathrm{z}|\mathrm{x})$,

$$\epsilon|\mathrm{x} \xleftrightarrow{f_1} \mathrm{h}_1|\mathrm{x} \xleftrightarrow{f_2} \mathrm{h}_2|\mathrm{x} \cdots \xleftrightarrow{f_n} \mathrm{z}|\mathrm{x}. \tag{2}$$

Given the base density $p(\epsilon|\mathrm{x})$ and the Jacobian J_i of each layer i of the transformation, the log-likelihood of the latent variable z can be expressed using the change of variables formula,

$$\log(p_\psi(\mathrm{z}|\mathrm{x})) = \log(p(\epsilon|\mathrm{x})) + \sum_{i=1}^{n} \log(|\det J_i|). \tag{3}$$

We consider simple spherical Gaussians as base distributions, $p(\epsilon|\mathrm{x}) = \mathcal{N}(0, I)$ for efficient sampling. In contrast to prior work on conditional normalizing flows (Lu and Huang, 2019; Atanov et al., 2019; Ardizzone et al., 2019) which use affine flows, we build upon (Ziegler and Rush, 2019) and introduce conditional non-linear normalizing flows. Conditioning is achieved by making each f_i a non-linear function of the condition x (more details in Appendix A). Next, we derive the form of ELBO with our conditional normalizing flow based prior using the change of variables formula (3) to easily compute the KL divergence. The ELBO can be expressed as, (full derivation in Appendix A)

$$\log(p_\theta(\mathrm{y}|\mathrm{x})) \geq \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})} \log(p_\theta(\mathrm{y}|\mathrm{z},\mathrm{x})) + \mathcal{H}(q_\phi) + \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})} \log(p(\epsilon|\mathrm{x})) + \sum_{i=1}^{n} \log(|\det J_i|) \tag{4}$$

To learn complex conditional priors, we jointly optimize both the variational posterior distribution $q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})$ and the conditional prior $p_\psi(\mathrm{z}|\mathrm{x})$ in (4) (akin to (Tomczak and Welling, 2018)). The variational posterior tries to match the conditional prior and vice-versa so that the ELBO (4) is maximized. This is our *Conditional Flow Variational Autoencoder (CF-VAE)*. Next, we discuss instabilities during training which leads to degenerate solutions and our novel regularization scheme.

Regularizing Posteriors for Stability. In (4), the entropy and the log-Jacobian of the joint objective are at odds with each other. The log-Jacobian favours the contraction of the base density. Therefore, log-Jacobian at the right of (4) is maximized when the conditional flow maps the base distribution to a low entropy conditional prior (and thus a low entropy variational distribution $q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})$). Ideally, the CF-VAE should learn to balance these terms. However, in practice we observe instabilities during training. Degenerate solutions emerge where either the entropy or the log-Jacobian terms dominate and the data log-likelihood is fully or partially ignored. Therefore, we regularize the posterior $q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})$ by fixing the variance to C. This leads to a constant entropy term which in turn bounds the maximum possible amount of contraction, thus upper bounding the log-Jacobian. Therefore, during training this encourages our model to concentrate on explaining the data. Note that, although $q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})$ has fixed variance, this does not significantly effect expressivity as the marginal $q_\phi(\mathrm{z}|\mathrm{x})$ can be arbitrarily complex due to our conditional flow prior. Moreover, we observe that the LSTM based decoders employed demonstrate robust performance across a wide range of values $\mathrm{C} = [0.05, 0.25]$.

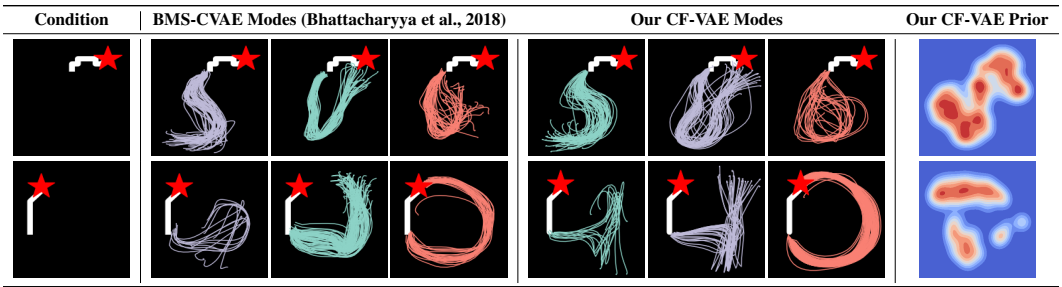

Figure 1: Random samples clustered using k-means. The number of clusters is set manually to the number of expected digits. The corresponding priors of our CF-VAE on the right. Note, our 64D CF-VAE latent distribution is (approximately) projected to 2D using tSNE and KDE.

3 Experiments

We evaluate our CF-VAE on two multi-modal sequence prediction datasets. In line with prior work (Lee et al., 2017; Pajouheshgar and Lampert, 2018), we use the negative conditional log-likelihood (-CLL) metric and the mean Euclidean distances of the oracle Top $k\%$ of K predictions. We include a detailed analysis of this metric in the Appendix F. We use a conditional flow architecture with 16 layers of conditional non-linear flows. Increasing the number of conditional non-linear flows generally led to "over-fitting" on the training latent distribution.

MNIST Sequences. The MNIST Sequence dataset (D. De Jong, 2016) consists of sequences of handwriting strokes of the MNIST digits. The state-of-the-art approach is the Gaussian prior CVAE based "Best-of-Many"-CVAE (Bhattacharyya et al., 2018). We follow the evaluation protocol of Bhattacharyya et al. (2018) and predict the complete stroke given the first ten steps. Additionally, we compare with, 1. A standard CVAE with uni-modal Gaussian prior; 2. A CVAE with a data dependent conditional mixture of Gaussians (MoG) prior; 3. A CF-VAE without regularization – without a fixed variance posterior $q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})$; 4. A CF-VAE without the conditional non-linear flow layers (CF-VAE-*Affine*, replaced with affine flows (Lu and Huang, 2019; Atanov et al., 2019)). Although up to our knowledge, no prior work integrates MoG priors with CVAEs, we experiment with a conditional MoG prior for fairness (see Appendix D and E). We use the same model architecture (Bhattacharyya et al., 2018) across all baselines.

We report the results in Table 1. We see that our CF-VAE performs best. It has a performance advantage of over 20% against the state of the art BMS-CVAE. We further illustrate the modes captured and the learnt multi-modal conditional flow priors in Figure 1. In contrast, the BMS-CVAE is unable to fully capture all modes – its predictions are pushed to the mean due to the strong model bias induced by the Gaussian prior. The results improve considerably with the multi-modal MoG prior ($M = 3$ components work best). We also experiment with optimizing the standard CVAE architecture. This improves performance only slightly (increasing LSTM encoder/decoder units to 256 from 48). Moreover, our experiments with a conditional AAE/WAE (Gu et al., 2018) based baseline did not improve performance beyond the standard CVAE, because the discriminator based KL estimate in AAE/WAEs tends to be an underestimate (Rosca et al., 2019). This illustrates that in practice it is difficult to map highly multimodal sequences to a Gaussian prior and highlights the need of a data-dependent multi-modal priors. Our CF-VAE still significantly outperforms the MoG-CVAE as normalizing flows are better at learning complex multi-modal distributions (Kingma and Dhariwal, 2018). Next, we see that without regularization ($\mathrm{C} = 0.2$) there is a 40% drop in performance, highlighting the effectiveness of our novel regularization scheme. We also see that affine conditional flow based priors leads to a drop in performance (77.2 vs 74.9 CLL) illustrating the advantage of our non-linear conditional flows.

Table 1: Evaluation on MNIST Sequences (CLL: lower is better).

Method	-CLL
CVAE (Sohn et al., 2015)	96.4
BMS-CVAE (Bhattacharyya et al., 2018)	95.6
CVAE - *optimized architecture* (Ours)	94.5
MoG-CVAE, $M = 3$	84.6
CF-VAE - *no regularization* (Ours)	104.3
CF-VAE - *Affine*, *regularized* (Ours)	77.2
CF-VAE - *regularized*, $\mathrm{C} = 0.2$ (Ours)	**74.9**

Stanford Drone. The Stanford Drone dataset (Robicquet et al., 2016) consists of trajectories of traffic participant e.g. pedestrians, bicyclists, cars in videos captured from a drone. The scenes are

Table 2: Five fold cross validation on the Stanford Drone dataset. Euclidean error at ($^1/_5$) resolution.

Method	Visual	Error @ 1sec	Error @ 2sec	Error @ 3sec	Error @ 4sec	-CLL
"Shotgun" (Top 10%) (Pajouheshgar and Lampert, 2018)	None	0.7	1.7	3.0	4.5	91.6
DESIRE-SI-IT4 (Top 10%) (Lee et al., 2017)	RGB	1.2	2.3	3.4	5.3	x
STCNN (Top 10%) (Pajouheshgar and Lampert, 2018)	RGB	1.2	2.1	3.3	4.6	x
BMS-CVAE (Top 10%) (Bhattacharyya et al., 2018)	RGB	0.8	1.7	3.1	4.6	126.6
MoG-CVAE, $M = 3$ (Top 10%)	None	0.8	1.7	2.7	3.9	86.1
CF-VAE - *regularized* (Ours, Top 10%)	None	**0.7**	**1.5**	2.5	3.6	84.6
CF-VAE - *regularized* (Ours, Top 10%)	RGB	**0.7**	**1.5**	**2.4**	**3.5**	**84.1**

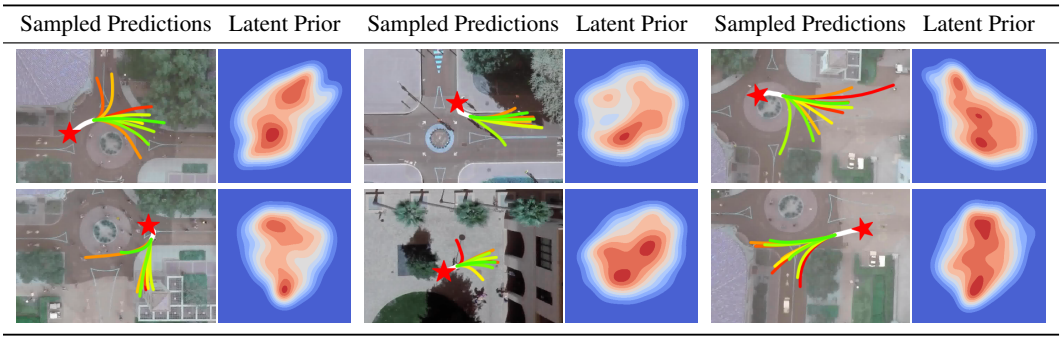

Figure 2: Randomly sampled predictions of our CF-VAE model on the Stanford Drone dataset. We observe that our prediction are clearly multi-modal and is reflected by the Conditional Flow Priors. Note, our 64D CF-VAE latent distribution is (approximatly) projected to 2D using tSNE and KDE.

dense in traffic participants with multi-model trajectories. Prior work follows two different evaluation protocols, 1. (Lee et al., 2017; Bhattacharyya et al., 2018; Pajouheshgar and Lampert, 2018) use 5 fold cross validation, 2. (Robicquet et al., 2016; Sadeghian et al., 2018, 2019; Deo and Trivedi, 2019) use a single standard train-test split. We evaluate our CF-VAE using the first protocol in Table 2 and the second in the Appendix G.

In addition to these state of the art models, (Pajouheshgar and Lampert, 2018) suggests a "Shotgun" baseline. This baseline obtains results at par with the state-of-the-art because it a good template which covers the most likely possible futures (modes) for traffic participant motion in this dataset. We report the results using 5 fold cross validation in Table 2. We additionally compare to a mixture of Gaussians prior (details in Appendix D). We use the same model architecture as in (Bhattacharyya et al., 2018) and a CNN encoder with attention to extract features from the last observed RGB image (details in Appendix C). These visual features serve as additional conditioning (x_m) to our Conditional Flow model. We see that our regularized CF-VAE model with RGB input and $C = 0.2$ performs best – outperforming the state-of-art "Shotgun" and BMS-CVAE by over 20% (Error @ 4sec). We see that our conditional flows are able to utilize visual scene (RGB) information to improve performance (3.5 vs 3.6 Error @ 4sec). We also see that the MoG-CVAE and our CF-VAE outperforms the BMS-CVAE, even without visual scene information. This again reinforces our claim that the standard Gaussian prior induces a strong model bias and data dependent multi-modal priors are needed for best performance. The performance advantage of CF-VAE over the MoG-CVAE again illustrates the advantage of normalizing flows at learning complex conditional multi-modal distributions. The performance advantage over the "Shotgun" baseline shows that our CF-VAE not only learns to capture the correct modes but also generates more fine-grained predictions.

4 Conclusion

In this work, we presented the first variational model for learning multi-modal conditional data distributions with Conditional Flow based priors – the Conditional Flow Variational Autoencoder (CF-VAE). Our rigorous experiments on diverse sequence prediction datasets show that our CF-VAE achieves state-of-the-art results. Furthermore, we address degenerate solutions leading to latent variable collapse using fixed variance posteriors. Additionally, we also show that our powerful Conditional Flow Variational Autoencoder can take advantage of diverse sources of conditioning information including scene context and interacting agents, leading to state of the art performance.

References

L. Ardizzone, J. Kruse, S. Wirkert, D. Rahner, E. W. Pellegrini, R. S. Klessen, L. Maier-Hein, C. Rother, and U. Köthe. Analyzing inverse problems with invertible neural networks. In *ICLR*, 2019.

A. Atanov, A. Volokhova, A. Ashukha, I. Sosnovik, and D. Vetrov. Semi-conditional normalizing flows for semi-supervised learning. In *ICML Workshop*, 2019.

M. Babaeizadeh, C. Finn, D. Erhan, R. H. Campbell, and S. Levine. Stochastic variational video prediction. In *ICLR*, 2018.

A. Bhattacharyya, B. Schiele, and M. Fritz. Accurate and diverse sampling of sequences based on a "best of many" sample objective. In *CVPR*, 2018.

X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel. Variational lossy autoencoder. In *ICLR*, 2017.

E. D. De Jong. The mnist sequence dataset. `https://edwin-de-jong.github.io/blog/mnist-sequence-data/`, 2016. Accessed: 2019-07-07.

N. Deo and M. M. Trivedi. Scene induced multi-modal trajectory forecasting via planning. In *ICRA Workshop*, 2019.

L. Dinh, D. Krueger, and Y. Bengio. Nice: Non-linear independent components estimation. In *ICLR*, 2015.

X. Gu, K. Cho, J.-W. Ha, and S. Kim. Dialogwae: Multimodal response generation with conditional wasserstein auto-encoder. *arXiv preprint arXiv:1805.12352*, 2018.

A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, 2018.

M. D. Hoffman and M. J. Johnson. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *NIPS Workshop*, 2016.

G. Holmes. The use of hyperbolic cosines in solving cubic polynomials. *The Mathematical Gazette*.

D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *NeurIPS*, 2018.

N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *CVPR*, 2017.

Y. Lu and B. Huang. Structured output learning with conditional generative flows. In *ICML Workshop*, 2019.

E. Pajouheshgar and C. H. Lampert. Back to square one: probabilistic trajectory forecasting without bells and whistles. In *NeurIPs Workshop*, 2018.

A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, 2016.

M. Rosca, B. Lakshminarayanan, , and S. Mohamed. Distribution matching in variational inference. *arXiv preprint arXiv:1802.06847*, 2019.

A. Sadeghian, F. Legros, M. Voisin, R. Vesel, A. Alahi, and S. Savarese. Car-net: Clairvoyant attentive recurrent network. In *ECCV*, 2018.

A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, S. H. Rezatofighi, and S. Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *CVPR*, 2019.

K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. In *NIPS*, 2015.

E. G. Tabak, E. Vanden-Eijnden, et al. Density estimation by dual ascent of the log-likelihood. In *Communications in Mathematical Sciences*, volume 8, 2010.

J. M. Tomczak and M. Welling. Vae with a vampprior. In *AISTATS*, 2018.

L. Wang, A. Schwing, and S. Lazebnik. Diverse and accurate image description using a variational auto-encoder with an additive gaussian encoding space. In *Advances in Neural Information Processing Systems*, pages 5756–5766, 2017.

T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *CVPR*, 2019.

Z. M. Ziegler and A. M. Rush. Latent normalizing flows for discrete sequences. In *ICML*, 2019.

Appendix A. ELBO with Conditional Non-Linear Normalizing Flows

First, we provide a complete derivation of (4). We begin from (1) and use (3) to determine the KL divergence,

$$\begin{aligned} -D_{\mathrm{KL}}(q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})||p_\psi(\mathrm{z}|\mathrm{x})) &= -\mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})}\log(q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})) + \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})}\log(p_\psi(\mathrm{z}|\mathrm{x})) \\ &= \mathcal{H}(q_\phi) + \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})}\log(p(\epsilon|\mathrm{x})) + \sum_{i=1}^{n}\log(|\det J_i|). \end{aligned} \tag{5}$$

Plugging 5 into (1) gives us (4).

Next, we describe the backward operation of our non-linear conditional normalizing flow. Note that while the forward operation is necessary to compute the likelihood in (4) during training, the forward operation is necessary to sample from the latent prior distribution of our CF-VAE. We use conditional non-linear normalizing flows with split coupling. Split couplings ensure invertibility by applying a flow layer f_i on only half of the dimensions at a time. To compute (3), we split the dimensions z^D of the latent variable into halfs, $\mathrm{z}^L = \{1, \cdots, {}^D\!/_2\}$ and $\mathrm{z}^R = \{{}^D\!/_2, \cdots, d\}$ at each invertible layer f_i. We then apply the following transformation each dimension z^j alternatively from z^L or z^R,

$$f_i^{-1}(\mathrm{z}^j|\mathrm{z}^R,\mathrm{x}) = \epsilon^j = a(\mathrm{z}^R,\mathrm{x}) + b(\mathrm{z}^R,\mathrm{x}) \times \mathrm{z}^j + \frac{c(\mathrm{z}^R,\mathrm{x})}{1 + \big(d(\mathrm{z}^R,\mathrm{x}) \times \mathrm{z}^j + g\big(\mathrm{z}^R,\mathrm{x}\big)\big)^2}. \tag{6}$$

where, $\mathrm{z}^j \in \mathrm{z}^L$. Note that, in (6) and in the corresponding forward operation f_i, the coefficients $\{a, b, c, d, g\} \in \mathbb{R}$ are functions of both the other half of the dimensions of z *and* the condition x unlike Ziegler and Rush (2019). Thus, the latent prior distribution on z is conditioned on x.

Next, we describe the forward operation. The forward operation consists of solving for the roots of the following equation (more details in (Ziegler and Rush, 2019)),

$$\begin{aligned} &- bd^2(\epsilon^j)^3 + ((\mathrm{z}^j - a)d^2 - 2dgb)(\epsilon^j)^2 \\ &+ (2dg(\mathrm{z}^j - a) - b(g^2+1))\epsilon^j + ((\mathrm{z}^j - a)(g^2+1) - c) = 0 \end{aligned} \tag{7}$$

This equation has one real root which can be found analytically (Holmes). As mentioned above, note that the coefficients $\{a, b, c, d, g\}$ are also functions of the condition x unlike Ziegler and Rush (2019).

Appendix B. Additional Evaluation of Conditional Non-Linear Flows

We compare conditional affine flows of (Atanov et al., 2019; Lu and Huang, 2019) and our conditional non-linear (Cond NL) flows in Figure 3 and Figure 4. We plot the conditional distribution $p(\mathrm{y}|\mathrm{x})$ and the corresponding condition x in the second and first columns. We use 8 and 16 layers of flow in case of the densities in Figure 3 and Figure 4 respectively. We see that the estimated density by the conditional affine flows of (Atanov et al., 2019; Lu and Huang, 2019) contains distinctive "tails" in case of Figure 3 and discontinuities in case of Figure 4. In comparison our conditional non-linear flows does not have distinctive "tails" or discontinuities and is able to complex capture the multi-modal distributions better. Note, the "ring"-like distributions in Figure 4 cannot be well captured by more traditional methods like Mixture of Gaussians. We see in Figure 5 that even with 64 mixture components, the learnt density is not smooth in comparison to our conditional non-linear flows. This again demonstrates the advantage of our conditional non-linear flows.

Appendix C. Additional Details of Our Model Architectures

Here, we provide details of the model architectures used across both the datasets.

MNIST Sequences. We use the same model architecture as in Bhattacharyya et al. (2018). The LSTM condition encoder on the input sequence x, the LSTM recognition network q_θ and the decoder LSTM network has 48 hidden neurons each. Also as in Bhattacharyya et al. (2018), we use a 64 dimensional latent space.

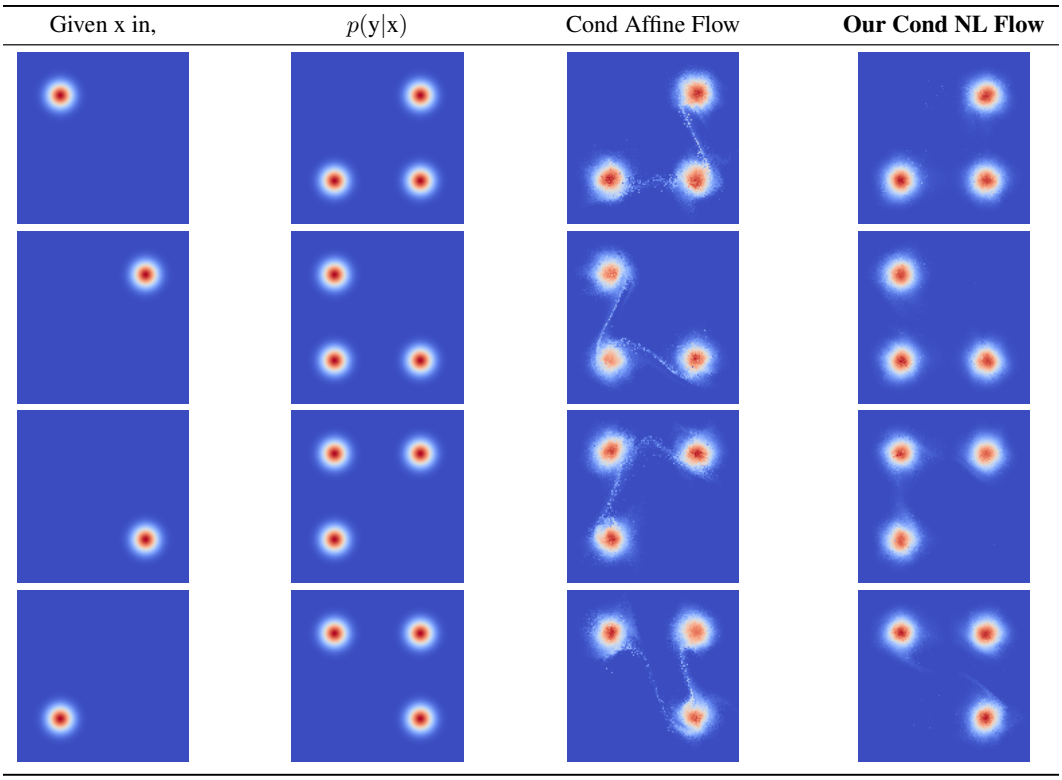

Figure 3: Comparison between conditional affine flows of (Atanov et al., 2019; Lu and Huang, 2019) and our conditional non-linear (Cond NL) flows. We see that the conditional affine flows cannot fully capture multi-modal distributions ("tails" between modes), while our conditional non-linear flows does not have distinctive "tails".

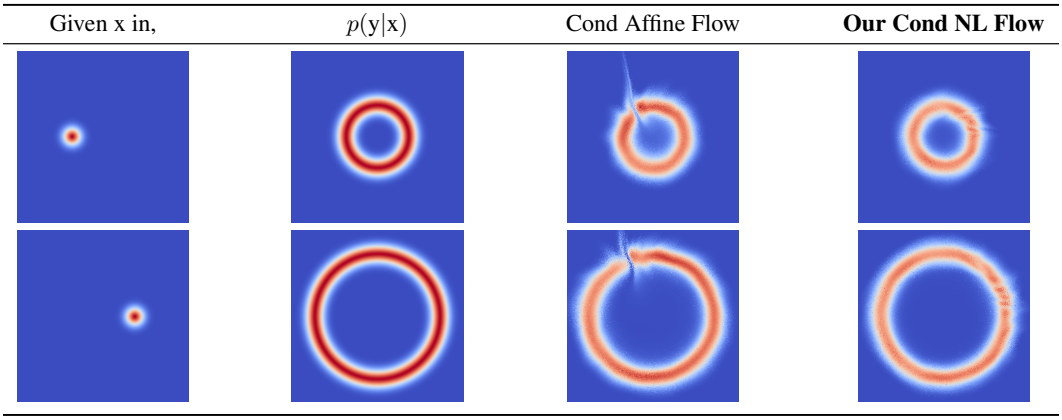

Figure 4: Comparison between conditional affine flows of (Atanov et al., 2019; Lu and Huang, 2019) and our conditional non-linear (Cond NL) flows. We see that the conditional affine flows cannot fully capture "ring"-like conditional distributions (note the discontinuity at the top), while our conditional non-linear flows does not have such discontinuities.

Stanford Drone. Again, we use the same model architecture as in Bhattacharyya et al. (2018) except for the CNN encoder. The LSTM condition encoder on the input sequence x and the decoder LSTM network has 64 hidden neurons each. The LSTM recognition network q_θ has 128 hidden neurons. Also as in Bhattacharyya et al. (2018), we use a 64 dimensional latent space. Our CNN encoder has 6 convolutional layers of size 32, 64, 128, 256, 512 and 512. We predict the attention weights on the final feature vectors using the encoding of the LSTM condition encoder. The attention weighted feature vectors are passed through a final fully connected layer to obtain the final CNN

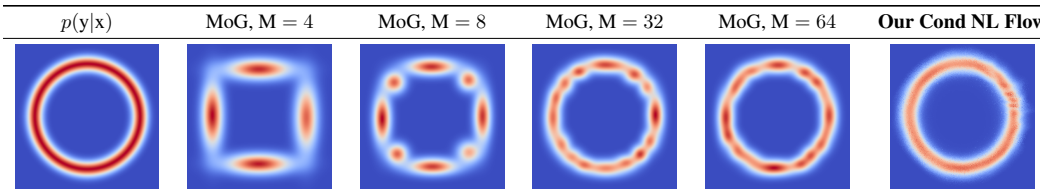

Figure 5: Comparison between our conditional non-linear (Cond NL) flows and a Mixture of Gaussians (MoG) model. We see that even with 64 mixture components, the learnt density is not smooth in comparison to our conditional non-linear flows.

encoding. Furthermore, we found it helpful to additionally encode the past trajectory as an image (as in (Pajouheshgar and Lampert, 2018)) as provide this as an additional channel to the CNN encoder.

Appendix D. Details of the mixture of Gaussians (MoG) baseline

In the main paper, we include results on the MNIST Sequence and Stanford Drone dataset with a Mixture of Gaussians (MoG) prior. In detail, instead of a normalizing flow, we set the prior to a MoG form,

$$p_\xi(\mathrm{z}|\mathrm{x}) = \sum_{i=1}^{M} p(\mathrm{c}_i|\mathrm{x})\mathcal{N}(\mathrm{z}; \mu_i, \sigma_i|\mathrm{x}). \tag{8}$$

We use a simple feed forward neural network that takes in the condition x and predicts the parameters of the MoG, $\xi = \{\mathrm{c}_1, \mu_1, \sigma_1, \cdots, \mathrm{c}_M, \mu_M, \sigma_M\}$. Note, to ensure a reasonable number of parameters, we consider spherical Gaussians. Similar to (4), the ELBO can be expressed as,

$$\log(p_\theta(\mathrm{y}|\mathrm{x})) \geq \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})} \log(p_\theta(\mathrm{y}|\mathrm{z},\mathrm{x})) + \mathcal{H}(q_\phi) + \mathbb{E}_{q_\phi(\mathrm{z}|\mathrm{x},\mathrm{y})} \log(p_\xi(\mathrm{z}|\mathrm{x})). \tag{9}$$

Note that we fix the entropy of the posterior distribution q_ϕ for stability.

Appendix E. Additional Evaluation on the MNIST Sequence Dataset

Here, we perform a comprehensive evaluation using the MoG prior with varying mixture components. Moreover, we experiment with a CVAE with unconditional non-linear flow based prior (NL-CVAE). We report the results in Table 3.

Table 3: Evaluation on MNIST Sequences (CLL: lower is better).

Method	-CLL
NL-CVAE	107.6
CVAE ($M = 1$)(Sohn et al., 2015)	96.4
MoG-CVAE, $M = 2$	85.3
MoG-CVAE, $M = 3$	84.6
MoG-CVAE, $M = 4$	85.7
MoG-CVAE, $M = 5$	86.3
CF-VAE	**74.9**

We see that the MoG-CVAE outperforms the plain CVAE. This again reinforces our claim that the standard Gaussian prior induces a strong model bias. We see that using $M = 3$ components with the variance of the posterior distribution fixed to $\mathrm{C} = 0.2$ leads to the best performance. This is expected as 3 is the most frequent number of possible strokes in the MNIST Sequence dataset. Also note that the results with the MoG prior are also relatively robust across $\mathrm{C} = [0.05, 0.2]$ as we learn the variance of the prior (see the section above). Finally, our CF-VAE still significantly outperforms the

MoG-CVAE (74.9 vs 84.6). This is expected as normalizing flows are more powerful compared to MoG at learning complex multi-modal distributions (Kingma and Dhariwal, 2018) (also see Figure 5).

We also see that using an unconditional non-linear flow based prior actually harms performance (107.6 vs 96.4). This is because the latent distribution is highly dependent upon the condition. Therefore, without conditioning information the non-linear conditional flow learns a global representation of the latent space which leads to out-of-distribution samples at prediction time.

Appendix F. Evaluation of the Robustness of the Top k% Metric

We use two simpler uniform "Shotgun" baselines to study the robustness of the Top k% metric against random guessing. In particular, we consider the "Shotgun"-u90° and "Shotgun"-u135° baselines which: given a budget of K predictions, it uniformly distributes the predictions between $(-90°, 90°)$ and $(-135°, 135°)$ respectively of the original orientation and using the velocity of the last time-step. In Table 4 we compare the Top 1 (best guess) to Top 10% metric with K= $50, 100, 500$ predictions.

Table 4: Five fold cross validation on the Stanford Drone dataset. Euclidean error at ($^1/_5$) resolution.

Method	K	Error @ 1sec	Error @ 2sec	Error @ 3sec	Error @ 4sec
		Top 1 (Best Guess)			
"Shotgun"-u90°	50	0.9	1.9	3.1	4.4
"Shotgun"-u90°	100	0.9	1.9	3.0	4.3
"Shotgun"-u90°	500	0.9	1.9	3.0	4.3
		Top 10%			
"Shotgun"-u90°	50	1.2	2.5	3.9	5.4
"Shotgun"-u90°	100	1.2	2.5	3.9	5.4
"Shotgun"-u90°	500	1.2	2.5	3.9	5.4
		Top 1 (Best Guess)			
"Shotgun"-u135°	50	0.9	2.0	3.1	4.5
"Shotgun"-u135°	100	0.9	1.9	3.0	4.3
"Shotgun"-u135°	500	0.9	1.9	3.0	4.2
		Top 10%			
"Shotgun"-u135°	50	1.4	2.9	4.5	6.2
"Shotgun"-u135°	100	1.4	2.9	4.5	6.2
"Shotgun"-u135°	500	1.4	2.9	4.5	6.2

We see that in case of both the "Shotgun"-u90° and "Shotgun"-u135° baselines, the Top 1 (best guess) metric improves with increasing number of guesses. This effect is even more pronounced in case of the "Shotgun"-u135° baseline as the random guesses are distributed over a larger spatial range. In contrast, the Top 10% metric remains remarkably stable. This is because, in order to improve the Top 10% metric, random guessing is not enough – the predictions have to be on the correct modes. In other words, the only way to improve the Top 10% metric is move random predictions to any of the correct modes.

Appendix G. Additional Evaluation on the Stanford Drone Dataset

We report results using the single train/test split of Robicquet et al. (2016); Sadeghian et al. (2018, 2019); Deo and Trivedi (2019) in Table 5. We use the minimum Average Displacement Error (mADE) and minimum Final Displacement Error (mFDE) metrics as in (Deo and Trivedi, 2019). The minimum is over as set of predictions of size K. Although this metric is less robust to random guessing compared to the Top $k\%$ metric, it avoids rewarding random guessing for a small enough value of K. We choose $K = 20$ as in Deo and Trivedi (2019). Similar to the results with 5 fold cross validation, we observe 20% improvement over the state-of-the-art.

Table 5: Evaluation on the Stanford Drone dataset using the split of Gupta et al. (2018); Zhao et al. (2019); Sadeghian et al. (2019); Deo and Trivedi (2019) (see also Table 2).

Method	mADE	mFDE
SocialGAN (Gupta et al., 2018)	27.2	41.4
MATF GAN (Zhao et al., 2019)	22.5	33.5
SoPhie (Sadeghian et al., 2019)	16.2	29.3
Goal Prediction (Deo and Trivedi, 2019)	15.7	28.1
CF-VAE (Ours)	**12.6**	**22.3**